



Proceedings of 52nd Annual Conference of the Southern African Computer Lecturers' Association (SACLA 2023)

Teach the future:
CS, IS, & IT Education in a changing world

19–21 July 2023, Muldersdrift
26 Degrees South Bush Boho Hotel
South Africa

Editors:
Henri van Rensburg
Dirk Snyman
Lynette Drevin
Günther Drevin

Editors

Henri van Rensburg
North-West University
Potchefstroom, South Africa

Dirk Snyman
University of Cape Town
Cape Town, South Africa

Lynette Drevin
North-West University
Potchefstroom, South Africa

Günther Drevin
North-West University
Potchefstroom, South Africa

Published online by the SACLA 2023 Organising Committee

School of Computer Science and Information Systems
North-West University
Private Bag X6001
Potchefstroom, South Africa
2520

ISBN: 978-0-6397-8221-8 (Electronic)

October 2023

Copyright Notice

All rights reserved. No part of this book may be reproduced, stored in a retrieval system, or transmitted, without prior written permission of the publisher. The SACLA 2023 organizing committee is not responsible for errors or omissions from individual papers contained in these proceedings. Technical electronic anomalies are possible and unavoidable during the compilation process. The publisher is not responsible for the accuracy and validity of information contained in these papers, nor is it responsible for the final use to which this book may be used.

© Copyright: The Authors

Preface

The 52nd Annual Conference of the Southern African Computer Lecturers' Association was held from July 19th to 21st, 2023, at the serene 26 Degrees South in Muldersdrift, South Africa. The event was co-located with the annual meeting of the South African Institute for Computer Scientists and Information Technologists (SAICSIT) for the second year running, thereby giving the delegates the opportunity to attend both events as there is a considerable overlap in the communities.

The theme of SACLA 2023, "Teach the Future: CS, IS, & IT Education in a Changing World," resonated deeply with the challenges and opportunities facing educators in today's fast-paced technological landscape. Over these three insightful days, we engaged in profound discussions, shared experiences, and embraced innovative approaches to ensure that our students are well-prepared for the ever-evolving demands of the digital age. Of note was the number of papers that addressed the challenges and opportunities afforded by the increasingly common use of large language models such as ChatGPT in the higher education landscape.

The international program committee comprised 44 members from more than 15 affiliations. A rigorous double-blind peer review process was followed for all submissions. Each submission was reviewed by at least three members of the program committee. This volume presents a collection of selected papers from the conference, representing the top ten submissions, based on reviewer ratings. This represents an acceptance rate of 24% for this volume from the 42 papers received. Reviewer feedback was provided to the authors, and they were requested to submit a change log and rebuttal to the program committee to indicate how any issues that were raised by the reviewers were addressed. A further 12 papers (29% acceptance rate) are included in the Springer CCIS volume 1862. More than 75% of the contributions in this online proceedings came from different universities. A best paper award was presented to Cheng-Wen Huang, Max Coleman, Daniela Gachago and Jean-Paul Van Belle for their paper entitled "*Using ChatGPT to Encourage Critical AI Literacy Skills and for Assessment in Higher Education*". The best paper was once more identified based on the ratings that were given by the reviewers.

In addition to the paper presentations, Jacob Spoelstra, Principal Data Scientist at Microsoft, was invited by the organising committee to give a keynote talk on "*ChatGPT demystified*" in which he elucidated the inner workings of ChatGPT and the underlying components of the large language model on which it is built. These include word embeddings, artificial neural networks, and attention mechanisms, shedding light on how they contribute to its exceptional performance. He further discussed the training process and the alignment of the models, based on human feedback. His talk equipped delegates to better understand this emerging technology and its implications in teaching and learning, and research. Furthermore, a workshop on curriculum content and challenges was conducted. Delegates were encouraged to share challenges and ideas in smaller groups, based on similar course content. Feedback on the commonly identified themes was presented during the annual general meeting of the association.

The organising committee would like to thank all the participants, including speakers, delegates, and reviewers for their contributions to a successful SACLA 2023 conference.

Finally, we wish to acknowledge the EasyChair conference management system which was used for managing the submissions and reviews of SACLA 2023 papers.

October 2023

Henri van Rensburg
Dirk Snyman
Lynette Drevin
Günther Drevin



Our Sponsors

We want to thank IITPSA, UiPath, the School of Computer Science and Information Systems:
North-West University, and Cengage for their contribution to SACLA 2023.



Message from the Conference Chair

Thank you for your participation in the 52nd Annual Conference of the South African Computer Lecturers' Association (SACLA 2023). I trust that you had a stimulating and informative conference at 26 Degrees South Bush Boho Hotel, Mulderdrift.

The goal of SACLA-conferences is to provide participants with an opportunity to share ideas, while maintaining an important level of academic input from all involved. Thank you for your enthusiastic participation in the curriculum session on the first afternoon, get-to-know-each-other exercise and sharing of ideas on module content and challenges. This enabled everyone to engage in stimulating discussions throughout the conference.

The accepted papers that were presented at the conference reflected current and future trends in teaching and learning in Computer Science, Information Systems and Information Technology. This is in keeping with the theme of SACLA 2023, Teach the future: CS, IS, & IT Education in a changing world.

We were fortunate to have had an international keynote speaker from the USA, Jacob Spoelstra – a principal data scientist from Microsoft, presenting on GPT Demystified. “The underlying components of GPT, including word embeddings, neural networks, and attention mechanisms, were presented, thereby shedding light on how they contribute to its exceptional performance.”

The HOD meeting was held on the first day, discussing important challenges and ideas within the different departments and institutions.

On behalf of the SACLA community, I wish to express our deepest appreciation to our sponsors, IITPSA, UiPATH, Cengage and the School of Computer Science and Information Systems, NWU.

A successful conference requires the effort of many individuals. We would like to thank the members of the program chair and committee for their hard work. We are grateful to the authors who presented their papers to this conference. The reviewers gave valuable feedback to authors of submitted papers. I also wish to extend my sincere thanks to all members of the organizing committee and congratulate them on a job well done. We thank the conference organizer, Mongoose C&D, for their planning and support of the logistics.

We hope that everyone had a good and informative time at SACLA 2023, 26 Degrees South Bush Boho Hotel.

Lynette Drevin
SACLA 2023 Conference Chair
School of Computer Science and Information Systems, North-West University

Message from the Program Committee Chair

It is with immense pleasure that I present to you the culmination of dedicated efforts and scholarly collaboration in shaping the program for the SACLA 2023 conference. The journey has been intense, filled with insightful reviews, and I extend my deepest gratitude to our diligent reviewers whose expertise has played a pivotal role in this process.

A special commendation goes out to our authors for their timely submissions, recognizing the critical role deadlines play in ensuring the smooth progression of the submissions and review process. Your willingness to share and contribute to the dialogue in this academic forum is truly commendable.

This year, we received 42 papers for review, a testament to the vibrant academic community engaged in Computer Science and Information Systems. Our program committee, consisting of both local and international experts, comprised 44 dedicated members, with 8 of them contributing from the international arena. The rigorous double-blind peer review process involved three reviewers for each paper, with additional expert opinions sought whenever needed.

We are thrilled to announce that 22 papers of exceptional quality have been accepted for presentation at SACLA 2023. Prior to publication, authors diligently incorporated corrections suggested by our peer reviewers. Among these, 12 outstanding papers will be featured in the prestigious Springer publication, *Communications in Computer and Information Science* and therefore, only the titles and brief abstracts of these papers are included in this publication.

Thanks again to our dedicated reviewers and authors for their invaluable contributions to SACLA 2023.

Henri van Rensburg
SACLA 2023 Program Committee Chair
School of Computer Science and Information Systems, North-West University

Organization

The 52nd Annual Conference of the Southern African Computer Lecturers' Association (SACLA 2023) was organized by the School of Computer Science and Information Systems, North-West University, South Africa.

Executive Committee

Conference chair

Lynette Drevin	North-West University (Potchefstroom Campus), South Africa
----------------	---

SACLA president

Estelle Taylor	North-West University (Potchefstroom Campus), South Africa
----------------	---

Program chairs

Henri van Rensburg	North-West University (Potchefstroom Campus), South Africa
Dirk Snyman	University of Cape Town, South Africa
Gunther Drevin	North-West University (Potchefstroom Campus), South Africa

Finances

Bobby Tait	University of South Africa, South Africa
Kobus van Aswegen	North-West University (Potchefstroom Campus), South Africa

Webmaster and Maintenance

Lance Bunt	North-West University (Vanderbijlpark Campus), South Africa
Marijke Coetzee	North-West University (Potchefstroom Campus), South Africa

Program committee

Shaun Bangay	Deakin University, Australia
Esiefarienrhe Bukohwo	North-West University (Mafikeng Campus), South Africa
Lance Bunt	North-West University (Vanderbijlpark Campus), South Africa
Alan Chalmers	The University of Warwick, England

Tendani Chimboza	University of Cape Town, South Africa
Marijke Coetzee	North-West University (Potchefstroom Campus), South Africa
Nosipho Dladlu	North-West University (Mafikeng Campus), South Africa
Tiny Du Toit	North-West University (Potchefstroom Campus), South Africa
Olaiya Folorunsho	Federal University Oye Ekiti, Nigeria
Lynn Futcher	Nelson Mandela University, South Africa
Leila Goosen	University of South Africa, South Africa
Irene Govender	University of KwaZulu-Natal, South Africa
Ruber Hernández-García	Universidad Católica del Maule, Chile
Barry Irwin	Noroff University College, Norway
Bassey Isong	North-West University (Mafikeng Campus), South Africa
Moemi Joseph	North-West University (Mafikeng Campus), South Africa
Christos Kalloniatis	University of the Aegean, Greece
Dimitrius Keykaan	North-West University (Potchefstroom Campus), South Africa
Eduan Kotzé	University of the Free State, South Africa
Wai Sze Leung	University of Johannesburg, South Africa
Janet Liebenberg	North-West University (Potchefstroom Campus), South Africa
Hugo Lotriet	University of South Africa, South Africa
Vusumuzi Malele	North-West University (Vanderbijlpark Campus), South Africa
Liesel Nel	University of the Free State, South Africa
Koketso Ntshabele	North-West University (Mafikeng Campus), South Africa
Jaco Pretorius	North-West University (Vanderbijlpark Campus), South Africa
Helen Purchase	Monash University, Australia
Linda Redelinghuys	North-West University (Potchefstroom Campus), South Africa
Rayne Reid	Noroff University College, Norway
Sebopelo Rodney	North-West University (Potchefstroom Campus), South Africa
Aslam Safila	University of Cape Town, South Africa
Rudi Serfontein	North-West University (Potchefstroom Campus), South Africa
Imelda Smit	North-West University (Vanderbijlpark Campus), South Africa
Hussein Suleman	University of Cape Town, South Africa

Estelle Taylor	North-West University (Potchefstroom Campus), South Africa
Alfredo Terzoli	Rhodes University, South Africa
Walter Uys	University of Cape Town, South Africa
Thomas van der Merwe	University of South Africa, South Africa
Charles van der Vyver	North-West University (Vanderbijlpark Campus), South Africa
Corné Van Staden	University of South Africa, South Africa

Keynote Presentation: Dr. Jacob Spoelstra



Bio

Jacob is a principal Data Scientist in Microsoft's Industry Solutions organization, part of a global innovation team focused on helping enterprise customers develop and deploy advanced AI solutions on Azure. At Microsoft he has created and led machine learning teams in both customer-facing and product groups, including Azure Machine Learning – developing services around language, recommenders and anomaly detection. In his three decades of experience in applied machine learning, he has held analytics leadership positions at several technology companies, including Opera Solutions, FICO and SAS Institute. Jacob earned BS and MS degrees in Electrical Engineering from the University of Pretoria, and a PhD in Computer Science from the University of Southern California. He and his wife, Tanya, have two boys, aged 19 and 21. They enjoy boating, hiking and snow sports. Jacob is a private pilot and is constantly looking for excuses to go flying. They live in San Diego, US.

GPT Demystified

In the half year since ChatGPT burst on the scene it has taken the world by storm as arguably the first system that exhibits real signs of artificial general intelligence, beyond (super) human performance in a narrow field, such as game playing. For once it feels like a lot of the hype is justified. Besides the uncanny ability to generate natural, human-like text, the GPT family of large language models have remarkable ability to answer factual questions, found uses as assistants for coding, provide medical and legal advice and even as therapists. Fascinating is that all of these are emergent properties, meaning that the model was not specifically designed for these tasks. So how does it do it?

We'll demystify the underlying components of GPT, including word embeddings, neural networks, and attention mechanisms, shedding light on how they contribute to its exceptional performance. We also discuss the training process and the alignment of models through human feedback. Although the essence of GPT's emergent properties remains elusive, understanding its limitations empowers us to harness its potential effectively.

Additionally, we showcase ongoing work on creating embodied AI agents that interact with humans in virtual environments, aiming to enhance human-machine interactions and create immersive experiences. Addressing the concerns surrounding superhuman artificial intelligence, we provide insights into the responsible application of AI and outline key principles to guide its usage. By approaching AI development and deployment mindfully, we can harness its potential for positive impact while minimizing potential risks.

Program

Wednesday, July 19, 2023		
11:00 - 17:00	Arrival and Registration Venue: Conference Lounge	
13:15 – 14:00	Tea / coffee on arrival Venue: Conference Lounge	
14:00 – 14:10	Welcome / Opening Venue: Thorn Tree	
	Venue: Rhino HOD-meeting (meeting of the head(s) of departments / subject groups)	Venue: Thorn Tree Session 1 Session chair: <i>Philip Machanick</i>
14:10 – 14:30	HoD-meeting (meeting of the head(s) of departments / subject groups)	Presentation 1_1: A framework for creating virtual escape rooms to teach computational thinking (#11). <i>Theane Janse van Rensburg and Machdel Mathee</i>
14:35 – 14:55		Presentation 1_2: Improving Student Participation in a Visual Application Programming Course at the University of Zululand (#28). <i>Thamsanqa Ndlovu and Alfredo Terzoli</i>
15:00 – 15:20		Presentation 1_3: Using Design Science Research to Enable Performance Prediction for IS&T Students (#26). <i>Rushil Raghavjee, Prabhakar Subramaniam and Irene Govender</i>
15:25 – 15:45		Presentation 1_4: Motivations and experiences of recognition of prior learning candidates in Information Systems postgraduate programmes (#24). <i>Thelma Chitsa and Gwamaka Mwalemba</i>
15:45 – 16:05	Tea / coffee Break Venue: Conference Lounge	
16:05 – 17:30	HoD-meeting (meeting of the head(s) of departments / subject groups) if time still needed	Curriculum activity – all delegates
17:00 – 17:30	Meeting of the SACLA Executive Committee	
18:30 – 21:00	Cocktail Dinner Venue: Boma	

Thursday, July 20, 2023	
9:00	Venue: Thorn Tree Session 1B starts Session chair: <i>Janet Liebenberg</i>
9:10 – 9:30	Presentation 1_5: An Alumni Satisfaction Model for Computing Departments (#5). <i>Andre Calitz, Margaret Cullen, Arthur Glaum and Jean Greyling</i>
9:35 – 9:55	Presentation 1_6: CS/IS/IT Programme Accreditation and SACAB support at Higher Education Institutions in Southern Africa (#34). <i>Estelle Taylor, Andre Calitz and Anastasia Petratos</i>
10:00 – 10:30	Sponsor: IITPSA
	Venue: Thorn Tree Session 2 Session chair: <i>Matthew Adigun</i>
10:35 – 10:55	Presentation 2_1: Exploring flipped learning in an introductory programming module: a literature review (#14). <i>Nita Mennega and Tendani Mawela</i>
11:00 – 11:20	Presentation 2_2: Transitioning an Introductory Programming course into a Blended Learning format (#27). <i>Aslam Safla, Hussein Suleman and James Gain</i>
11:20 – 11:40	Tea / coffee break Venue: Conference Lounge
11:40 – 12:00	Presentation 2_3: A Consolidated Catalogue of Question Types for Programming Courses (#20). <i>Anitta Thomas</i>
12:05 – 12:25	Presentation 2_4: Demystifying the Impact of ChatGPT on Teaching and Learning (#39). <i>Tapiwa Gundu and Collin Chibaya</i>
12:30 – 12:50	Presentation 2_5: Using ChatGPT to Encourage Critical AI Literacy Skills and for Assessment in Higher Education (#29). <i>Cheng-Wen Huang, Max Coleman, Daniela Gachago and Jean-Paul Van Belle</i>
12:55 – 13:15	Presentation 2_6: Beyond the Hype: A Cautionary Tale of ChatGPT in the Programming Classroom (#38). <i>Grant Oosterwyk, Pitso Tsibolane, Popyeni Kautondokwa and Ammar Canani</i>
13:15 – 14:00	Lunch Venue: Chowbaby
	Venue: Thorn Tree Session 3 Session chair: <i>Grant Oosterwyk</i>
14:00 – 14:20	Presentation 3_1: Blended Agile Learning of Computer Architecture under Covid (#18). <i>Philip Machanick</i>
14:25 – 14:45	Presentation 3_2: Analysis of students' behaviour in Java programming class in blended learning environment using Process Mining techniques (#43). <i>Bukohwo Michael Esiefarienrhe and Thusoyaone Joseph Moemi</i>

14:50 – 15:10	Presentation 3_3: Value of explicit instruction in teaching computer programming to post-graduate students: The Kirkpatrick Training Evaluation Model (#17). <i>Pakiso Khomokhoana and Ruth Wario</i>
15:15 – 15:35	Presentation 3_4: Factors determining the success of online learning videos for programming (#40). <i>Janet Liebenberg and Suné Van der Linde</i>
15:35 – 16:00	Tea / coffee break Venue: Conference Lounge
16:00 – 17:00	Venue: Thorn Tree Keynote address: <i>Dr Jacob Spoelstra</i> Jacob is a principal Data Scientist in Microsoft's Industry Solutions organization, part of a global innovation team focused on helping enterprise customers develop and deploy advanced AI solutions on Azure.
18:30 - late	Venue: Lobola Gala Dinner

Friday, July 21, 2023	
	Venue: Thorn Tree Session 4 Session chair: <i>Hussein Suleman</i>
8:35 – 8:55	Presentation 4_1: A Minimum Viable Automated Assistant Architecture for ICT Project-Based Assessments (#41). <i>Jacqui Muller and Marijke Coetzee</i>
9:00 – 9:20	Presentation 4_2: Reducing Project Contention in an Undergraduate Project Allocation System (#31). <i>Stephen Levitt and Ken Nixon</i>
9:25 – 9:45	Presentation 4_3: Increasingly Forgetful and Distracted? A cross-temporal meta-analysis of everyday cognitive failures among university students (#1). <i>Daniel B le Roux, Bianca Finsterbusch, Jurgen Saunders and Bronte De Agrela</i>
09:45 – 10:45	AGM
10:45 – 11:25	Tea / coffee break Venue: Conference Lounge
11:30 – 11:50	Presentation 4_4: Towards Using African Thought to Decolonise the Curriculum– A Scientific Perspective (#25). <i>Matthew Adigun and Skhumbuzo Zwane</i>
11:55 – 12:15	Presentation 4_5: The Impact of Social Media Use on Digital Well-being of University Students (#19). <i>Enwill Isaks, Zane Davids, Lisa Seymour and Sharon Geeling</i>
12:20 – 12:40	Presentation 4_6: First Year Computing Students' Career Choice Influencers (#15). <i>Margaret Cullen, Andre Calitz, Malibongwe Twani and Jean Greyling</i>
12:40	Closing
13:00 – 14:00	Lunch Venue: Chowbaby

Table of Contents

Student centered teaching and learning

A Minimum Viable Automated Assistant Architecture for ICT Project-Based Assessments	1
<i>Jacqui Muller and Marijke Coetzee</i>	
Analysis of students' behaviour in Java programming class in blended learning environment using Process Mining techniques	17
<i>Bukohwo Michael Esiefarienrhe and Thusoyaone Joseph Moemi</i>	
Blended Agile Learning of Computer Architecture under Covid	30
<i>Philip Machanick</i>	
Improving Student Participation in a Visual Application Programming Course at the Univer- sity of Zululand	47
<i>Thamsanqa Ndlovu</i>	
Using Design Science Research to Enable Performance Prediction for IS&T Students	59
<i>Rushil Raghavjee, Prabhakar Subramaniam and Irene Govender</i>	

AI and future movements; Programming

Beyond the Hype: A Cautionary Tale of ChatGPT in the Programming Classroom	74
<i>Grant Oosterwyk, Pitso Tsibolane, Popyeni Kautondokwa and Ammar Canani</i>	

Beyond the classroom

Towards Using African Thought to Decolonise the Curriculum– A Scientific Perspective	92
<i>Matthew Adigun and Skhumbuzo Zwane</i>	
The Impact of Social Media Use on Digital Well-being of University Students	108
<i>Enwill Isaks, Zane Davids, Lisa Seymour and Sharon Geeling</i>	
Increasingly Forgetful and Distracted? A cross-temporal meta-analysis of everyday cognitive failures among university students	125
<i>Daniel B le Roux, Bianca Finsterbusch, Jurgen Saunders and Bronte De Agrela</i>	
CS/IS/IT Programme Accreditation and SACAB support at Higher Education Institutions in Southern Africa	141
<i>Estelle Taylor, Andre Calitz and Anastasia Petratos</i>	

Springer publication (title and abstract)

Student centered teaching and learning

A framework for creating virtual escape rooms to teach computational thinking	159
<i>Theane Janse van Rensburg and Machdel Matthee</i>	
Value of explicit instruction in teaching computer programming to post-graduate students: The Kirkpatrick Training Evaluation Model	160
<i>Pakiso Khomokhoana and Ruth Wario</i>	
Reducing Contention in an Undergraduate Capstone Project Allocation System	161
<i>Stephen Levitt and Ken Nixon</i>	
Factors determining the success of online learning videos for programming	162
<i>Janet Liebenberg and Suné Van der Linde</i>	
Exploring flipped learning in an introductory programming module: a literature review	163
<i>Nita Mennega and Tendani Mawela</i>	
Transitioning an Introductory Programming course into a Blended Learning format	164
<i>Aslam Safla, Hussein Suleman and James Gain</i>	

AI and future movements; Programming

Demystifying the Impact of ChatGPT on Teaching and Learning	165
<i>Tapiwa Gundu and Collin Chibaya</i>	
Using ChatGPT to Encourage Critical AI Literacy Skills and for Assessment in Higher Education	166
<i>Cheng-Wen Huang, Max Coleman, Daniela Gachago and Jean-Paul Van Belle</i>	
A Consolidated Catalogue of Question Types for Programming Courses	167
<i>Anitta Thomas</i>	

Beyond the classroom

An Alumni Satisfaction Model for Computing Departments	168
<i>Andre Calitz, Margaret Cullen, Arthur Glaum and Jean Greyling</i>	
Motivations and Experiences of Recognition of Prior Learning Candidates in Information Systems Programmes	169
<i>Thelma Chitsa and Gwamaka Mwalemba</i>	
First Year Computing Students' Career Choice Influencers	170
<i>Margaret Cullen, Andre Calitz, Malibongwe Twani and Jean Greyling</i>	

Author Index	171
---------------------------	-----

A Minimum Viable Automated Assistant Architecture for ICT Project-Based Assessments

J Muller ¹[0000-0003-2709-5444] and M Coetzee ²[0000-0002-9157-3079]

School of Computer Science and Information Systems, Unit for Data Science and Computing
North-West University, South Africa,
¹jacqui.jm77@gmail.com, ²marijke.coetzee@nwu.ac.za

Abstract. Automated assistants for marking student assignments in a university setting, focused on ICT-based modules, can provide significant benefits for instructors, such as reducing workload and ensuring consistent and objective evaluations of student work. However, existing automated assessment systems have limitations and tend to focus on marking programming assignments. In addition, there is a lack of research on assessing project-based tasks in a flexible, accurate, consistent, and cost-efficient manner. This paper introduces an automated assistant architecture that not only assesses assignments but also supports instructors in the assessment process, from creating rubrics to identifying at-risk students. Several requirements are determined to guide the design of the automated assistant architecture. A proof of concept was developed, using Design Science Research (DSR), to illustrate the viability of the automated assistant architecture. The automated assistant architecture was instantiated for a large class within a domain where considerable improvements in time and cost were recorded versus performing manual assessments. Finally, the paper concludes with recommendations for successfully implementing automated assessment architectures in various educational settings.

Keywords: Automated Assistants, Architecture, Process Automation, Automated Assessment Grading.

1 Introduction

Automated assistants for marking student assignments in an ICT setting can provide numerous benefits for lecturers (Chen & Chen, 2019). Automatic marking can help to reduce the workload associated with manual grading (Messer, 2022), freeing up time for lecturers to focus on other aspects of their work. Additionally, automated marking can help ensure consistent and objective evaluations of student work and provide valuable insights into student performance (Lee *et al.*, 2017). However, benefits such as cost-savings are primarily determined by the architecture and technology stack implemented.

The traditional method of assessing student work is time-consuming, resource-intensive, and subjective. In addition, with the increasing demand for online education, there is a need for an efficient and reliable way to assess student work without

compromising the quality of feedback. An automated assessment architecture can potentially solve this problem. However, there is a lack of research on how instructors can assess the different types of tasks that are part of project-based work in a flexible, accurate, consistent as well as time and cost-efficient manner. This paper introduces the concept of an automated assistant architecture that provides more than just assessing assignments. The aim is to make use of an architecture that supports an instructor with the assessment process, from creating rubrics to identifying at-risk students, while capturing metrics to determine the level of success and saving of the implementation.

In this paper, the background context to the research problem is provided in Section 2. Section 3 briefly describes the design science research methodology. a literature review identifies the strengths and weaknesses of existing automated assessment systems to solve the research problem in Section 2. Next, a novel automated assistant architecture model is developed and tested that addresses the limitations of existing models. A mechanism to identify at-risk students is demonstrated. The reliability of the automated assessment architecture is evaluated to provide consistent and unbiased feedback. The improvement in time and cost versus manually assessing student submissions presented. Finally, recommendations are made on successfully implementing automated assessment architectures in various educational settings.

2 Background Context

Problem-based and project-based learning are well-established pedagogy in computer science and by extension, ICT learning environments. Students learn by attempting open-ended problems facilitated by an instructor to achieve a course's learning outcomes. Students can learn by integrating educational content with their experiences (Duffy & Jonnasen, 1992). Learning opportunities are experiential and aim to develop problem-solving skills and critical-thinking abilities (Jonnasen, 2000). For an ICT setting, these are essential learning outcomes. Furthermore, project-based learning ensures increased engagement with professional practice (Fincher & Petre, 1998).

The application of project-based learning at the authors' institution requires a large group of 270 third-year students to participate in a module where they must implement five individual projects. The aim is to expose them to new technologies and experiential learning to prepare them for the industry. In addition, students are expected to acquire several certifications and badges by completing online courses in their own time. By the end of the course, a successful student is expected to gain an appreciation of full-stack development, design patterns, source control, testing approaches and data analytics. The student submits a final portfolio of evidence that collates all work done throughout the semester. To illustrate to heavy workload, consider the following. All students are required to create a GitHub repository for project management of the project and to commit all code. Instructors need to evaluate how code was committed over a time period. If they had to manually inspect the commit history of each of the 270 students, it would be a daunting task. For this purpose, an automated assistant could automatically access a student's commit history and evaluate the range of dates the student made commits to update this to a student rubric. Such an assessment would provide

a more detailed and accurate evaluation of a student's ability to complete a project more professionally.

A summative assessment process is followed, where each of the five projects is graded using a comprehensive rubric. A rubric and a portfolio of evidence must be created for each student for each project. Rubrics grade different aspects of projects like using GitHub for source control, API methods implemented, design pattern code hierarchies, automated testing strategies using robotic process automation (RPA), and analytic views of data using Power BI. An agile software development process is followed where students report on each sprint completed as a retrospective and upload all certifications acquired for the project being completed. Two lectures evaluate the assessments, making the task extremely daunting. Students need quick feedback as a project completes, to be able to apply their knowledge in the next project. An automated assessment tool, tailor-made for the task, is therefore imperative to use. The automated assessment tool should realise multiple benefits, including time-saving, efficiency improvement, increased consistency, increased throughput rates, and reduced employee workload, among other benefits.

3 Research methodology

The design of the automated assistant architecture is performed using a design science research (DSR) approach. DSR is a cyclical research approach that focuses on designing, developing, and evaluating innovative artefacts or solutions to real-world problems in specific domains (Simonb 2019). DSR involves three interrelated cycles: the relevance cycle, the design cycle, and the rigour cycle. These cycles are illustrated in Figure 1 and are explained by Peffers et al. (2007). According to Hevner (2007), this approach combines rigorous scientific methods with practical problem-solving skills to create innovative solutions that are both effective and efficient.

The relevance cycle focuses on identifying relevant research problems. In the case of designing an automated assistant architecture, the research focuses on understanding the environmental needs of students and lecturers, the assessment processes, as well as various technologies and architectures that can be utilized as inputs. This understanding provides the foundation for the subsequent design and evaluation of the solution. The design cycle considers the development of theories and artefacts that integrate relevant knowledge from the available knowledge base to create new knowledge that can be utilized in the design of the solution as innovative artefacts or solutions. The rigour cycle focuses on evaluating the effectiveness of the artefacts or solutions. This research would involve evaluating the effectiveness of the solution in addressing the identified problems and meeting the environmental needs of students and lecturers.

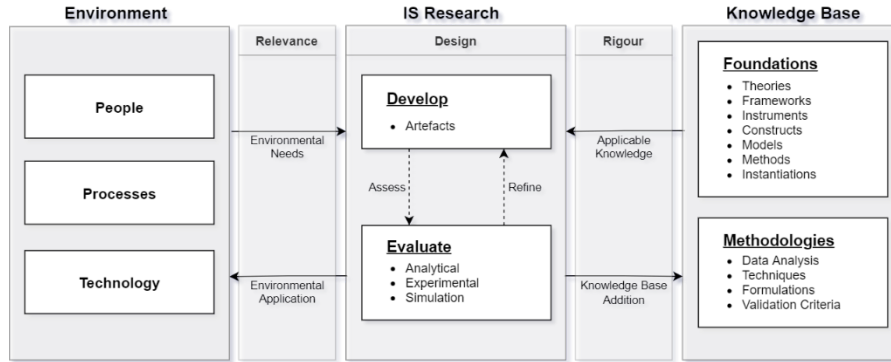


Fig. 1. Design Science Research

Building upon the principles and iterative nature of DSR, this study considers the context provided throughout the research as a foundation for an initial general design phase which is then extended into a “proof-of-concept” (POC) build and evaluation phase. Based on the outcomes obtained from the POC, the next phase involves enhancing the existing design of the generic framework, as depicted in Figure 1.

4 Related work

Automated assessment systems for higher education programming courses are an active field of research, as manual assessments are very time-consuming (Gordillo 2019). A constant in the world is that the number of people learning programming increases while the size of the teaching staff stays constant. More than 100 automated assessment systems for programming assignments have been reported in the literature (Barra et al., 2020). Automated assessment systems drastically reduce the workload of teachers, thus enabling them to give more assignments. Furthermore, the consistency and objectivity of assessments are improved and can even be performed better than if done by humans (Ala-Mutka 2005). Even if it is rudimentary, the feedback generated by assessment systems can provide students with information about the quality of their solutions and guidance on how to improve them and help them to improve their skills. In large groups, such personalised feedback is imperative to provide.

General concerns raised in literature are related to the types of assignments that should be assessed; the methods and techniques to analyse code; the types of systems that can support automated assessments; the feedback to students and instructors; how the system is integrated into the learning process and the quality and impact of assessments. The integrated evaluation of project-related tasks, such as submitting Google forms, tracking GitHub repositories, and completing sprint retrospectives, to name but a few, are not addressed. Current trends in this field include open-source tools and frameworks, machine-learning models, and feedback mechanisms. By leveraging existing resources, developers of automated assistant systems can focus on customising tools to their specific needs rather than building everything from scratch. As instructors

may have varying requirements, a more generalised architecture may be helpful that can be customised according to the needs of a specific learning activity.

Current research points towards the integration of open-source tools and frameworks for automated grading (Aiken, 2019), (Zanoni et al., 2021). Del Pino et al. (2012) identify four main features of automated code assessment systems: organising the assignments, receiving and storing learners' submissions, supporting automatic or semi-automatic assessment and providing feedback.

The focus of current research is generally on the detailed assessment of programming assignments instead of on the whole process of the automated creation of individual student rubrics and the population of evaluation rubric criteria for various types of tasks. Current trends in this field include open-source tools and frameworks, machine-learning models, and feedback mechanisms. By leveraging existing resources, developers of automated assistant systems can focus on customising tools to their specific needs rather than building everything from scratch. As instructors may have varying requirements, a more generalised architecture may be helpful that can be customised according to the needs of a specific learning activity.

5 Design of a generalised automated assistant architecture

This research takes a generic approach to automated assessments, whereby various domains and activities can be accommodated within the architecture. For this research, an *automated assistant architecture* is defined as an architecture that identifies a set of pluggable components that can be combined to perform a complete assessment process for project-based learning. In the example presented in this research, more tasks must be assessed than just programming code. Therefore, instructors define an instantiation of the architecture based on their domain requirements.

To define the components of the architecture, process engineering is valuable as an optimised, automated assistant process can be created to improve efficiency and effectiveness (bin Mohd Yusoff et al., 2021). The process of automatically assessing student submissions is subject to constraints dictated by a specific domain. For example, in the case of the authors' project-based module, it is also required to evaluate whether a student has submitted a sprint report after a project and has completed a certification over and above the evaluation of the project code. In addition, some domains may require adding steps to the process using pluggable components.

The design of the automated assistant architecture is driven by a set of requirements, defined next. Thereafter key components are described, and their interaction illustrated. The importance of a 3-tier software architecture is motivated for process automation. This research adopts a minimum viable architecture that focuses on the core, foundational architectural components to construct a working base to ensure agility. Thereafter the 3 tiers of the architecture is described.

5.1 Automated assistant architecture requirements

The focus of the automated assistant architecture is to support instructors with the assessment of various tasks that are part of a large project. This is especially important for exit-level ICT students who must deliver a project at the end of their studies. Requirements that drive the development of the architecture of the automated assistant are as follows:

- The automated assistant architecture is instructor driven.
- The automated assistant architecture uses pluggable open-source and vendor tools/architectures that are cost-effective and straightforward to use.
- A project needs to be scaffolded into smaller tasks that can be assessed individually.
- A general task rubric should be defined that identifies all assessment criteria.
- The general task rubric should follow a consistent format which enables the automated assessment of rubric criteria.
- A rubric for each task should be created for each student on a class list.
- All student data should be stored securely and in a central repository for administrative, analytics and auditing purposes.
- An automated tool should be used to assess a task, such as a programming assignment, the upload of a certificate, a report of a sprint, data tables in a database, or elements of a GitHub project, to name but a few.
- The results from the automated tool should be used to populate each student rubric.
- A report should be based on the centralised student data source to allow instructors insight into student progression.

As a starting point to almost any ICT-based project, a project brief and rubric would need to be developed to translate the requirement and expected outcomes of the assessment to the student. The rubric would be used as a template which all student rubrics should be based off for the assessment of that specific project, in accordance with the class list. The rubric should remain consistent and contain enough detail to holistically assess the student's ability to meet the requirements to pass the project.

All student submissions should be stored in a central repository, like a Learning Management System (LMS) or even GitHub. An automated tool can be created to assess the tasks contained within the student submissions and assign a mark to the student through their automatically generated rubric. All student results can be concatenated into a single data source, with a single view from which analytics can be derived to identify different categories of students based on their performance. The automated tool should be instructor-driven for the initial phases of implementation, meaning that the instructor should manually kick off the automated process per project to ensure that the automated tool runs correctly. It is also important that the instructor moderates the content generated by the automated tool for the initial iterations of the design cycles to evaluate the accuracy and consistency of the tool, compared to the marking criteria.

5.2 Key components

The overarching key components that should exist within an automated assessment assistant are shown in Figure 2. The process should at least include accessing a standard rubric template and creating a rubric for each student from a class list, accessing and assessing student submissions, capturing marks and disseminating the auto-assessed marks. Multiple additional components may be added to enhance the solution, such as the ability to identify at-risk students.

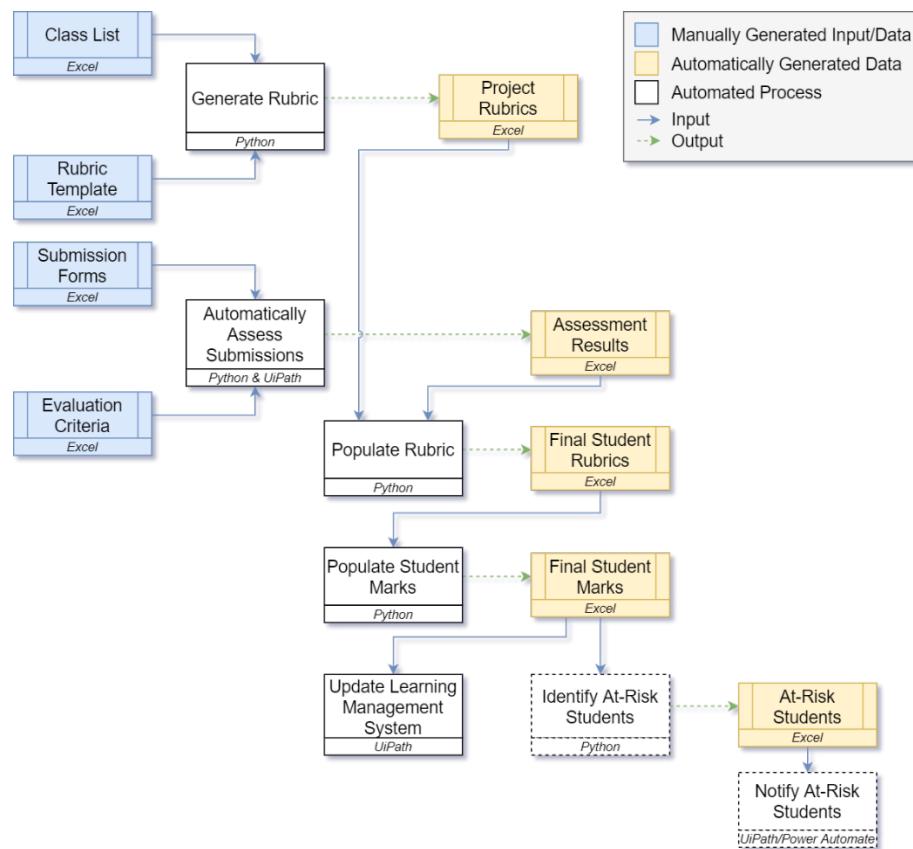


Fig. 2. High-level process for key components

Figure 2 indicates technologies such as Python, UiPath and Power Automate are used to automate the process. UiPath was used to execute all Robotic Process Automation (RPA) tasks in this implementation. RPA is software that mimics the interactions of an instructor with a computer (Ribeiro et al, 2021). It can perform rule-based tasks such as reading an Excel spreadsheet containing a rubric, and a classlist, and generating a student rubric for each student in a folder on a computer. When a student updates a GitHub repository, an RPA script can access each student's repository, read commit history, perform date calculations, update a student rubric with a mark, and add a

comment based on the evaluation. Therefore, RPA can perform any task the instructor can do by following a set of rules in an automation script, thereby saving time and cost. Automation tools such as RPA can be seen as the backbone of the proposed architecture.

5.3 Minimum Viable Architecture

A minimum viable architecture is an architecture that includes the most important components to ensure a working implementation. From this foundation, the architecture can continually be improved and expanded (Pirker & Lechner, 2019). For this research, various technologies and platforms exist that could enable the architecture. The architecture could be expanded to include anti-plagiarism tools, code coverage tools, automated testing solutions and more based on the requirements within a specific environment.

The execution of scripts within the architecture may vary from 'fully attended (instructor-driven)' to 'partially unattended (no human intervention required)', depending on the maturity of the implementation (Axmann & Harmoko, 2020). For example, fully attended execution would refer to the instructor manually executing the scripts, while unattended execution would refer to the tool executing all processes, in the correct order, without any intervention from a person..

5.4 Automated assistant architecture design

Software architecture is essential in developing process automation because it provides a systematic approach to designing, implementing, and maintaining software systems that can automate business processes (Sobhy et al., 2021). Software architecture in process engineering ensures scalability, flexibility, reusability, maintainability, integration with other systems, and reliability. By having a solid software architecture, developers can build robust and efficient process automation systems that meet the needs of their organisations and are scalable. Software architecture is made up of multiple layers, with each layer maintaining responsibility over one major component category of the solution (Gomaa, 2017). A typical 3-tier architecture consists of a client (presentation) tier, an application (logic or integration) tier, and a database (data) tier (Hussain & Zomayda, 2017). One of the most significant advantages of using an n-tier architecture is the ability to make changes while minimising the impact such a change would have on the entire solution. For example, if a data source were to be changed, only the direct data access mechanism would need to change instead of the whole solution. In the next sections, each of the layers are described, starting from the data layer.

Data Layer

A data layer in software architecture separates an application's data storage and management component from the other parts of the software, such as the user interface or business logic (Murugesan, 2018). The data layer provides a centralised and consistent

way to access, manipulate, and store data, allowing different parts of the software to share and reuse data as needed (Davis, 2017).

The data layer can contain data manually provided by either a student or instructor, and data that is automatically generated as shown in Figure 3. A minimum viable architecture should minimally contain a class list as an essential data source. Additionally, for automated marking purposes, the rubric template would need to be provided so that a rubric can automatically be created and populated, per student, by the automated assistant. The student would also need to provide their assessment submission manually. Students may be prompted to provide feedback once the submission has been assessed. The feedback may be provided in different forms, with different purposes, for example, evaluating the impact of the automated assessment, Agile project retrospective feedback (what went well, what did not go well and what could be improved) and more.

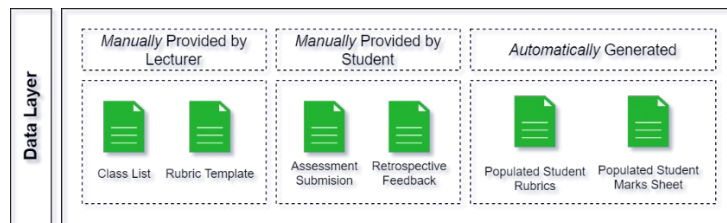


Fig. 3. Automated assistant data layer

Although Figure 3 shows the minimum datasets required, additional datasets can be included to enrich the process. Ideally, the student submissions could be stored separately from the class list and consistently stored, which may introduce more functionality into the integration layer.

Integration (Logic) Layer

The purpose of an integration layer in software architecture is to provide a unified interface for different components of an application to interact and exchange data with each other, as well as with external systems and services (Murugesan, 2018). The integration layer acts as a bridge between different parts of the software, enabling them to communicate and share data seamlessly and efficiently while providing a level of abstraction and decoupling that helps reduce complexity and improve maintainability (Davis, 2017).

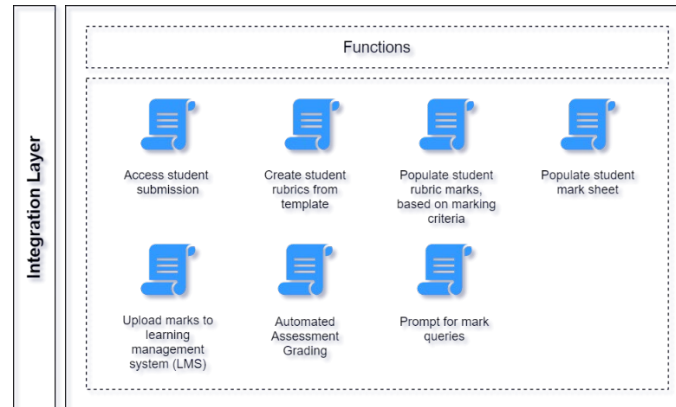


Fig. 4. Automated assistant integration layer

As part of the integration layer, multiple technologies and concepts may be introduced to develop the reusable functions listed in Figure 4. Technologies like Intelligent Automation (IA), Robotic Process Automation (RPA) and Digital Process Automation (DPA) may be used to automatically process assessment of tasks. For example, IA can be used to provide personalised feedback to students, DPA can be used to automate processes that span across multiple applications such as a plagiarism tool and a learning management system. As mentioned, RPA uses technology to mimic human behaviour and tasks, executing them the same way a person would. Therefore, RPA is useful and valuable for automating instructor-driven tasks.

Presentation Layer

The presentation layer in software architecture defines how data and functionality are presented to the user. This layer defines the user interface, including the look and feel layout and interaction mechanisms. It is responsible for rendering data and functionality in a way that is easy for the user to understand and use (Murugesan, 2018). The presentation layer is typically the front-end component of an application and interacts with the user through a graphical user interface (GUI) or a web interface (Davis, 2017) and differs between implementations. Dashboards may be introduced to summarise reports, but the essential metrics that would be useful to know as real-time as possible. Figure 5 provides the minimum set of reports that are generated by a variety of tools.

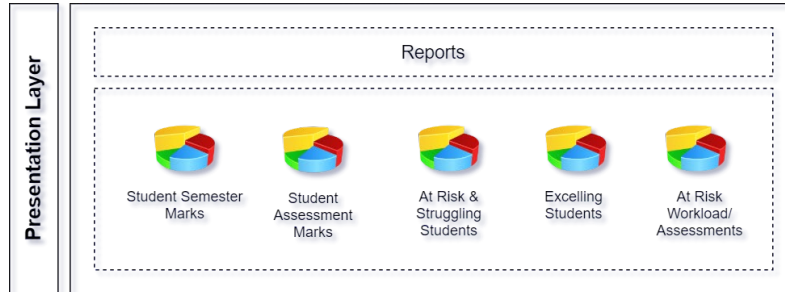


Fig. 5. Automated assistant presentation layer

Visualising and understanding the above metrics allows for data-driven decision-making and can assist lecturers in taking proactive steps to course-correct wherever necessary.

6 Implementation and discussion of results

A proof of concept (POC) was developed from the generic minimal viable architecture presented above to create a tool that aligned with the requirements addressed by the generic architecture. The generic architecture was applied to a use-case within an ICT project-based module, driven heavily by a partially attended automation approach as the first iteration of the design and build phases to realise efficiency and constraint improvement. The module is an exit-level module and consisted of approximately 270 students, led by two lecturers in different locations, covering the same five projects across multiple campuses.

6.1 Technical architecture and design

The POC was developed with a hybrid approach utilising the Google platform for all three layers while Microsoft (specifically Power BI) was used to further enrich the presentation layer. The architecture of the automated assistant POC is illustrated in Figure 6.

Most of the functions were developed in Python, through Google Colab and would be executed manually when needed. UiPath was used to execute front-end automation (commonly known as RPA) functions to deal with very specific assessments. The biggest hurdle instructors face is to familiarise themselves with the technology and create automated assessments using a variety of tools.

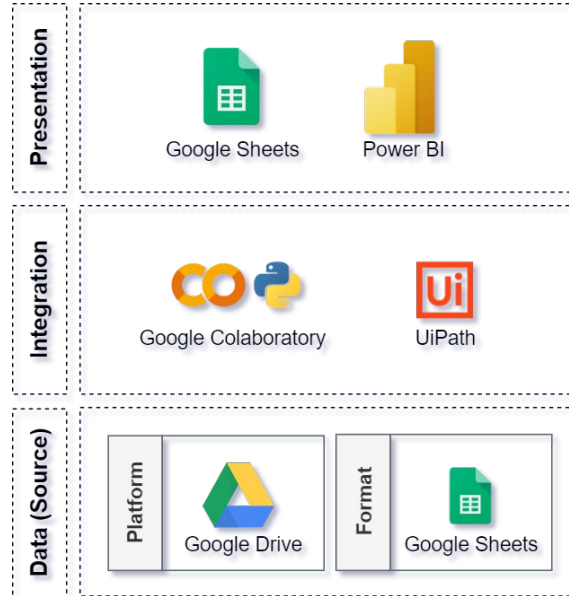


Fig. 6. Automated assistant POC architecture

6.2 Evaluation

The evaluation of automated solutions, specifically process automation solutions, can be measured in terms of time and cost, by comparing the metrics before the implementation of the process automation with the metrics post-implementation (Muller, 2022). Donmez et al. (2008) also propose a framework that takes similar criteria into account (as improved constraints) when evaluating the implementation of automation, with the addition of focused comprehensive understanding (domain knowledge), improved consistency (constructing validity), statistical efficiency and non-intrusivity (measurement of technical efficiency).

As seen in Table 1, the implementation of the automated assistant POC, across 5 projects, contributed to an overall time saving of 1 153 hours. The 1 153 hours saved takes into account the human effort of developing and moderating the automated assistant. The automated assistant POC serviced 1 280 assessments across 270 students, over five projects. The time saved was gathered through manual observation of executing the process manually compared to automated execution.

Table 1. Automated assistant POC time savings (in Hours)

Phase	Human Time (hours)	Execution Time (hours)	Development Time (hours)	Moderation Time (hours)	Total Hours Saved (per project)
Prep	62	3	3	0	65
P1	104	24	4	20	128
P2	232	24	4	20	256
P3	231	25	10	15	256
P4	364	20	12	8	384
P5	32	32	2	30	64
	1025	128	35	93	1 153

Based on the numbers provided above, the development time is assumed to decrease in the next iteration as very few changes need to be made (based on the success of the evaluation). Less moderation time will be needed in the next iteration as moderation was used to ensure the automated assistant was performing as required. As moderation and development time decrease, the human input into the loop decreases which would contribute to the exponential increase in total hours saved over the next few iterations before stabilising.

The cost saving could be calculated by taking the total hours saved and multiplying it by the per-hour rate of the employee who would be marking these assessments manually. For example, if the lecturer of a module cost the university R200.00 per hour, the university would be realising a saving of R230 600.00 by implementing the automated assistant. The cost and time saved can be focused on research (as an example) instead of admin-intensive tasks.

Table 2 is subject to the following evaluation criteria:

- **Improved constraints:** Has there been an improvement in time and/or cost constraints?
- **Focused domain knowledge:** Does each component of the solution have its distinct domain focus?
- **Improved consistency:** Does the retrospective solution provide improved consistency in output based on consistent input and processing business rules?
- **Improved efficiency:** Has there been an efficiency improvement witnessed? As the statistical sample set increases and the solution is scaled, does the efficiency exponentially increase?
- **Non-intrusivity:** Does each component of the solution maintain the non-intrusivity of students? Does the implementation have a negative impact on the students?

As complexity is introduced and changes are made, the scripts may still need to change and could impact the overall efficiency and time/cost saving. The higher the number of students, the more apparent the efficiency becomes and the more value it adds to be implemented. As it stands, this solution would not be useful for less than 20 students.

Table 2. Automated assistant POC evaluation criteria

Project	Improved Constraints	Focused Domain Knowledge	Improved Consistency	Improved Efficiency	Non-Intrusivity
P1	✓	✓	✓	✓	✓
P2	✓	✓	✓	✓	✓
P3	✓	✓	✓	✓	✓
P4	✓	✓	✓	✓	✓
P5	✓	✓	✓	✓	✓
PoE	✓	✓	✓	✓	✓

7 Future Work

The initiative summarised in this paper is part of an ongoing project that will continue to be improved. This research can be expanded to improve analytics and metrics that are being visualised. Including more metrics or reports would be a good way to dig deeper into existing data to extract new insights about student progress and success. Alerting can also be added to the solution so that lecturers are notified based on certain thresholds being obtained. This would be a useful mechanism to have as soon as a student who is struggling and is at risk of failing is identified. The design and the existing solution can be enhanced to include more audit logging which could focus on providing more detail in the rubrics to decrease project queries. Other architectures can be trialled, and results can be compared to this study to further evaluate the scope of benefit.

8 Conclusion

In conclusion, automated assistants for marking ICT student assignments have the potential to revolutionise the traditional method of assessing student work. The benefits of automated marking include reduced workload, consistent and objective evaluations of student work, and valuable insights into student performance. The development of a novel automated assistant architecture model and its evaluation demonstrated improved reliability, consistency, as well as reduced time and cost compared to manual assessment. The recommendations made in the paper will be valuable in successfully implementing automated assessment architectures in various educational settings, ultimately providing a more efficient and reliable way to assess student work without compromising the quality of feedback.

References

1. Ala-Mutka, K.M. A survey of automated assessment approaches for programming assignments. *Comput. Sci. Educ.* 2005, 15, 83–102
2. Awad, A. & Al-Rousan, T. 2015. Comparison between Microsoft Office and Google Docs. *International Journal of Scientific & Technology Research*, 4(5):4(5), pp. 149-152.
3. Axmann, B. and Harmoko, H., 2020, September. Robotic process automation: An overview and comparison to other technology in industry 4.0. In 2020 10th International Conference on Advanced Computer Information Technologies (ACIT) (pp. 559-562). IEEE.
4. Barra, E., López-Pernas, S., Alonso, Á., Sánchez-Rada, J.F., Gordillo, A. and Quemada, J., 2020. Automated assessment in programming courses: A case study during the COVID-19 era. *Sustainability*, 12(18), p.7451.
5. bin Mohd Yusoff, A.R., Vasilopoulou, M., Georgiadou, D.G., Palilis, L.C., Abate, A. & Nazeeruddin, M.K. 2021. Passivation and process engineering approaches of halide perovskite films for high efficiency and stability perovskite solar cells. *Energy & Environmental Science*, 14(5):2906-2953.
6. Chen, Y. & Chen, W. 2019. Automated Grading System for Programming Assignments. *Journal of Information and Education Technology*, 9(6):463-471.
7. Davis, J. 2017. *Software Architecture for Developers*. Leanpub.
8. del Pino, J.C.R.; Rubio-Royo, E.; Hernández-Figueroa, Z.J. A Virtual Programming Lab for Moodle with automatic assessment and anti-plagiarism features. In *Proceedings of the 2012 International Conference on e-Learning, e-Business, Enterprise Information Systems, & e-Government*, Las Vegas, NV, USA, 16–19 July 2012.
9. Dessoff, A. 2010. Google and Microsoft go to school. *The Education Digest*, 76(4):4.
10. Donmez, B., Pina, P.E. & Cummings, M.L. 2008. Evaluation criteria for human-automation performance metrics. In. *Proceedings of the 8th Workshop on Performance Metrics for Intelligent Systems* organised by. p. 77-82.
11. Education., G.F. 2021. *Empowering Students and Teachers with the Best Tools for Learning*.
12. Gordillo, A., 2019. Effect of an instructor-centred tool for automatic assessment of programming assignments on students' perceptions and performance. *Sustainability*, 11(20), p.5568.
13. Ghaemi, M. & Abolfazli, M. 2018. The Advantages of N-Tier Architecture in Developing Software Applications. *International Journal of Computer Science and Information Security*, 16(5):84-90.
14. Gomaa, H. 2017. *Software Architecture Design Patterns in Java*. Jones & Bartlett Learning.
15. Hazzan, O., Lapidot, T. and Ragonis, N., 2020. *Guide to teaching computer science*. Springer International Publishing.
16. Hevner, A.R. 2007. A three-cycle view of design science research. *Scandinavian Journal of information systems*, 19(2):4.
17. Hussain, M. & Zomayda, A.Y. 2017. *Handbook of Research on Scalable Computing Technologies*. IGI Global.
18. Lee, J., Kim, Y. & Kim, J. 2017. Automated Grading System for Object-Oriented Programming Assignments. *Journal of Educational Technology Development and Exchange*, 8(1):1-12.
19. Messer, M. 2022. Automated Grading and Feedback of Programming Assignments. Paper presented at the *Proceedings of the 27th ACM Conference on Innovation and Technology in Computer Science Education Vol. 2*
20. Mowbray, T. 2015. *Building Secure Software: How to Avoid Security Problems the Right Way*. Addison-Wesley Professional.

21. Muller, J. 2022. Evaluating time, scope and cost of human labour versus digital process automation capabilities. North-West University (South Africa).
22. Murugesan, S. 2018. Cloud Computing Principles and Paradigms. John Wiley & Sons.
23. Pirker, A. and Lechner, N.H., 2019. Designing Secure Architecture of Health Software using Agile Practices. In Central European Conference on Information and Intelligent Systems (pp. 269-280). Faculty of Organization and Informatics Varazdin.
24. Peffers, K., Tuunanen, T., Rothenberger, M.A. & Chatterjee, S. 2007. A design science research methodology for information systems research. *Journal of management information systems*, 24(3):45-77.
25. Ribeiro, Jorge, Rui Lima, Tiago Eckhardt, and Sara Paiva. "Robotic process automation and artificial intelligence in industry 4.0—a literature review." *Procedia Computer Science* 181 (2021): 51-58.
26. Ren, Y. & Zhang, Y. 2020. Reliability and Security: A Study on the Importance of Software Reliability in Enhancing Software Security. *Journal of Information Security*, 11(2):111-116.
27. Riehle, D. & Neuhäuser, H. 2017. *Handbook of Software Architecture*. Springer.
28. Simon, H.A. 2019. *The sciences of the artificial*: MIT Press.
29. Sobhy, D., Bahsoon, R., Minku, L. & Kazman, R. 2021. Evaluation of software architectures under uncertainty: a systematic literature review. *ACM Transactions on Software Engineering and Methodology (TOSEM)*, 30(4):1-50.
30. Team, PP 2020. *Power Platform Fundamentals*. Microsoft Press.
31. Wang, X. & Chen, W. 2019. The Google Ecosystem: A Study on the Interplay between Products, Services, and Platforms. *Journal of Business and Economics*, 10(7):1235-1244.
32. Whittaker, J. 2018. *Security in Software Systems*. Cambridge University Press.

Analysis of students' behaviour in Java programming class in blended learning environment using Process Mining techniques

Thusoyaone Joseph Moemi and Bukohwo Michael Esiefarienrhe

Department of Computer Science and Information Systems, North-West University,
South Africa

{Joseph.Moemi;Michael.Esiefarienrhe}@nwu.ac.za

Abstract. Some reasons attributed to students' poor performance in programming classes is the inability of the teachers to monitor students' activities in class and ascertain whether they are doing what they ought to do during programming hands-on exercises and during discussions of concepts and statements syntax and semantic. This research examines how students are monitored during classes in a year two Java programming module named CMPG 211 at the North-West University undergraduate laboratory for the first semester of 2022 academic session. The total number of students who logged into the Learning Management System (LMS) during class sessions are registered with timestamps. The methodology used in this research is the constructive science research as it is geared towards developing a solution towards an identified problem. During classes, students' participation in the various activities, namely: revision of previous lessons, teaching session, practical session, questions, and answers session, etc., were recorded LMS they used to log into the class. At the end of the lesson, these details are collected as event logs. The logs were analysed using Jupyter notebook 3.0. The total number of students who attended the class were 99, out of which only 72 students participated in all the activities in the class. 6 students only log in, disappear during the lessons, and returned towards the end to take part in the revision/summary of the lessons and did not participate in the Questions and Answers session. 21 students participated in all activates except Questions and Answers session. Therefore, the total students that avoided the questions and answers sessions out of the 99 students who attended the class were 27. The shows that students when not controlled, will likely miss Questions and assessment sessions. It also shows that some students just show up in class without participating in any activities. Their aim is simply to sign the attendance register and disappear. Using process mining techniques to manage the activities of students will create awareness among the students of the controls in place and hence stimulate to a greater extent their willingness to participate in all the class activities hereby increasing the pass rate for the course. It is recommended that teachers of technical and practical based courses should develop similar system to check the movement and participation of students in class activities.

Keywords: Blended learning and Process mining, Students behaviour in Java class, Process model in learning, Python in blended learning experience, student performance in Java.

1 Introduction

Process mining is an area of study in information mining. Process mining is an approach utilized in educational process mining (EPM), where we gather students' activity logs and use the logs to discover behavioural learning process information. To construct process models based on activity log data, a time stamp is applied to each activity, thereby producing event logs. Process mining combines machine learning, data mining, and process analysis and modelling [1], [2]. The approach encompassed in process mining can be utilized to build process models to identify, observe, and improve processes making use of the knowledge extracted from log files in various information systems. Process mining involves three fundamental steps: discovering the process, testing conformance, and enhancement. Process discovery is the initial phase where a process model is created in accordance with the event log file, but the knowledge used is inferred. Conformance testing is the second phase; it validates actual occurrences of events that correspond to the current real model and the reverse. Here, we parse the event log to create a new, enhanced model, this is the third procedure [2], [3].

EPM is a growing area of study where the process is central to the discovery, conformance investigation, and visualization of learning behavior of students [4], as opposed to simply deriving intriguing predictive outcomes from enormous amounts of academic data. EPM is an area in data mining which emphasizes discovering processes, analyzing conformance, and enhancing the entire educational process using information concealed in educational data sets and event records [2], [5]. Learning management system (LMS) is a form of e-learning-related software comprised of a variety of technologies that allow users to distribute content, execute and evaluation task to many students over a network and it also supports synchronous user communication [5], [6], [7]. The central element of a LMS is the event record, which contains process mining data on activities performed by students and instructors. Blended learning is a teaching approach used in educational institutions to maximize both technology, such as online and classroom (face to face) methods of learning [7]. In this research, blended learning was considered effective as it facilitates critical thinking by the students, allows teaching and learning without time boundaries using online streaming, notes, and online activities. Furthermore, through blended learning, students can practice without time limitations as against physical classes because of the online component in the learning method.

When beginner programmers learn how to write programs, they face challenges understanding how to think algorithmically, how to get problem solved, how to use a new integrated development environment (IDE), knowing programming patterns for the paradigm they are operating in, that is, structured programming (SP), object orientated programming, functional programming, etc [3], [8]. Students are required to handle areas in which they do not have a deep knowledge in some of the cases [8]. To become experienced and proficient in programming individually, students need to participate in class activities as much as possible as the LMS, lecturers and other student support systems (SSS) are always available to them [6], [9]. Teaching programming is however heterogeneous from the view of both computer software (CS) and available resources such as hardware, compilers, simulators etc. Many of challenging problems that

beginner programmers face is their inability to understand fundamental concepts needed to have a firm knowledge of the course, lack of adequate teaching and learning strategies that will facilitate the simplification of the language syntax and semantics, lack of adequate monitoring tools available to keep with students' progress and the use of obsolete course materials which are not benchmarked with industry. Our goal in this research is multidimensional. First, is to identify the concentration level of students regarding ongoing lectures and how they behave in class when classes are ongoing. The research also seeks to know how students follow up with various instructions given regarding the conduct of practical. Simply stated, this research intends to uncover how well do the learners follow activity as outlined at the beginning of the class in the outcome section of the topic for discussion and to also discover the behavior of the learners with regards to activities carried out in the programming course. To be able to track each of these activities, the research resorted to using research questions which were tied to various research objectives. Therefore, the following research questions guided the research namely: How do we capture student behavior in programming class? And how can process mining techniques be employed to reveal the pattern of students' behavior in programming courses? The two research questions were tied to the following research objectives namely: to use process mining techniques with the aid of event log to study students' behavior during Java programming classes and to develop a process model to unpack students' behavior regarding the various activities in a Java programming class respectively.

The remaining of the part of this article is organized as follows: Section 2 discusses some work done by other researchers related to this research. Section 3 discusses the research methodology used in this work and section 4 discusses the results obtained in this research. Section 5 presents the concluding remarks and the various recommendation from this work while section 6 gives the future direction of research in this domain of using process mining to understand students' behaviour in programming courses as a basis to improve their performance in practical based courses.

2 Brief Literature Review

Process mining is an interdisciplinary field that bridges the gap between information technology and procedural research. Thus, procedure models can be identified, conformance testing can be conducted, varying or obstructing conditions may be examined, and recommendations for improvements may be advised/advocated. To demonstrate the domain-focused applicability of unstructured event log data, Martin et al. [10], Bogarín et al. [11] and Esiefarienrhe and Ncube [12] implemented procedure mining in the fields of education, healthcare, and tuberculosis respectively. Event log data was utilized to develop a domain-particular model that includes an acceptable theoretical framework for developing added value service processes [10], as well as a comprehensive examination of learners online learning behaviour by conducting massive open online courses (MOOCs) [13] and a behavioural investigation of causes responsible for the high rate of students drop out of school [14].

Thiyagarajan and Prasanna [15] used flipped classroom as a teaching strategy, the authors evaluated pre-class activities using process mining on student's clickstream data which was generated through learning platform called "Delta". The students were categorized as credible or mediocre based their pre-assessment scores which were derived using fuzzy and heuristics miners. The various models derived from the application of process mining about the behaviour of the students were analysed. Our research mainly focused on understanding students' behaviour in a typical class activities using patterns.

Using a public API dataset for experiences, Hameed and Mrutyunjaya [16] seeks to understand and predict students learning behaviour using inductive visual miner (IvM) and directly follow visual miner (DFvM). The dataset contains data that track students' classroom activities and how they are involve in online communities and participation. The results from their process mining experiment shows how effective process mining models can be used to understanding how students behave during thre learning process. This work bears some similarity with our research because they both deals with understanding students learning behaviour using process miner although their work uses an existing dataset, ours use dataset that are directly gathered from an underlying software developed for the purpose collecting students' behaviour during classes.

The study conducted by Arpasat et al [17] used process mining and discovery models namely Dotted Chart Analysis, Fuzzy Miner, and Social Network Miner on the data extracted from open online courses. The process mining models were used to uncover the overall behavioral statistics of the students, to identify bottlenecks and loopback behavior through frequency-and time-performance-based approaches and lastly to analyze the working together relationships of the students.

In the context of online learning and assessment, Libor, Zounek and Rohlikova [18] used process mining techniques to explore students' behaviour and interaction patterns in various online quiz-based activities on a LMS. Using a process-oriented approach, they analysed students' interaction data embodied in the LMS and their results shows helpful behavioural patterns needed to understand the perspectives in leaning analytics and how EDM can be employed to understanding students interaction sequence.

The study by Cenka *et al.*, [19] used process mining techniques to conduct weekly assessment during a semester course of study of students learning behaviour in online platform using LMS. Their study revealed the following findings namely course materials, assignment and forum were the most frequently used online, LMS were mostly accessed on lecture days and students who used the LMS got the best grades whereas, students who do not make use of the LMS frequently got poor grades. The high grades students process model were more complex meaning there were more activities to model than lower grade students. The conclusion from the study shows that using systematic teaching strategy impact better on students' engagement and learning.

The authors of [20] attempted to use a process mining layer that utilized the Moodle LMS extracted data set. The primary objective is to locate students who are at the greatest risk of dropping out and who are likely to fail early on, as well as to substantially improve students' academic outcomes. For their experiments, the authors utilized a procedure mining software adapted from PM4Py modules, including event log processing and clustering finding. The importance of process mining was also demonstrated as a

tool to understanding the treatment pathways for tuberculosis in the healthcare sector as Esiefarienrhe and Ncube [12] in their research used process mining techniques to develop a software that keeps track of tuberculosis treatment pathways and the drugs for different regiment to generate a data log needed to assist health practitioners in the treatment and control of the disease. Their work brought to fore the importance of processing mining as a tool to understanding the details of what happened underneath the data. This research has a similar aim except that in this research, it is applied to understanding learners' behaviour and responses to the various class activities in a programming class for an object-oriented language called Java.

3 Research Methodology

Through concentrating on development processes, the objective of constructive research methodology is to enhance the functional performance of a system. Typically, categories of concepts such as algorithms, human or computer interactions, methodologies for design (including process models), and languages are the focus of constructive research. Applications of constructive research methodology are most prevalent in the fields of Engineering and Computer Science, although they are not limited to these fields and may be found across numerous disciplines [12]. We apply the following steps in our research taking consideration the chosen methodology.

- **Problem identification and sources of data:** Due to the difficult in teaching and learning a programming language such as Java, mastering its syntax and semantics, and the limitation in the skills of problem solving among students particularly as most of them are mathematically weak coupled with various peer influence among learners which most times distract them from concentration, there is need to develop a system that can monitor the activities of learners in class and how they respond to the various exercises given particularly programming assignment. One way this study partly overcame the above issues was to adopt and implement blended learning method in the course delivery. This enables us to use both contact and online teaching strategies and also give and monitor assignments compliance online and class. At the end of this phase, we have clearly understood the problems with learners and have selected the strategies to solve the identified problems. One strategy is to use blended learning for the course delivery and the second is to implement a process mining method using heuristics miners to generate a process model that can enable us to understand the behavioral patterns of the learners during class activities. Armed with these strategies, we decided to proceed to the next phase in the methodology.
- **Design/development and Data collection procedure:** This phase entails the design and development of interventions or strategies aimed at improving the monitoring of students' activities in the blended learning environment. It encompasses the process of creating and implementing systems, processes, or tools that enable effective monitoring of students' engagement and participation during programming activities. This phase may involve designing monitoring protocols, creating technology-based solutions, or developing guidelines for lecturers to follow. In this study, combined the

university LMS called efundi and a tracking software developed for the purpose of collecting data as log files from the various activities the students are engaged with.

The data used in this study were partly collected from the log activities of students' interaction with the university LMS (efundi) specifically the log in and log out time for each student and the workstation ID. As LMS (efundi) was not developed for process mining applications, we had to develop a system using Java that captures the log of activities conducted in class. The various activities labeled "Activity" in figure 1 constitute the milestone used in the assessment of the students' behaviour in response to how they participated in class. The case ID, activity, start date and time, end date and time, PC ID, personal laptop ID, and activity agent as shown in the table 1 are saved into a database file called log trace for every student during contact class session and during online session. The response to the activities is automatically captured by the system as soon as a student partake and/or complete these activities. At the end of the class, the lecturers approve the log file and generate a trace copy with approval timestamp signifying a backup process for the data collected. This is done for every class conducted throughout the semester.

- **Demonstration:** Discovering a process model from event log which contains the response of students to the various activities in class, a Heuristic miner model was used to analysed the dataset for any causal dependencies. To analyze these relations, the authors in [21] introduced the following notations.

Let W be an event log over T , i.e., $W \subseteq T^*$. Let $a, b \in T$:

1. $a >_W b$ iff there is a trace $\sigma = t_1 t_2 t_3 \dots t_n$ and $i \in \{1, \dots, n-1\}$ such that $\sigma \in W$ and $t_i = a$ and $t_{i+1} = b$,
2. $a \rightarrow_W b$ iff $a >_W b$ and $b \not>_W a$,
3. $a \#_W b$ iff $a \not>_W b$ and $b \not>_W a$, and
4. $a \parallel_W b$ iff $a >_W b$ and $b >_W a$,
5. $a >>_W b$ iff there is a trace $\sigma = t_1 t_2 t_3 \dots t_n$ and $i \in \{1, \dots, n-2\}$ such that $\sigma \in W$ and $t_i = a$ and $t_{i+1} = b$ and $t_{i+2} = a$,
6. $a >>>_W b$ iff there is a trace $\sigma = t_1 t_2 t_3 \dots t_n$ and $i < j$ and $i, j \in \{1, \dots, n\}$ such that $\sigma \in W$ and $t_i = a$ and $t_j = b$

Using the Heuristic miner algorithm in Disco process mining software, which was developed and licensed by Fluxicon, we were able to generate appropriate process model that enable us to analyze the dataset resulting in the graph in Fig.1

- Evaluation

The evaluation method is used to assess the outcomes and effectiveness of the implemented interventions or strategies. This phase involves collecting data, analyzing results, and evaluating the impact of the monitoring methods on student performance and engagement. It may include gathering quantitative data, such as attendance rates and participation levels, as well as qualitative data through process mining techniques to gather students' perspectives on the monitoring approach.

- Findings

The findings method involves the analysis and interpretation of the collected data from the evaluation phase. We examine the data to identify patterns, trends, and correlations related to student engagement and performance. This method aims to derive meaningful insights and draw conclusions based on the data analysis, highlighting the effectiveness or limitations of the monitoring interventions in addressing the identified problem in blended learning.

- Reflection

The reflection method involves a critical examination and thoughtful consideration of the research process, outcomes, and implications. We engage in self-reflection and assess the strengths and weaknesses of the implemented monitoring strategies. We analyze the overall research process, including the design, execution, and evaluation, to identify areas for improvement and potential future research directions. Reflection allows for a deeper understanding of the research findings and their significance within the broader context of blended learning.

4 Results and Discussion

4.1 Creation of the event log

As soon as class begins, each student logs into the LMS which is a learner's trace system that keeps track of students' interaction with respect to the various activity and record the timestamps at which the events occurred within the system, producing a log file for the day's class activities. The student's identity is recorded by the system and the following information, PC ID, personal laptop ID, and the time the student log in and log out of each activity are recorded. These activities labeled "Activity" in figure 1 constitute the milestone used in the assessment of the students' behaviour in response to how they participate in class. The case ID, activity, start date and time, end date and time, PC ID, personal laptop ID, and activity agent as shown in the table 1 are saved into a database file called log trace. Case ID is a unique identifier for each student.

Activity is information about each task that took place during each class. Start date and time as well as End date and time are the start time and end time for each task that took place in class. PC ID is a unique identifier for the personal computer available in the lab. Personal laptop ID is a unique identifier for personal laptops that students bring along to class. Activity agent is an identifier that is used to indicate the person that will be championing a specific task during class. From the event log (trace file) collected, we created the process model by installing pm4py and running the heuristics miner on the log and the result is shown in fig 1. The class started with 99 while 6 students came late to class and only participated at revision ending. Therefore, 87 students and the 6 students that came late to class continued to revision ending. From the 93 students, 72 students continued to the question and answer session and proceeded to logging out of the system. 21 students logged out of the system after the revision ending activity and 6 students logged out of the system after the practical demonstration.

Table 1. Event logs from the LMS developed for trace logging of activity

	CaseID	Activity	StartDate	EndDate	PCID	PersonalLaptopID	ActivityAgent
0	Case 1	Time to Settle Down	28.2.22 8:00	28.2.22 8:10	PC 1	NaN	Lecturer
1	Case 1	Revision of Previous Lecturer	28.2.22 8:10	28.2.22 8:25	PC 1	NaN	Lecturer
2	Case 1	Teach	28.2.22 8:25	28.2.22 9:30	PC 1	NaN	Lecturer
3	Case 1	Practical Demo/ Class Hands on	28.2.22 9:30	28.2.22 10:30	PC 1	NaN	Lecturer
4	Case 1	Revision Ending	28.2.22 10:30	28.2.22 10:55	PC 1	NaN	Lecturer
...	...	...	...	...	...	...	...
532	Case 93	Revision of Previous Lecturer	28.2.22 8:10	28.2.22 8:25	PC 2	NaN	Lecturer
533	Case 93	Teach	28.2.22 8:25	28.2.22 9:30	PC 2	NaN	Lecturer
534	Case 93	Practical Demo/ Class Hands on	28.2.22 9:30	28.2.22 10:30	PC 2	NaN	Lecturer
535	Case 93	Revision Ending	28.2.22 10:30	28.2.22 10:55	PC 2	NaN	Lecturer
536	Case 93	Q and A (practical and theory)	28.2.22 10:45	28.2.22 11:00	PC 2	NaN	Lecturer

537 rows × 7 columns

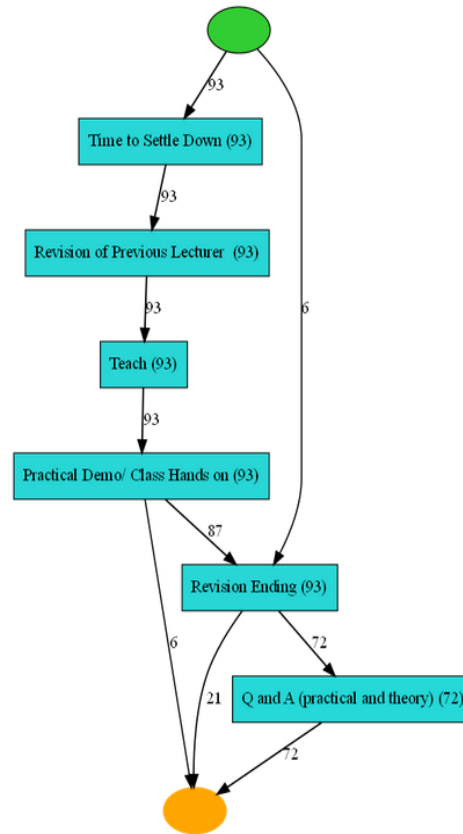


Fig. 1. Process model

The findings presented in this research shows how students can cleverly (unknown to the lecturer avoid participating in major class activities and such behaviour can contribute to the high rate of failure and lack of skills to develop and master programming. This work will contribute to the ongoing discussion on the factors influencing students' poor performance in programming classes. One significant factor identified is the lack of effective monitoring by lecturers, which prevents them from ascertaining whether students are actively engaged during programming hands-on exercises and discussions on concepts, syntax, and semantics.

By examining the monitoring practices in the year two Java programming module, CMPG 211, at the North-West University undergraduate laboratory, the study provides valuable insights into the extent of student participation and the potential consequences of inadequate monitoring. The use of the Learning Management System (LMS) to register students' logins and record their activities during class sessions allowed for a detailed analysis of student engagement.

The results revealed that out of the 99 students who attended the class, only 72 students consistently participated in all the activities throughout the class. Additionally, a concerning pattern emerged, with some students logging in but then disappearing during the lessons, only to return for the revision/summary portion of the class and opting out of the crucial question-and-answer sessions. Furthermore, a substantial number of students completely avoided the question-and-answer sessions altogether.

These findings raise several important points for discussion. Firstly, they underscore the significance of monitoring students' activities in programming classes. Without proper monitoring, students are more likely to miss important assessment and interactive sessions, which can negatively impact their understanding and overall performance. This highlights the need for teachers to actively monitor student engagement and take appropriate measures to ensure their participation in all aspects of the class.

Moreover, the results reveal a subset of students who merely attend class to fulfill attendance requirements, demonstrating a lack of intrinsic motivation to actively participate in the learning activities. This calls for further exploration into the underlying reasons behind this behavior and the potential strategies to address it. It is crucial for educators to create an environment that fosters students' intrinsic motivation, encouraging them to genuinely engage in the learning process rather than passively fulfilling formalities. The result from the work is significant to the development of instructional methods and support structures because it will assist educational planners to realize the importance of participatory learning and pay less emphasis on theoretical concepts for courses that are practical based. Instructional methods should evolve which supports peer-to-peer support and human-computer interactions so that the system can automatically provide feedback for each module outcome based on the student interaction and involvement with the topic and exercises.

To address these issues of students avoiding class activities, the study suggests the use of process mining techniques to manage student activities. Implementing such techniques can help create awareness among students regarding the monitoring measures in place, potentially increasing their willingness to participate and contribute to all class activities. By effectively managing student engagement, the pass rate for technical and practical-based courses, such as programming, can be enhanced.

5 Conclusion

In conclusion, this research has demonstrated the potential of process mining techniques for analyzing programming student behavior in blended learning environments. By identifying patterns and factors that influence learning outcomes, lecturers can make data-driven decisions to optimize their teaching strategies and support systems, ultimately improving the quality of programming education. The study specifically examined the monitoring practices in a year two Java programming module, CMPG 211, at the North-West University undergraduate laboratory during the first semester of the 2022 academic session.

The findings clearly highlight that the inability of lecturers to effectively monitor students during programming hands-on exercises and discussions of concepts, syntax,

and semantics contributes to students' poor performance. By analyzing the data collected through the software system, which recorded students' logins and participation in various class activities, valuable insights were obtained.

Out of the 99 students who attended the class, only 72 students actively participated in all the activities. Additionally, some students exhibited patterns of disengagement, logging in but disappearing during the lessons and only returning for the revision/summary of lessons without participating in the crucial question-and-answer sessions. Furthermore, a significant number of students avoided the question-and-answer sessions altogether.

These findings emphasize the importance of implementing effective monitoring strategies to ensure student engagement and participation. By using process mining techniques and creating awareness among students regarding the monitoring controls in place, their motivation and willingness to actively participate in all class activities can be significantly enhanced. This, in turn, has the potential to improve the pass rate for the course.

The research has been able to answer the research questions and objectives set out in section 1. The research objectives were stated as: to use process mining techniques with the aid of event log to study students' behavior during Java programming classes and to develop a process model to unpack students' behavior regarding the various activities in a Java programming class respectively. The first objective was achieved by identifying the various activities that can be used to monitor students behaviour in a Java programming class. These activities were incorporated in the log software that records the activities students engaged with in class as can be found in section 4.1. The second objective saw the creation of process model to analyse the students' behaviour using the heuristic miner algorithm in section 3 and the model shown in fig 1. With these results, the objectives of the study has been achieved.

Based on the outcomes of this research, it is recommended that lecturers of technical and practical-based courses consider developing similar monitoring systems to track student movement and participation during class activities. By implementing such systems, lecturers can better identify and address disengagement issues, ultimately fostering a more interactive and productive learning environment.

This study underscores the importance of monitoring students' activities in programming classes and provides valuable insights for educators and institutions to enhance student performance and engagement in technical and practical courses.

One of the obvious shortcomings in this work is the limited size of dataset that was used in the mining process as the data was collected from a semester course of study. We do strongly believe that with more data available, the behaviour patterns of students' as seen from the data may change and this may add more insight into how students' behaviour in class with regards to learning activity can be managed. We also hope that the tracking software can be fully incorporated into the university LMS so that more activity can be tracked about different aspects of the students' behaviour and gender as well as students background can be added for a more robust system.

6 Future work

Future research will continue to explore the applications of process mining in educational contexts and develop innovative methods to enhance student learning experiences taking into consideration their diversities and environmental factors. We will want to discover how gender, family background, peer groups, associations and affiliations affect students' behaviour in class with regards to learning and knowledge retention in future work. The development of a comprehensive databases with timestamps for process conformity checks, except handling and student performance analysis will be the goal of this research in future. Also, a standard group and control group consisting of students with the same characteristics will be created so that the effect of the application of the process mining to students' performance can be measured and analyzed.

References

1. AlQaheri, H., Panda M.: An Education Process Mining Framework: Unveiling Meaningful Information for Understanding Students' Learning Behavior and Improving Teaching Quality. *Information* 13(1), 29 (2022).
2. Andreswari R., Fauzi R., Valensia L., Chanifah S.: Conformance Analysis of Student Activities to Evaluate Implementation of Outcome-Based Education in Early of Pandemic using Process Mining. *SHS Web of Conferences* 139, 03018 (2022).
3. Theptudborvornun C., Narksarp W., Porouhan P., Arpasat P., Intarasema S., Premchaiswadi W.: Analysis of Learners' Participative Behavior from Active Learning Management by Process Mining Technique. In: 2020 18th International Conference on ICT and Knowledge Engineering (ICT&KE), pp. 1–4. Rajamangala Univ of Technology, Srivijaya (2020).
4. Onur, D., Antonio, M., Eric, R., Marcos, S., Jorge, M., Vicente, T. Carlos, F.: Individual Behavior Modeling with Sensors Using Process Mining. *Electronics* 8(7), 766 (2019).
5. Real, E.M., Pinheiro P.E., de-Oliveira, L., Cristina, BV., Stiubiener, I.: Educational Process Mining for Verifying Student Learning Paths in an Introductory Programming Course. In: 2020 IEEE Frontiers in Education Conference (FIE), pp. 1–9. Uppsala, Sweden (2020).
6. Bogarín, A., Cerezo, R., Romero, C.: Discovering learning processes using Inductive Miner: A case study with Learning Management Systems (LMSs). *Psicothema* 30(3), 322–329 (2018).
7. Nafasa P., Waspada I., Bahtiar N., Wibowo, A.: Implementation of Alpha Miner Algorithm in Process Mining Application Development for Online Learning Activities Based on MOODLE Event Log Data. In: 2019 3rd International Conference on Informatics and Computational Sciences (ICICoS), pp. 1–6. Semarang, Indonesia (2019).
8. Cerezo, R., Bogarín, A., Esteban, M., Romero, C.: Process mining for self-regulated learning assessment in e-learning. *J. Comput. High. Educ.* 32(1), 74–88 (2020).
9. Balogh, Z., Kuchárik, M.: Predicting Student Grades Based on Their Usage of LMS Moodle Using Petri Nets. *Appl. Sci.* 9(20), (2019).
10. Martin, N., De Weerd, J., Fernandez-Llatas, C., Gal, A., Gatta, R., Ibanez, G., Johnson, O., Mannhardt, F., Marco-Ruiz, L., Mertens, S., Munoz-Gama, J., Seoane, F., Vanthienen, J., Wynn, M.T., Boileve, D.B., Bergs, J., Joosten-Melis, M., Schreten, S., Van-Acker, B.:

- Recommendations for enhancing the usability and understandability of process mining in healthcare: *Artif. Intell. Med.* 109, 101962 (2020).
11. Bogarín, A., Cerezo, R., Romero, C.: A survey on educational process mining. *WIREs Data Min. Knowl. Discovery* 8(1), e1230 (2018) doi: 10.1002/widm.1230.
 12. Esiefarienrhe, B.M., Ncube, A.: Healthcare information System for tuberculosis and the creation of event log for process mining. *Indian Journal of Computer Science and Engineering* 14(2), (2023)
 13. Maldonado-Mahauad, J.M., Pérez-Sanagustín, R.F., Morales, N., Munoz-Gama, J.: Mining theory-based patterns from Big data: Identifying self-regulated learning strategies in Massive Open Online Courses. *Comput. Hum. Behav.* 80, 179–196 (2018).
 14. Salazar-Fernandez, J.P., Sepúlveda, M., Munoz-Gama, J.: Influence of Student Diversity on Educational Trajectories in Engineering High-Failure Rate Courses that Lead to Late Dropout. In: *IEEE Global Engineering Education Conference (EDUCON)*, pp. 607–616. Dubai, United Arab Emirates (2019).
 15. Thiagarajan, G., Prasanna, S.: A Process Mining approach to analyze learning behavior in the flipped classroom. In: *2nd International Conference on Communication, Computing, and Industry 4.0 (C2I4)*, pp. 1-7. Bangalore, India (2021).
 16. Hameed, A., Mrutyunjaya, P.: An Education Process Mining Framework: Unveiling Meaningful Information for Understanding Students' Learning Behavior and Improving Teaching Quality. *Information* 13(1), 1-19 (2022).
 17. Arpasat, P., Premchaiswadi, N., Premchaiswadi, P., Wichian, P.: Applying Process Mining to Analyze the Behavior of Learners in Online Courses. *International Journal of Information and Education Technology* 11(10), 436-443 (2021).
 18. Libor, J., Zounek, J., Rohlikova, L.: Using process mining to analyze students' quiz-taking behavior patterns in a learning management system. *Computer in Human Behavior* 92, 496-506 (2019).
 19. Censka, B.A.N., Santoso, H.B., Junus, K.: Analysing Student Behaviour in a Learning Management System Using a Process Mining Approach. *Knowledge Management & E-Learning* 14(1), 62-80 (2022).
 20. Hachichaa, W., Ghorbel, L., Champagnat, R., Zayani, C.A., Ikram A.: Using Process Mining for Learning Resource Recommendation: A Moodle Case Study, *ScienceDirect*, <https://www.sciencedirect.com/science/article/pii/S1877050921015763>, last 2022/09/15.
 21. Weijters, A.J.M.M., van der Aalst, W.M.P., Alves de Medeiros, A.K.: *Process Mining with the HeuristicsMiner Algorithm*. 1st edn. Technische Universiteit Eindhoven, Eindhoven (2006)

Blended Agile Learning of Computer Architecture under COVID

P Machanick^[0000-0001-6648-7032]

Rhodes University, Makhanda, South Africa
p.machanick@ru.ac.za

Abstract. The COVID-19 pandemic presented unique challenges. 2021 was an interesting year because we had overcome the worst of the teething problems of remote learning but were able to resume some in-person activities. I present experiences from a second-year computer architecture course that I have taught since 2014 to illustrate that some lessons from operating under pandemic conditions can apply to running courses under more normal conditions. 2021 was an interesting year because we reintroduced in-person pracs partway through this course, allowing students to reflect on the difference this made. Reflection on what did and did not work in the course points to possible improvements in pedagogy in more “normal” times. In isolation, the very positive feedback in a course survey may be flattering but there are useful insights to be drawn from what worked. Drawing on ideas from the social construction model of education, where students should be actively involved in learning, and agile software development, results in some insights that may generalize. The kind of feedback that is part of agile development can be layered on top of formative assessment. Empathy with difficulties faced by a class can make a class more involved in the strategy for course delivery. In 2022, without COVID constraints, some of the lessons were applied with positive outcomes.

Keywords: COVID-19, Computer Architecture, blended agile learning.

1 Introduction

I have taught a second-year computer architecture course since 2014, using my own book-length notes (the book is self-published and available on Amazon [10]). Given lengthy experience with this course, it presented a useful opportunity to research the effect of pandemic interventions particularly those that pointed to post-pandemic improvements in pedagogy.

Two factors made for relatively easy adaptation to COVID conditions: it is my own material so I can adapt it easily and I had no need to concern myself about criteria for career advancement, so I could neglect research and other factors unrelated to the course. Consequently, I was able to try a range of strategies without concern for the cost in extra time.

What made it challenging to structure presenting the course as a research exercise was the changing environment. For this reason, I do a retrospective analysis of factors

that influenced outcomes and build a model of what happened after the event. Usually, a research project starts from a research question or hypothesis to test, a research method that fits the problem and a literature review to ground the project in known art. In this case, a retrospective study works the other way around. I study practices developed during the course as a way to define a research approach that elucidates lessons for future courses.

The 2021 run of the course built on lessons from 2020, where students were sent home just before the course started. 2020 was a disaster for several reasons. Notes were printed for students and no one thought to advise them to pick them up before they left. Rhodes University had a big increase in NSFAS students, with about 70% of new science students the year before being in this category, and assuming everyone could go home and use Zoom was far from the reality of that category of student.

Once the students were sent home, plans were made to purchase laptops for NSFAS students out of their textbook allowance and free data was negotiated with cell networks for campus IP addresses. Students were also given a monthly data allowance. It took a good fraction of the 4 weeks allocated to the course to work through all these issues so the 2020 instance of the course had a smaller practical component.

One of the problems in 2020 was that the “free” access to our own sites turned out to apply only to TCP/IP traffic and live video generally – and specifically the Big Blue Button (BBB) system – uses UDP [5, 17]. Replaying videos, on the other hand, did use TCP/IP and hence was free. We had initially chosen BBB specifically as we could host it on our own system and hence put it into the range of IP addresses included in free traffic. Consequently, when students were off campus and had limited data, using BBB was only really useful for reviewing recorded question and answer sessions.

In 2021, these problems were mitigated as those students who needed lab facilities, including Computer Science, were permitted back on campus though at first not with full in-person contact. Classes where a sufficiently large venue was not available to accommodate 50% occupancy were run as alternating in-person and online sessions: half of the class at a time was allowed into the lecture, and the rest could follow on BBB. BBB in this situation worked better because students were mostly on campus, so they did not need free data. Halfway through the course, in-person pracs were allowed so we could transition to this in the last two out of four weeks.

The course ran for four weeks from 3 May 2021 to 28 May 2021. That presented interesting challenges as the course ran during a dip in COVID infections that took off again with the Delta variant [18] before the June exam session. On 1 March, the country went down to alert level 1, On 31 May, alert level went up to 2 then level 4 on 16 June. It went down to 4 on 16 June and up again to level 4 on 28 June, then down to level 3 26 July–12 September [16].

Fig. 1 (adapted from [18]) illustrates how timing of the Delta variant varied across the country over March–September 2021.

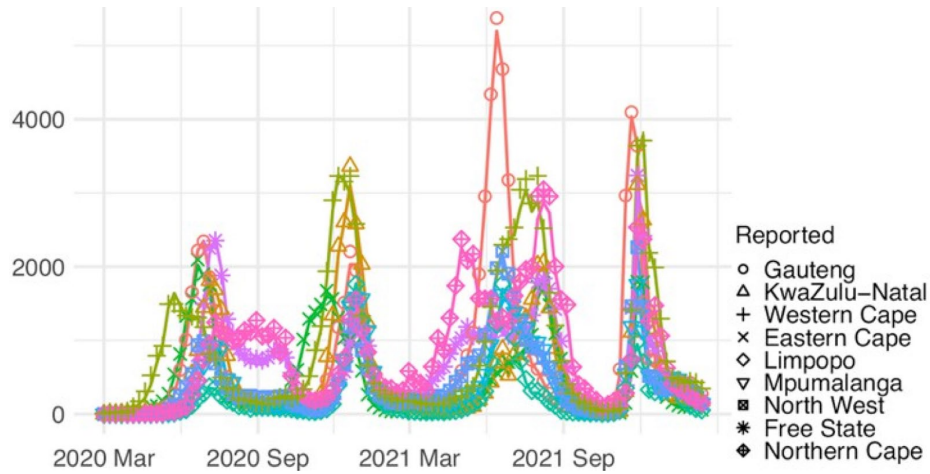


Fig. 1. Timing of COVID cases per 100,000 population in 2021.

Blended learning captures the essence of the approach used. However, since the underlying reality kept changing, I could not plan the whole thing out in advance. Instead, I constantly reviewed the approach as events unfolded. In effect what developed was a kind of agile blended learning.

Although my focus was on the class not on doing research, I did obtain ethics clearance for a survey for the 2021 run of the course, in anticipation of being able to share lessons.

Outside the pandemic scenario, an agile blended learning approach would be useful to develop further so I outline some of the lessons arising from problems and resulting solutions.

2 Applicable Theories and Models

I start from reviewing educational theory and how it can apply in a situation of a diverse class with a rapidly changing learning context. From this start, I review agile methods in software development for insights into how an adaptive educational strategy can be developed.

The social construction model of education turns out to be a good fit to the core ideas of agile software development.

2.1 Educational Background

There are many theoretical approaches to education. One of the earliest applied to Computer Science is Piaget's theory of learning stages starting from the sensory-motor preverbal stage and ending with formal operational stage, where children learn to form abstract concepts and to think logically and form plans [13], Though Piaget's work focused on children, with the last stage starting at about age 12, it has become

the basis for the constructivist approach to education in Computer Science, which emphasizes building mental models [3].

Constructivism at heart assumes that the main learning task is constructing a mental model but does not take into account the context of learning. Social constructivism does take context into account and considers interactions between learners and educators [2].

In a highly diverse class, particularly one where the educator is from a very different background than many members of the class, context and communication matters.

To go a step further, the social construction model starts from social interactions and assumes that all knowledge is created by interpersonal interaction [9]. Very little work in Computer Science education has built on this idea; one study has explored the role of dialog in bridging socio-cultural gaps between learner and teacher [15]. The pedagogic approach in this study, PRIMM is based on a feedback-based method with the following steps:

- *Predict* what given code does
- *Run* it to test the predictions
- *Investigate* how the given code is structured
- *Modify* the given code
- *Make* a new program based on the learned structures

In the context of a COVID-constrained class, particularly one with very different demographics to the last in-person class, how can any of these ideas apply?

An educational philosophy has to be operationalised: how is it put into practice? The PRIMM approach is one example. In the COVID context, where differences in ability to access technology and the playing field altered as regulations relaxed, a multimodal approach was needed.

Blended learning, with a mix of modalities and technologies, is one approach that has seen increasing favour, particularly as mixed-mode learning has been forced by the COVID pandemic [8]. Some say that the name is inaccurate as it is a teaching rather than a learning strategy [12]. This may seem to be a distinction without a difference since whether an approach is seen as a teaching or learning strategy is simply a matter of the orientation of the observer. However, a better way of looking at it is to see blended learning as a way to operationalise a given education model.

The simplest teaching technology, chalk and talk, does not imply a specific educational model. A lecturer who mumbles incoherently whilst illegibly scrawling on a board clearly has a different model of pedagogy in mind than one who challenges a class to answer hard questions and writes up a summary.

Without entering into a debate as to whether blended learning is misnamed, it can be taken as the technology adopted; how to make best use of it is especially difficult in the COVID context as reality kept shifting. The approach I adopt for this course is rapid informal review of how well the strategy is working, with micro-adjustments based on observing how the class is doing. Specific details of this approach are not generalisable as the conditions are not repeatable. However, the general idea is similar to that of agile development with rapid review and feedback cycles. There have been

studies of how well agile development held up over COVID conditions including lockdowns; unsurprisingly, teams that were co-located had the worst negative effects [14]. However another study shows that these effects could be mitigated in some cases producing better outcomes, resulting in support for a future blended approach, in which co-location was only required if really necessary [11].

Putting all these ideas together, a COVID-constrained class has to take into account rapidly changing circumstances, student diversity that may be harder to gauge from a distance than with personal interactions and rapid reviews of the effectiveness of any approach are useful.

All of this is easy to propose as a basis for an educational strategy after the event but, during the class, since unexpected developments kept arising, rapid reviews were reactive, rather than planned. It is also important to keep in mind that students are neither customers nor pedagogy experts so feedback cycles have to be based on this reality.

However, lessons from the class suggest that a feedback-based approach, generalised from PRIMM, would be worth using even outside of the pandemic context.

2.2 Fitting Agile Development to Pedagogy

Agile development is a well-established approach to software projects. Pedagogy is not the same thing as developing software so not all ideas will apply. Nonetheless agile software development (ASD) has proved to be a useful approach for adaptive software development, avoiding locking in decisions that are hard to undo at a later stage.

A recent analysis of the theoretical core of ASD identifies the essential idea as the ability to “anticipate, create, learn from and respond to changes in user requirements through a process of continual readiness” [1]. That is a very generic description that can apply to any scenario where change is a factor. This description is further amplified as being achieved by:

- incremental design with iterative development
- cycles of inspection and adaptation
- continuous involvement of the customer
- collaborative and cooperative work with close communication

This generic framework can fit a course situation well, if there is a need to adapt rapidly to changing circumstances. The biggest difference is the relationship between the teacher and student, which is not that of developer and customer. It is also important to remember that students have many demands on their time, so review and feedback should not take a form that puts extra demands on them.

Tactics I used to approximate to rapid feedback cycles included putting mini-quizzes on our online Moodle platform, which helped me to assess how well the class was keeping up. I did not advertise them for this purpose but rather made them available for the class as formative assessment. Another tactic was frequent discussion with tutors on how hand ins were going, and whether any of their group would benefit

from extending a deadline. I also kept in close communication with the class through their WhatsApp group.

None of these interventions exactly fit the agile model but are in the spirit of rapid feedback and adjusting the strategy to fit observed reality.

The approach of rapid feedback cycles also draws on some of the ideas of social construction: making the class feel that they are part of the process of shaping assessments brings them closer to the community of practice that they aspire to join.

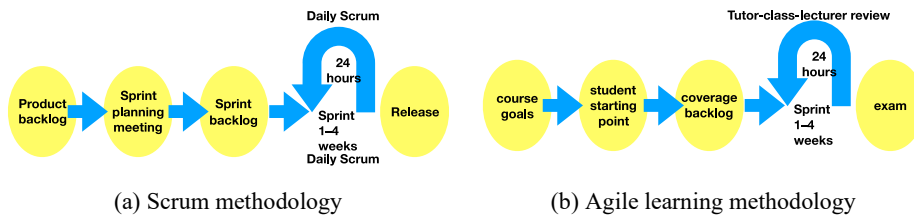


Fig. 2. Agile software development vs. education.

Fig. 2 shows how educational practice can be mapped onto the Scrum approach, one of the more popular approaches to Agile development [7]. A key difference is that education is not about creating an artifact but about changing the learner’s abilities. Hence, there is not as clear an endpoint so I replace that by the exam.

To dig deeper into agile development than this would not be a good fit to pedagogy as we are not trying to create a software artifact.

2.3 Putting it all Together

Returning to the idea that knowledge is socially constructed, the teacher in this situation, while partially in the role of the “developer” in ASD terms, is also trying to bring the student to their level of understanding. Feedback cycles involving the “customer” – in this case, the students – is a good fit to social construction.

This is a useful insight not just for the changing ground of a pandemic but also for pedagogy in general. Treating the class as an integral part of feedback cycles rather than as consumers of knowledge is consistent with the social construction model. However, a key difference from ASD is that students should be developing their understanding to be closer to that of the lecturer, rather than retaining a distinct role (the customer or end user is not trying to become a developer).

3 Differences in the course across years

The course was last run without COVID conditions in 2019; in 2020, it was run right after students were sent home. In 2021, lessons of 2020 could be applied even as restrictions reduced while in 2022 restrictions were mostly eased.

A key differentiator of 2019 as well was that it was the last year before changes in the terms of the National Student Financial Aid Scheme (NSFAS) resulted in a big majority of the students (about 70%) at Rhodes University being on NSFAS and hence from relatively poor households.

In 2019, as with previous years, the course was run over 4 weeks with 5 lectures and one prac per week, totalling 4 pracs. Each prac consisted of shorter questions to hand in before the end of the session and one or more to be completed no later than the day before the next prac. I generally did not give extensions as doing so impacts on other courses and time to do the next prac, as well as delaying publishing solutions.

In 2020, everything was set up to run the course in the normal way, including printing lecture notes for students. When the university decided to send all the students home, there was very little time to react and as a result the class went home without their notes. The initial proposal was to run classes on Zoom but the reality of NSFAS students not going home to fast Internet or a home computer sank in and there was a scramble to organize free data and to buy computers out of the NSFAS book allowance. Logistics of this took up much of the time for the course, so I adjusted the pracs to be possible to do without a computer. For the reasons noted in the Introduction, I made short videos to substitute for the lectures, aiming to make each no more than 25 minutes. The idea behind this: a longer video is harder to watch without losing concentration. If a student viewing a shorter video finds something hard to follow, it is reasonably easy to backtrack to play it again. Each video was about as much material as I would put into a 45 minute lecture; the option to replay made it reasonable to explain relatively rapidly.

Since streaming video was not available as part of the free data allocation, I ran Q&A sessions on BBB and recorded them, so those who relied on free data (particularly in 2020 when all the students were at home) could view them later.

As an indication of trust, in all three years, the class included me in their own WhatsApp group.

I adjusted to fit the scenario where students with poor connectivity struggled to keep up by being more flexible on deadlines and taking the best 3 out of 4 prac marks.

In 2021, with the class equipped with computers, it was possible to plan a rerun of the course reusing the lecture videos and to aim to do more complete pracs. However, as the start of term approached, the university decided to allow Computer Science students (and others doing lab subjects) back on campus.

2021 started out looking almost like the latter half of 2020, except that once students were allowed back on campus, we could rely on adequate connectivity. Only 50% of seats could be filled and our lecture venue was too small to allow this for the whole class, so we alternated which half of the class was physically present and the other half was supposed to follow online (I used BBB and recorded the session for those who missed it). Changing to in-person pracs partway through required an adjustment. The announcement was too late to change the third prac question, but the fourth one reverted to the pre-COVID model of tutorial-style questions to be answered before leaving the lab. While we could do full in-person pracs in the second half, we continued with the 50% split of the lecture venue. As with 2020, I was flexi-

ble on deadlines and took the best three out of four prac marks as students were still struggling more than in a normal year to keep up.

In addition to the lectures (50% in person), video lectures and pracs, I also ran BBB Q&A sessions.

Given the big differences between years, I do not over-analyze. Class demographics changed particularly between 2019 and 2020 and managing COVID was very different in each year. Class results were reasonably comparable, but assessment strategies also changed from year to year.

4 Reflections

Had the run of the course been intended as a research project, I would have kept a reflective diary. Unfortunately I cannot draw on detail from the students' WhatsApp group as that is not covered by ethics approval.

A course survey (using Google Forms – see the Appendix for the questions, excluding those not answered; the first page is shown in Fig. 3) was completed by 23 members of the class of 73, 31.5%. This is in line with expectations for a return rate [4] if no special steps are taken to encourage participation [19].

A self-administered anonymous online survey is limited by factors that may bias students towards or against participation. Generally selection bias of this type is more likely to skew positive, though the overall effect does not necessarily invalidate conclusions [6]. The possibility of such bias, however, is another reason not to over-interpret the results.

Computer Architecture under Covid – Philip Machanick 2021

This form allows you to give feedback on my computer architecture course; it does not collect any information that identifies you and you can withdraw any time. There is no reward for taking part of penalty for not doing so or for withdrawing.

The results may be used in research or a promotion application. This research is approved by the Rhodes University ethics approval process (ethical clearance certificate number 2021-5127-6190).

***Required**

1. Do you give consent to use your response in research? *

Mark only one oval.

- ☐ Yes
☐ No

2. Is this your first time doing this course? *

Mark only one oval.

- ☐ Yes *Skip to question 9*
☐ No – repeating; did it in 2020 *Skip to question 3*
☐ No – repeating from another year *Skip to question 8*

Fig. 3. First page of student survey is shown in printable format for compactness.

One bias I could detect: 8 students were repeating the class from 2020 but none of those responded on the survey (there is a question to check for this).

Students responded from 21 July 2021 to 9 September 2021; 19 out of 23 responses were received no later than 25 July. 9 September was the exam date, postponed from June because of COVID. The last two responses were the afternoon after the exam. Two responses are not insufficient to assess post-exam perceptions.

4.1 Responses

Constrained responses. Most questions took the form of selection from options. I summarize here some of the most interesting responses.

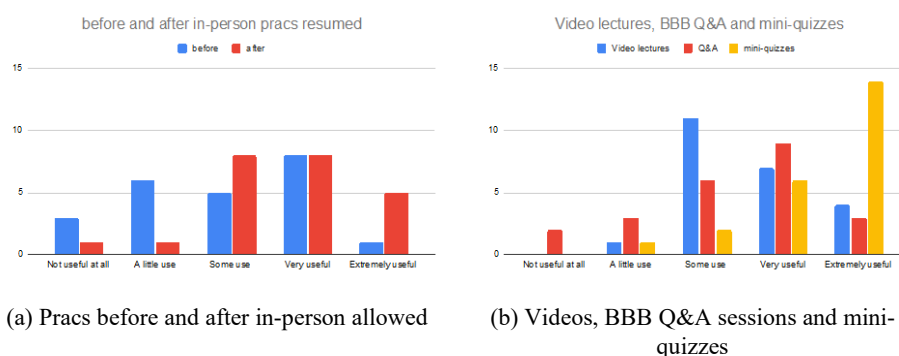


Fig. 4. Adjustments in presentation.

I graph responses on novel approaches introduced for COVID in Fig. 4. Response to the effectiveness of pracs before and after full in-person pracs were introduced (4a) show a clear bias towards full in-person pracs being better, which is no surprise. More interesting is the responses on how useful video lectures, BBB Q&A sessions were and mini-quizzes are. It is possible that responses were biased towards attitudes of those more inclined to fill in an online form, but it is a bit surprising that mini-quizzes are the intervention seen most positively. My previous experience is that formative assessments are mostly ignored (“Is it for marks?” requires a “Yes” to motivate students); possibly pitching them as preparation for other assessments helped.

Of these adjustments, Q&A sessions are the one least obviously useful outside of a pandemic situation as students are in principle free to consult their lecturer outside of lecture times. However, it is my experience that very few do so. Given how positively these online Q&A sessions were viewed by this class, it could be worth doing these in future: they differ from face-to-face consultations in being less personal and also benefit from being recorded for later review.

Flexibility on deadlines could be seen as COVID-specific. The question “How was timing of deadlines?” offered the following choices:

- Good to allow flexibility for those who needed more time

- Understand allowing flexibility for those who needed more time but did not suit me
- Not a big issue for me
- A bit less flexibility would be better

Forcing everyone to work to deadlines is best

Responses (Fig. 5) are heavily skewed towards supporting the concept; only one respondent was strongly opposed to flexibility. However, outside the COVID context, it is not clear that this sort of flexibility would be advantageous. The related idea of assessing pracs as best 3 out of 4 means that it is possible for class members to skip content, which is not necessarily helpful for preparation for other assessments or meeting course objectives.

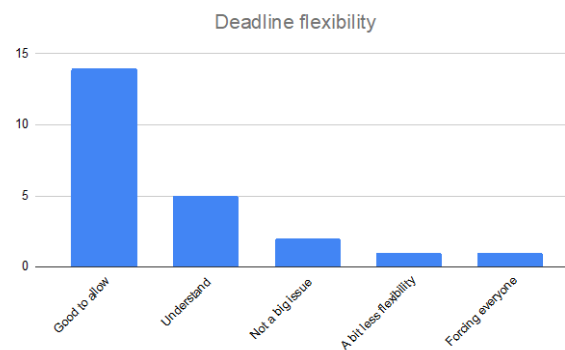


Fig. 5. Responses on deadline flexibility.

Written responses. For responses to a write-in question, “I tried to get through material reasonably fast then allowed time to catch up. Please say a few words about how this did or did not work for you.”, I manually coded the responses as negative, neutral and positive. Of the 23 responses, 3 were negative, 5 were neutral and 14 positive. I scored a response as positive if it supported the approach unconditionally, neutral if it included positives and negatives and negative if it noted no positives.

The biggest source of unhappiness was going too fast; too much content was also mentioned as were inadequacies of tutoring or tutors.

At the end of the survey, I asked two open questions: “Overall positive comments” and “Overall negatives and areas for improvement”. 17 of the 23 positive comments focused on the way the course was presented or the lecturer. Comments included patience with students, understanding the difficulties of online learning and differences in individual circumstances. Compared with previous surveys, the level of interest and positivity is far stronger, including the desire to study the subject further.

Negatives covered a number of areas: whether there was too much content or it was too fast at the beginning, whether the video lectures could have more detail and

quality of tutoring. A complaint in common with previous years was the absence of extra tutoring that we call ADP (academic development programme) that is only offered to first-year students. ADP is supposed to be a catch-up programme for those with inadequate schooling and is never offered in second year.

On the whole the negative comments are not much different from a typical year.

Tutoring issues could have been separated more cleanly if this was asked more specifically in the survey. Negatives about tutoring were not coupled with complaints about the course or lecturer. The most serious complaint about the course was pacing – while some appreciated going fast at first then slowing down, others felt that there was too much content.

4.2 Course Results

Table 1 compares the class mark across years. Since the architecture module is examined as part of a bigger course (CS201, the first semester of second year Computer Science), we do not have separate exam results on record for the module. The higher class mark in 2020 may reflect a dramatic cut in the practical content (students who were sent home could not be assumed to have a computer). 2019 was the last year before a big increase in NSFAS students and was not run under COVID conditions. 2021 and 2022 are the most similar in terms of cohort; the dip in results in 2021 reflects refining the approach under partial COVID conditions and the result in 2022 shows that the lessons learned paid off once COVID restrictions were eased, with a similar result to the 2019 class who were on average from a much strong socio-economic demographic.

4.3 Putting it all together

Overall student responses were far more positive than in any other survey I have run. There are various reasons for this. Being able to put other things aside for the course, many years of experience with education, a deliberate attempt at developing empathy with the class and the general desire at uncertain times to feel supported.

Table 1. Summary of Outcomes.

Year	Class mark %
2019	54.4
2020	57.2
2021	46.3
2022	54.4

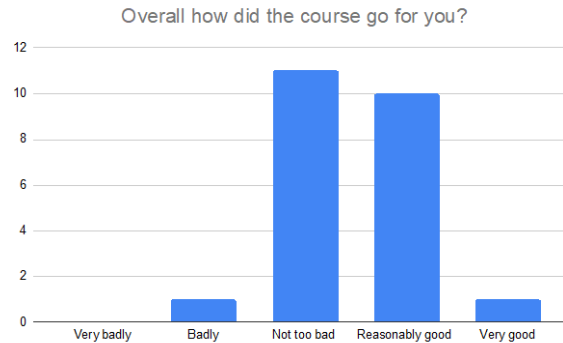


Fig. 6. Overall response on the course.

The effect as seen by respondents is positive overall, as indicated in Fig. 6. To put this in context, all previous courses use relatively high-level languages like Java and Python; coding at machine level is quite a culture shock.

Some of the experience is not repeatable. The strange situation in 2021 of partially being under COVID restrictions and partly not is unique. Some accommodations made for the circumstances would not apply outside that scenario. Students may feel disconcerted by sudden changes in strategy in more “normal” times.

However there is something to learn from this experience about reading a class. The tricky part is formalising this. A strategy like agile development in which feedback is in formal stages would be difficult to implement without imposing extra work on students, a particularly problematic issue with those who are struggling and already time-poor.

Table 2 summarizes lessons from this course that could generalise. Creative strategies like mini-quizzes that provide formative assessment in small doses help to take the pulse of a class. Encouraging participation in formative assessment is tricky; promising that the results *could* be used as summative assessment works though those who want curriculum to be a rigid contract may be offended.

Table 2. Summary of Lessons.

Approach	Assessment
Listening to class	Good but complaints may not be representative
Small online quizzes	Great for feedback if they are done
Flexibility on deadlines	Good for weaker students
Dropping lowest assessment	Reduce need for weaker students to catch up
Keep class guessing what counts	Encourages participation in formative assessment
Don't treat teaching as a rigid contract	Rapid feedback cycles imply adaptation

5 Conclusions

2021 was a relatively successful year despite the complications of COVID and rapidly changing circumstances. At the start of the course, the class was on campus, but in-person lab sessions were not yet permitted. We did not have a lecture venue big enough for the entire class so we had to split the class into alternating in-person attendance with the rest of the class participating either live on BBB or able to watch the recording later. Halfway through the course, we switched to in-person pracs, with lectures remaining as alternating between in-person and online.

While the class was very diverse, the difficulties of 2020, when many went home without a computer or good Internet, were mitigated. Students being able to return to campus and hence good Internet, as well as having their own computer by that time.

Given all the variations from year to year, a comparison is difficult. In 2022, when I reused the same ideas, the overall average for CS201 was 64%, up from 59% in 2021. As illustrated in Table 1, the architecture class mark in 2022 was the same as the class mark for 2019, with a class with a much lower NSFAS component, illustrating that the lessons from the COVID years worked well with COVID-related obstacles removed.

An obvious question to ask is whether the things that worked well in 2021 translate to more “normal” times. When students are dealing with a difficult situation, empathy goes a long way. A former colleague quoted Anne Galloway from Victoria University of Wellington as saying: “Best advice I got when I entered academia: ‘We’re all smart. Distinguish yourself by being kind.’”¹ Is that such a bad thing to aim for?

In a situation where everyone else is struggling and you have the time on your hands to look after students better, it is not that difficult to distinguish yourself in this way. However, a bit of empathy for students can always work.

How about adapting agile ideas, and the social construction model?

Tactics I used like mini-quizzes as a way of keeping touch with how the class is doing can work. The trick is how you use them. If you see them as feedback points to adjust your strategy, you can sell them to the class as preparation for other assessments (like class tests).

Encouraging the class to engage with you on tactics also builds in the ideas of the social construction model. Rather than seeing education as filling a vacuum in students’ heads, the social construction model makes them active participants in learning.

It would be useful to rerun some of these ideas with a finer-grained survey to understand student attitudes as each idea is developed. However, it is important to remember that the more challenged a student is, the more time-poor they are. One approach could be to offer a programme like ADP in a class where you are attempting to understand better where the problems are, and use feedback from ADP to improve pedagogy. Online Q&A sessions, recorded for later review, can also be helpful.

¹ The original quote is from Twitter but ironically, considering the message, the author has been banned from Twitter.

An agile approach to blended learning can be a good fit to the social construction model.

The big challenge is to include novel strategies without creating an imposition on the students who most need improved pedagogy. Integrating feedback on the approach into additions to the course that aid students may take some imagination but as my 2021 experience shows, such strategies can produce worthwhile results.

The biggest single lesson? Empathy goes a long way. It is hard with students whose background is significantly different from your own but to the extent that you can put yourself in your students' shoes, you can be nimble in adapting to changing or challenging circumstances.

Ethics approval

This research is approved by the Rhodes University ethics approval process (ethical clearance certificate number 2021-5127-6190).

References

1. Baham, C., Hirschheim, R.: Issues, challenges, and a proposed theoretical core of agile software development research. *Information Systems Journal* **32**(1), 103–129 (2022). <https://doi.org/10.1111/isj.12336>
2. Barak, M.: Science teacher education in the twenty-first century: A pedagogical framework for technology-integrated social constructivism. *Research in Science Education* **47**, 283–303 (2017). <https://doi.org/10.1007/s11165-015-9501-y>
3. Ben-Ari, M.: Constructivism in computer science education. *Journal of computers in Mathematics and Science Teaching* **20**(1), 45–73 (2001), <https://www.learntechlib.org/primary/p/8505/>
4. Conn, C., Norris, J.: Investigating strategies for increasing student response rates to online delivered course evaluations. *Proceedings of the National Convention of the Association for Educational Communications and Technology* **1**, 77–85 (2003)
5. Ekudden, E., Horn, U., Melander, M., Olin, J.: On-demand mobile media – a rich service experience for mobile users. *Ericsson Review* **78**(4), 168–177 (2001)
6. Goos, M., Salomons, A.: Measuring teaching quality in higher education: assessing selection bias in course evaluations. *Research in Higher Education* **58**, 341–364 (2017). <https://doi.org/10.1007/s11162-016-9429-8>
7. Hayat, F., Rehman, A.U., Arif, K.S., Wahab, K., Abbas, M.: The influence of agile methodology (Scrum) on software project management. In: 2019 20th IEEE/ACIS International Conference on Software Engineering, Artificial Intelligence, Networking and Parallel/Distributed Computing (SNPD). pp. 145–149 (2019). <https://doi.org/https://doi.org/10.1109/SNPD.2019.8935813>
8. Kumar, A., Krishnamurthi, R., Bhatia, S., Kaushik, K., Ahuja, N.J., Nayyar, A., Masud, M.: Blended learning tools and practices: A comprehensive analysis. *IEEE Access* **9**, 85151–85197 (2021). <https://doi.org/10.1109/ACCESS.2021.3085844>
9. Machanick, P.: A social construction approach to computer science education. *Computer Science Education* **17**(1), 1–20 (2007). <https://doi.org/10.1080/08993400600971067>
10. Machanick, P.: MIPS2C: programming from the machine up. *RAMpage Research* (2015)

11. Neumann, M., Bogdanov, Y., Sager, S.: The covid 19 pandemic and its effects on agile software development. In: 2022 The 5th International Conference on Software Engineering and Information Management (ICSIM). pp. 51–60 (2022). <https://doi.org/10.1145/3520084.3520093>
12. Oliver, M., Trigwell, K.: Can ‘blended learning’ be redeemed? E-learning and Digital Media **2**(1), 17–26 (2005). <https://doi.org/10.2304/elea.2005.2.1.17>
13. Piaget, J.: Cognitive development in children: Piaget development and learning. Journal, of Research in Science Teaching **2**, 176–186 (1964)
14. Schmidt, P., Gutfreund, K.: Agile software development during the covid-19 pandemic: A technology company survey. In: Proceedings of the 55th Hawaii International Conference on System Sciences (HICSS). pp. 7111–7120 (2022). <https://doi.org/10.24251/HICSS.2022.856>
15. Sentance, S., Waite, J.: Teachers’ perspectives on talk in the programming classroom: Language as a mediator. In: Proceedings of the 17th ACM Conference on International Computing Education Research. pp. 266–280 (2021). <https://doi.org/10.1145/3446871.3469751>
16. South African Government: About alert system (2021), <https://www.gov.za/covid-19/about/about-alert-system>, accessed 31 March 2022
17. Uhl, C., Freisleben, B.: Performance improvements of BigBlueButton for distance teaching. IADIS International Journal on Computer Science & Information Systems **17**(1) (2022)
18. Yang, W., Shaman, J.L.: COVID-19 pandemic dynamics in South Africa and epidemiological characteristics of three variants of concern (Beta, Delta, and Omicron). Elife **11** (2022). <https://doi.org/10.7554/eLife.78933>
19. Young, K., Joines, J., Standish, T., Gallagher, V.: Student evaluations of teaching: the impact of faculty procedures on response rates. Assessment & Evaluation in Higher Education **44**(1), 37–49 (2019). <https://doi.org/10.1080/02602938.2018.1467878>

Appendix: Student Survey Questions

1. Do you give consent to use your response in research (Yes | No)?
2. Is this your first time doing this course? (Yes | No – repeating; did it in 2020 | No – repeating from another year)?

Questions about repeating from 2020 and previous years are skipped as no one answered them; students would not see these unless they said that they were repeats from previous years.

9. How was the pace of the course? (Very slow | Slow | About right | A bit fast)
10. I tried to get through material reasonably fast then allowed time to catch up. Please say a few words about how this did or did not work for you. (*written answer*)
11. How was timing of deadlines? (Good to allow flexibility for those who needed more time | Understand allowing flexibility for those who needed more time but did not suit me | Not a big issue for me | A bit less flexibility would be better | Forcing everyone to work to deadlines is best)

12. How useful were video lectures? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
13. How useful were Q&A sessions on Big Blue Button? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
14. How useful were pracs BEFORE we allowed the in-person option? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
15. How useful were pracs AFTER we allowed the in-person option? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
16. How useful were mini-quizzes for encouraging learning? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
17. How useful was the class test for encouraging learning? (Not useful at all | A little use | Some use | Very useful | Extremely useful)
18. How much do you believe you have learnt about MIPS assembly programming? (Nothing | A little | Some | Reasonably good amount | A lot)
19. How much do you believe you have learnt about logic (circuits and proofs)? (Nothing | A little | Some | Reasonably good amount | A lot)
20. How much do you believe you have learnt about logic (circuits and proofs)? (Nothing | A little | Some | Reasonably good amount | A lot)
21. Does humour from a lecturer help with interest in the course? (Nothing | A little | Some | Reasonably good amount | A lot)
22. How well do you think I understood your problems in working remotely and tried to work around them? (Nothing | A little | Some | Reasonably good amount | A lot)
23. How well did this course motivate you to learn more? (Not a subject I like at all | Not my favourite subject | This is enough | I would like to learn more | I would like to do an advanced course or research in the area)
24. How well did this course help with other courses? (Nothing applies to other courses | A little applies to other courses | I see where it applies but it did not help a lot with understanding other courses | It helps with some parts of other courses | I helped me understand useful parts of other courses)
25. Overall how did the course go for you? (Very badly | Badly | Not too bad | Reasonably good | Very good)
26. Overall positive comments (*long written answer*)
27. Overall negatives and areas for improvement (*long written answer*)

Improving Student Participation in a Visual Application Programming Course at the University of Zululand

Ndlovu Thamsanqa¹[0009-0006-8170-1352] and Terzoli Alfredo¹[1111-2222-3333-4444]

¹ University of Zululand, KwaDlangezwa, KwaZulu-Natal, SA
ndlovut@unizulu.ac.za and terzolia@unizulu.ac.za

Abstract. A study is being conducted at the University of Zululand's focussed on students in their second year of Management and Information Systems (MIS). The aim of the study is to identify problems in the current teaching-learning endeavour. Programming is notorious for being difficult, especially for students who are not taking Science, Technology, Engineering or Mathematics (STEM) majors. To try to mitigate that difficulty, a teaching strategy incorporating Problem-Based Learning and Computational Thinking is being proposed. The literature examined provided an initial understanding and interpretation of our situation, covering both teaching and learning perspectives. Problem-Based Learning is used as a conduit for delivering content on fundamental concepts. Computational Thinking coordinates the entire learning through a series of clearly defined steps. The expected contribution of this study is to highlight challenges in teaching programming for both students and teachers moving from a transactional paradigm to an event-driven one and suggest remedies where possible.

Keywords: Problem-Based Learning (PBL), Computational Thinking (CT), Software Development Lifecycle (SDL), Fundamental Ideas.

1 Introduction

An elementary programming course in undergraduate education aims at imparting knowledge and capacitating students with foundation and skill to solve real-world problems with computational logic and instruments. It has been noted that programming is difficult for most novice and beginner students. The students at the University of Zululand come from diverse backgrounds with varying levels of exposure to knowledge (in particular, programming related) in their secondary schooling. Profiling our student cohort reveals that *programming-related* skills had already been introduced in their first semester of study, but the comprehension is weak due to the abstract nature and complexity of programming. It makes it even worst for students who are not taking computer majors and pedagogical intervention is necessary by fostering *general education skills* (language, critical thinking and group discussion, study approaches, etc.). Programming-related skills (problem solving, computational thinking, syntax and semantics of a language, etc.) are considered a prerequisite and weak comprehension of such makes programming difficulty [1].

Language plays a pivotal role in the world of computers and spans over several disciplines including, but not limited to programming (programming languages), for specification (specification languages), for verification (logic calculi), in databases (query languages), in operating systems (command languages), but is central to human interaction and communication [2]. Generally, language competence is necessary in motivating students to participate in meaningful ways and progress in their knowledge building effort.

Language barriers may stigmatize a student's take on programming subject and in fact may adversely affect the entire teaching-learning endeavour. Students from privileged backgrounds seem to have a strong grasp of a wide variety of contexts and can easily communicate their thoughts and ideas.

Some research contributions suggest that successful deployment of an effective curriculum requires adaptation and sensitivity to local needs [13]. This is where the role of the teacher comes in, that is to localize content to a "common language" while taking full advantage of opportunities as students adapt throughout the entire learning process.

What motivates our research effort, is the attempt to help students who are in their second programming course. The concerned students are not computer majors (science, technology, engineering, and mathematics) and issues are bound to surface when moving to another programming paradigm. Some issues for inclusion in our research roadmap on teaching-learning Visual Application Programming has been framed by a scheme of questions as follows:

RQ1: What issues arise from students previously introduced to transactional/batch programming moving into event-driven programming?

RQ2: Is the advent of visual interface as a means of accessing the same (text-based) program a deterrent or not?

RQ3: Are the skills acquired earlier sufficient enough to survive through a wide spectrum of education and still be relevant in future?

RQ4: What teaching aides and approaches can we adopt to ensure progression in learning by building on the existing knowledge, taking in account the evolutionary nature of the field?

This research endeavour keeps a clear distinction for inclusion between learning (RQ1, RQ2, and RQ3) and teaching (RQ4). The idea is to find a better way of understanding and expressing our situation.

2 Paradigm Shift and Challenges in the Contemporary Teaching-Learning Approach

The underlying study conducted aims at designing an effective programming course tool and further gives an insight into our pedagogical approach and the underlying theories anchoring our hypothesis.

The university has adopted blended learning into 4CPS242, an intermediary Visual Programming course where traditional contact instructional approach is being combined with emerging online approach. Various Learning Management Systems (LMS)

such as Moodle, SAM, etc have been adopted and play an important role in making teaching-and-learning a possibility.

For the kind of students that we seem to attract, the curriculum for primary and secondary schooling does not intensify programming as an integral part of subjects and most students encounter programming for the first time at tertiary level. Programming as a learning area is mostly taught along a vertical dimension [2]. In other words, programming concepts might be taught at various levels and differing levels of detail and formalization. This might not be true for all learning institutions but in our case a concept can be revisited at an advanced level, in greater depth and complexity as education progresses. Thus, the fundamentalness of an idea can prove to be robust and survive throughout the entire educational spectrum (first year – final year). First semester course starts by introducing students to computer architecture, programming languages and software development fundamentals. Initially, students learn how the computer system works. The practical component of it teaches programming to solve “simple” problems. Seemingly, programming is related to problem solving and there is a positive correlation between computational logic, numeracy, and programming. Table 1 below, depicts various issues addressed by various researchers from primary studies conducted and merits the inclusion for discussion in the following sections. Essentially, student skills that are a prerequisite in the first programming course of our MIS offering and those that should have been attained at the end of our 4CPS242 module are tabulated below as follows:

Table 1. Student skills classification scheme adapted from [6, 18, 20, 16, 17]

Category	Skill	Reference	Novice	Intermediary
Programming related	problem-solving	[2, 6, 15 - 20, 22 - 25]	●	●
	computational thinking	[1, 9, 15, 17 - 19, 23, 31, 32]	●	●
	mathematical ability	[2, 15, 19, 22, 26 - 28]	●	●
	previous knowledge in programming	[1, 17, 21]		●
	run-time aspects consideration	[1, 2, 17, 25, 26, 29]		●
General education & other skills	language competency	[1, 2, 17, 28, 30]	●	●
	critical thinking and group discussion	[1, 16, 18]		●
	time management, study skills and motivation	[1, 15, 18, 19, 22 - 24, 28, 30, 31]	●	●

Over time, the contents of the core CS classes have changed to varying degrees, and the electives are far more diverse and cross-domain [21]. Due to the dynamic evolution of computer science as a discipline, it is necessary that students obtain a schematic depiction of the fundamental ideas, approaches, and techniques of reasoning in computer science. In the long run, a strong grasp of fundamentals at the early stages of development seems to enable students to acquire new concepts successfully [2]. Not only is this beneficial for student's academic endeavors but also for post-tertiary training in that new concepts will appear to be mature variants of concepts accrued earlier.

Student skills drawn from the primary literature review in consideration were categorized into two, namely *Programming-Related* and *General Educational and Other Skills* as can be seen in Table I above. The following sections are framed based on the classification scheme prosed, where relevance and attempt to answer research questions (*RQ1 – RQ4*) would be mapped. Matters arising from literature conducted serve as a guide for our inclusion policy and marshals the discussion of issues that follow.

3 A Curriculum Oriented Towards Fundamental Ideas

Basic programming learning is well known for being complex for many beginner students at the tertiary level. Fundamental programming concepts may include *data types*, *data structures*, *flow control structures*, *classes and objects*. Essentially, a program may consist of several of these building blocks that work together to accomplish a task.

The primary objective of a Visual Application programmer, just like any other programmer, is the generation of code for a particular computational task. Graphical User Interface (GUI) or simply UI design is, in most cases, considered artistic and prevalent within the creative arts discipline. On the other hand, coding is usually considered technical and fits well within the sciences domain. Furthermore, putting a clear distinction between 'coding' skills and 'design' skills has proven to be essential in our case.

Our student cohort tends to be drawn towards the Integrated Development Environment (IDE) tool, probably because of its visually appealing nature and not necessarily on the subject matter at hand which is code generation. As a result, throughout the course there's tremendous progress at the early stages when the focus is on UI lessons, but once the students get to the problem-solving part it appears more like we have hit a brick wall. This has been a growing concern on our part and problem-solving has been cited as central to programming ability (RQ2). It is considered a prerequisite for allowing students to venture into a programming course and has been among the most cited skills (twelve publications).

3.1 Problem-Based Learning (PBL) Teaching Paradigm

PBL learning is a teaching model that is student-centered and has its origins in the 1960's [14]. Its (PBL) strength lies in the application of concepts in solving of problems and generally students work in groups. Through guidance and proper marshalling, learners exhibit strength in developing a viable solution to a defined problem by conducting research, integrate theory and practice, and apply knowledge and skills [4]. The benefit of this is active participation of students that could potentially increase learning outcomes if well planned and executed. Group discussion and critical thinking skills (RQ4) were cited in three publications, associated with the students' willingness to participate, creation of federated knowledge, and peer grading. This skill is considered intermediary and can be acquired by students as they mature at the university.

Problem solving is a central aspect of computer science and programming that requires some abstract thinking in order to provide a creative solution to a given problem. The *Pyramid Theory of Learning* shows that passive listening teaching method has the lowest learning effect [15]. On the other hand, knowledge absorption can increase significantly in learning-by-doing approach. Moreover, rapid improvements have been noted in applying or teaching other peers. Class observation suggests that group learning, active and participatory engagement in a class setting is the most effective. From this we can see that different teaching-learning approaches have their strengths and weaknesses and produce different results on learning process. Therefore, in our intervention we strongly advocate *critical thinking and group cooperative learning, master knowledge* and promote *willingness to participate*. All these traits enable a student to fully realize the transformation of theoretical knowledge into abilities and skills. We strongly believe that if this approach is applied, in combination with other described above (focus on fundamentals), would shape students to become *solution-oriented* and produce the expected learning outcomes.

Problem solving process might be divided into several steps depending on the orientation of the problem (see Figure 1, below):

1. Determining the problem – data collection, description, collation,
2. Problem analysis - abstraction, mathematical modelling,
3. Learning issues – defining data structure, algorithm, flowchart, coding,
4. Meetings and reports – debugging, performance check,
5. Presentation of solutions and reflections – problem backtracking,
6. Conclusion, integration, and evaluation – installation, delivery.

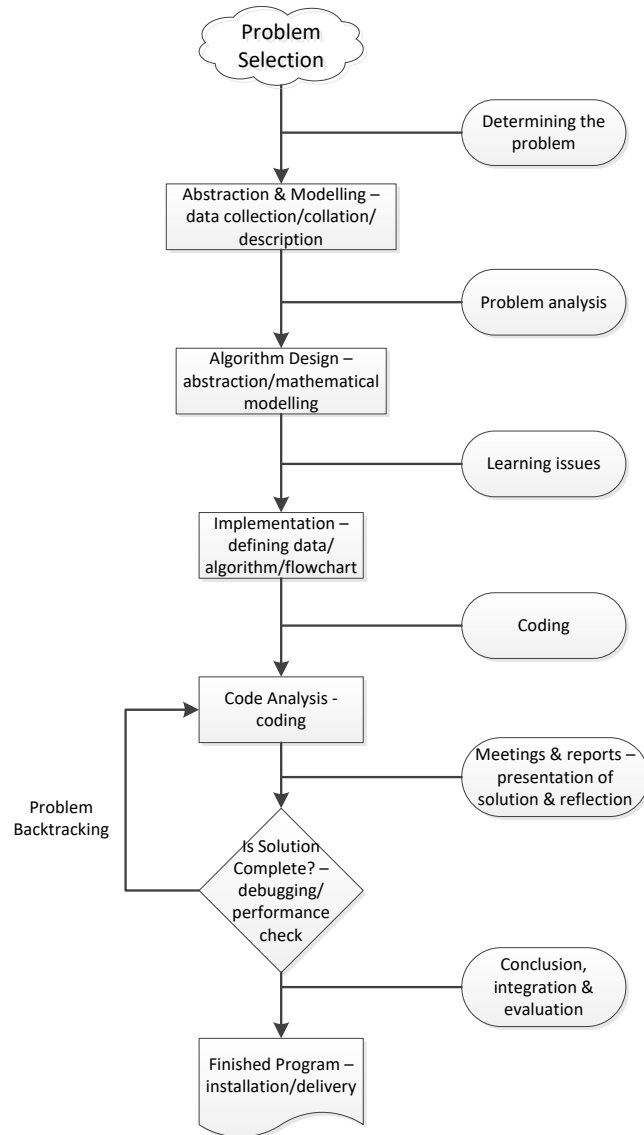


Fig. 1. The flowchart of the proposed *problem-solving* process [19, 16]

The premise for effective implementation of our teaching-learning strategy lies in the fusion of *active participatory learning*, *creation of federated knowledge* and lastly, *peer-grading and the ability to reach a consensus viewpoint*. Furthermore, teachers should hold student's position in high regard throughout the learning process.

In a review of research on novice programmers and introductory programming frequent reports of poor performance and low retention rates have been observed as well [5]. A beginner's expectations and perception in introductory programming course may

affect their viewpoint about the subject negatively [7, 8]. A few cases have been noted with caution in our situation where students are not eager to learn directly in a formal class environment or interacting with whoever authority in charge. Even though they may be attending classes regularly, their passive participation suggests that they would rather learn from their peers/seniors outside of class. It has been noted with caution that prior exposure to programming is not always beneficial, especially to those students that are struggling. Our observations suggest that prior exposure to programming (RQ1) may adversely affect student's morale and conduct with peers in a class environment. Also, ten publications highlighted this tendency as affecting motivation and morale of students who had a bad experience of a programming course.

Other researchers have suggested reasons for failure rates in Introductory Computing in the socio-cultural environment rather than attributing it to individual student's traits [17]. This is where students tend to become isolated and operate in a cocoon, the results of which is a social environment cheering competition over cooperation. A similar trait has been observed with our sample set as well, resulting in a polarized class situation and this may have adverse effects on the pass rate which could be evaluated empirically. Our biggest challenge, and of course within the education sector, is creating student participation and willingness to succeed in a class environment. So, this is where the role of the teacher is most important, which is responsible for choosing and managing what learning activities that are appropriate to be carried out in the classroom [3].

3.2 Fundamental Ideas Crucial in Increasing Learning Outcomes

As mentioned earlier, computer programming has been around for some time and each student is most likely to face several paradigm-shifts during his working career. If that supposition holds true, then we can conclude that much of their academic knowledge would become obsolete over time. So, it is vitally important that the skills acquired earlier must be sustainable and be relevant in the long run (RQ3).

Some studies suggest that the focal point of programming lessons should be the structure of science which relies on the notion of *Fundamental Principles* [2, 6, 15, 18, 19]. The justification for such is that there is always the need to adapt and cope with future developments in the world of computers, given its dynamic evolution. A fundamental notion of computer science (and programming as well) is a schema for thinking, acting, describing, or explaining [2]:

1. Is multidisciplinary and/or observable in multiple ways.
2. May be taught on a vertical dimension across various intellectual levels. Fundamentally, treatment of an idea at primary school level vs. tertiary level may differ only by level of detail and formalisation.
3. Has chronological significance historically and its development would be evident in the long run.
4. Its manifestation can be seen in everyday life and is related to ordinary language, interaction patterns and thinking.

A curriculum oriented towards the notion of fundamental concepts should revisit these basic concepts repeatedly, building upon them the desired outcomes until the student has attained a full grasp of them. Basically, it has been observed that programming concepts are intertwined in many ways and an exact separation and assignment is challenging. *Problem solving, computational thinking, mathematical ability, language competency, time management and study skills* are some of the most important skills throughout the entire learning process and mastery of such is a major determinant of one's success.

4 Computational Thinking Coordinating Problem-solving Process

Computational Thinking (CT) is key during problem solving process, which the employment of fundamental concepts of computing to solve problems, design systems, and understand human behavior [9]. Cross-domain activities that use scenarios and cases that are perhaps concept intensive, help improve effective arithmetic and logical CT skills [17, 23]. Programming education is often practical intensive and promotes and promotes cooperation and collaboration among students often assisted by the teacher in the computer laboratory.

Throughout the whole computation process, whether be it formation of problem-solving ideas or programing, the goal of thinking is around computability or operability [15]. Conceptual grasp is accumulated up by revisiting the same concept several times in greater depth and detail (RQ3). We have observed that there is the skills shortfall in mapping mathematical models into computer models and also the application of CT is weak. This growing concern has been cited by fifteen authors all in all, seven on mathematical skill and nine on CT skills, respectively. Both these skills are an essential part of problem solving which is crucial in gaining mastery in programming. Also, student's lack of flexibility in algorithm application is a serious obstacle. Another critical concern within our sample set is that students tend to focus exclusively on the "visible" static aspect of concepts [17, 25], program text in the UI but not necessarily with the computer architecture in mind (RQ1 and RQ2). Students tend to have a skewed view of a program in that it is not perceived as writing instructions for the notional machine to execute where run-time aspects are key and play an important role. Six publications have highlighted this problem as a growing concern.

All of this is paving way for teaching programming at an advanced level in the later stages of the course development. On the other hand, we should enhance student's ability to solve scientific problems, and the intuitive ability to transform such thinking patterns to produce functional modules that can act and interface with one another. Effective application of CT requires well roundedness in understanding and making use of logic and tools while coding a program's instructions.

Despite varying views on what instructing a computer is, when it comes to software development the dominant approach is based on Software Life Cycle SDL. According to [10] this methodology is based on a five-step classical problem solving: *What, How, Do it, Test, and Use*. As a result of this, it seems reasonable to analyze this (SDL)

process along the tracks of fundamental ideas paving way to the required solution. In this process, the teacher prepares a trigger question at each step, progressively guiding students to critique emanating solutions, eventually marshalling students to form a complete understanding of the knowledge points about CT. Continuous interaction with teachers is key in maintaining student's cognitive sense of the problem [11] as they might get destructed easily. Let us consider the stages as follows:

Problem Selection: The teacher chooses and manages what learning activities are appropriate to be carried out in the classroom. The strategy is selecting a problem that covers topics of interest extensively and in detail. The growing diversity of topics relevant to an education in Computer Science and the increasing integration of computing with other disciplines creates peculiar challenges for this effort.

Problem Analysis: In this phase the problem to be solved and all relevant aspects of the environment in which the proposed software system would be used are formally stated, then follows data collection, collation, and description. Some studies indicate that some kind of live data can increase student's comprehension of the problem [6, 18, 19, 21]. This process is mainly about requirements engineering.

Abstraction and Modelling: One of the CT's features is abstraction that can be used to create multiple instances of the problem and further compare and contrast solutions that follow. Central to this stage is the issue of computability and complexity in a theoretical sense.

Algorithm Design: While the problem analysis stage describes properties of the software product without paying attention to how they will be implemented, the programmers then develop a model of the system that satisfies the requirements when compiled into a program. Students are instructed to design the algorithm that can solve the problem within a finite number of steps.

Implementation: Central to this stage is the idea of conversion of an algorithm into an executable program consisting of control structures (sequence, loop, alternative etc.), data structures (aggregation, generalization etc.), and modules.

Code Analysis: The main idea in this stage is the quality control. All components of the program are checked either formally or by test cases for partial and total correctness. After checking for correctness, the program's performance is assessed. This is where presentation of solutions and reflections in class occurs.

Finished Program: Central to this stage is the process of conclusion, integration, and wrapping up of the proposed solution. In real life SDL, the process entails installation, delivery to the client and maintenance of the product.

Throughout the entire teaching-learning process, the student participation is key, and the teacher must stimulate student's thinking, with full inquiry, provoke reasoning, and provide clues that stimulate them to conclude the subject matter. This growing concern has been highlighted by three authors cited as listed under general education skills category.

5 Effectiveness Analysis

To effectively analyze our teaching method combining PBL and CT on students' achievement, a comparative study is being conducted in the 2022-2023 second semester. The population sample consists of second year Information Science students who had been introduced to Introductory Programming in their first semester of study. We plan to conduct the study in three cycles consisting of planning, implementation, and reflection [12].

Data gathering will be two types of data, namely quantitative data and qualitative data. Quantitative data can be obtained from test and assignment results. While qualitative data will be taken from student's critical thinking skills, their interest in participation, their interaction with prescribed material, and student's ability to describe learning outcomes.

6 Discussion

Problem solving is essentially, considered a crucial skill in programming. Some authors in this review argue that the primary objective of a programming course is to gain proficiency in employing CT [1, 9, 15, 17 - 19, 23, 31, 32]. This could effectively be achieved by building on the strength of fundamental ideas and improving mathematical abilities, rather than imposing syntax and semantics overhead of a programming language. Even though problem solving matter has been raised by several authors, the subject matter lacks consistency and rigor and is still a pending issue to be considered by research community.

The advent of visual interface as a means of accessing the same (text-based) program has created challenges for our student cohort in making progress and meeting (our) expectations. The explanation behind this could emanate from the fact that our student cohort have already been introduced to programming in the first semester of study. So, we are dealing with individuals who might have been adapted to certain tendencies good or bad, formal or informal that serve as a winning formula for them or as a way of seeking refuge in this challenging and ever evolving world of computers. Previous experience in programming and associated side-effects have been noted by several authors contacted in our primary literature [1, 17, 21].

There has been persistent and widespread reports of high failure rate and low retention rates on novice programmers and introductory programming [5]. A beginner's expectations and experiences in introductory programming course may affect their attitudes toward the subject negatively [7, 8]. A few cases have been noted with caution in our situation where students are not eager to learn and interact with classmates and with whoever authority in charge. Even though they may be attending classes regularly, their passive participation suggests that they would rather learn from their peers/seniors outside of class. It has been noted with caution that prior exposure to programming is not always beneficial, especially to those students that are struggling. Our observations suggest that prior exposure may adversely affects students' expectations, work habits,

attitude and confidence, and perceptions of self and peers. Three authors [1, 17, 21] in our reviewed papers also raise this concern.

As highlighted by several authors [24, 26], computer programming has been around for some time and each student is most likely to face several paradigm-shifts during his working career. Fundamental principles in programming have remained robust and relevant over time [2]. So, it is vitally important that the skills acquired earlier must be sustainable and relevant over time [33].

Lack of ability to map mathematical models into computer models and also the employment of CT is weak. This growing concern has been cited by several authors in our reviewed literature, on mathematical skill [2, 15, 19, 22, 26, 27, 28] and others on CT skills [1, 9, 15, 17 - 19, 23, 31, 32]. Both these skills are an essential part of problem solving which is crucial in mastery proficiency in programming. Also, student's lack of flexibility in algorithm application is a serious impediment.

Another critical concern within our sample set is that students tend to focus exclusively on the "obvious" static aspect of concepts, that is visible in program text but not necessarily with the computer architecture in mind (RQ1 and RQ2). Students tend to have a skewed view of a program in that it is not perceived as writing instructions for the notional machine to execute where run-time aspects are key and play an important role. Six publications [1, 2, 17, 25, 26, 29] have highlighted this problem as a growing concern.

Pedagogical interventions are crucial to mitigate and remedy the obstacles we face in our teaching-learning endeavor. Constant curricular updates and reviews are necessary to keep our programming study activity afloat considering the evolutionary nature of the world of computers. PBL teaching paradigm empowers students by enabling them to conduct research, integrate theory and practice, and apply knowledge and skills to develop a viable solution to a defined problem [4, 16, 19]. The benefit of such is active participation of students that could potentially increase learning outcomes if well planned and executed. Group discussion and critical thinking skills were cited in several publications, associated with the student's willingness to adopt new study skills [1, 15, 18, 19, 22, 23, 24, 28, 30, 31] to stay confident and motivated.

Concepts involved in programming are too abstract to comprehend and are unusually tightly coupled. The problem with such is that the comprehension of one concept might determine the comprehension of another closely linked concept. This has proven to be challenging for students, particularly who are not taking computer science majors and come from sub-optimal educational backgrounds. So, there is a dire need to adopt creative teaching-learning strategies that would improve student interest and participation in the process.

References

1. Medeiros, R., P., Ramalho, G., L., Falcão, T., P.: A Systematic Literature Review on Teaching and Learning Introductory Programming in Higher Education. In: IEEE Transactions on Education, vol. 62, no. 2. doi:10.1109/TE.2018.2864133 (2019).

2. Schwill, A. Computer science education based on fundamental ideas. In: Passey, D., Samways, B. (eds) Information Technology. IFIP Advances in Information and Communication Technology. Springer, Boston, MA. doi:10.1007/978-0-387-35081-3_36 (1997).
3. Prince, M.: Does Active Learning Work? A Review of the Research. In: The Research Journal for Engineering Education, 93 (3). doi:10.1002/J.2168-9830.2004.tb00809 (2004).
4. Larmer, J.: Project-based learning vs. problem-based learning vs. X-BL. Retrieved from <http://www.edutopia.org/blog/pbl-vs-pbl-vs-xbl-john-larmer> (2014).
5. Robins, A. Novice Programmers and Introductory Programming. In S. Fincher & A. Robins (Eds.), The Cambridge Handbook of Computing Education Research (Cambridge Handbooks in Psychology, pp. 327-376). Cambridge: Cambridge University Press. doi:10.1017/9781108654555.013 (2019).
6. Rizvi, M., Humphries, T.: A Scratch-based CS0 course for at-risk computer science majors. In: Frontiers in Education Conference Proceedings, Seattle, WA, USA, 2012, pp. 1-5, doi:10.1109/FIE.2012.6462491 (2012).
7. Kinnunen, P., Simon, B.: My program is ok—am I? Computing freshmen’s experiences of doing programming assignments. *Computer Science Education*, 22(1), 1–28. doi:1080/08993408.2012.655091 (2012).
8. Rogerson, C., Scott, E.: The fear factor: How it affects students learning to program in a tertiary environment. In: *Journal of Information Technology Education: Research*, 9(1), 147–171. doi:10.28945/1183 (2010).
9. Yizhen, Z.: The concept and detailed definition of computational thinking of Carnegie Mellon University. In: American computer journal, Communications of ACM. March (2006).
10. Blum, B. I. Beyond Programming: To A New Era of Design. In: Oxford University Press (2006).
11. Eichmann, B., Goldhammer, F., Greiff, S., Pucite, L., and Naumann, J. The role of planning in complex problem solving. doi:10.1016/J.COMPEDU.2018.08.004. (2019).
12. Sumarni, W., S. Wardani, S. Sudarmin, D. N. Gupitasari. Project Based Learning (PBL) to Improve Psychomotoric Skills: A Classroom Action Research, Indonesian Journal Of Science Education, 5 (2): 157-163. doi:10.15294/JPII.V5I2.4402 Authors: (2016).
13. Mehran, S. Curriculum Guidelines for Undergraduate Degree Programs in Computer Science. In: The Joint Task Force on Computing Curricula Association for Computing Machinery IEEE-Computer Society. doi:10.1145/2594168 (2013).
14. Savery, J., R. Overview of problem-based learning: Definitions and distinctions. *Interdisciplinary Journal of Problem-based Learning*, 1(1). doi:10.7771/1541-5015.1002 (2006).
15. Ding H. Teaching Research on Computer Programming based on Cultivation of Computational Thinking. In: IEEE International Conference on Computer Science and Educational Informatization (CSEI), Kunming, China, 2019, pp. 72-75, doi:10.1109/CSEI47661.2019.8938909. (2019).
16. Lubis, R., R., Irwanto, I., Harahap, M., Y. Increasing Learning Outcomes and Ability Critical Thinking of Students Through Application Problem Based Learning Strategies. In: *International Journal for Educational and Vocational Studies*, 1 (6), 524-527. doi:10.29103/IJEVS.V1I6.1679 (2019).
von Hausswolff, K., Practical thinking while learning to program – novices’ experiences and hands-on encounters, *Computer Science Education*, 32:1, 128-152, doi:10.1080/08993408.2021.1953295 (2022).
17. Figueiredo, J., García-Peñalvo, F., J. Increasing student motivation in computer programming with gamification. In: IEEE Global Engineering Education Conference (EDUCON). doi:10.1109/EDUCON45650.2020.9125283 (2020).

18. Gao, P., Lu, M., Zhao, H., Li, M. A New Teaching Pattern Based on PBL and Visual Programming in Computational Thinking Course. In: The 14th International Conference on Computer Science & Education (ICCSE 2019), Toronto, Canada. doi:10.1109/ICCSE.2019.8845503 (2019).
19. Thorgeirsson, S., Su, Z. Algot: An Educational Programming Language with Human-Intuitive Visual Syntax. In: IEEE Symposium on Visual Languages and Human-Centric Computing (VL/HCC). doi:10.1109/VL/HCC51201.2021.9576166 (2021).
20. Voas, J., Kuhn, R., Paulsen, C., Schaffer, K. Computer Science Education in 2018. In: IT Professional, vol. 20, no. 1, pp. 9-14., doi:10.1109/MITP.2018.011021350 (2018).
21. Mladenović, M., Krpan, D., Mladenović, S. Introducing programming to elementary students novices by using game development in python and scratch. In: International Conference on Education and New Learning Technologies. doi:10.21125/EDULEARN.2016.1323 (2016).
22. Li, X., Gu, C. Teaching reform of programming basic course based on SPOC blended teaching method. In: 15th International Conference on Computer Science & Education (ICCSE), Delft, Netherlands, 2020, pp. 411-415. doi:10.1109/ICCSE49874.2020.9201802 (2020).
23. Maravić Čisar, S., Radosav, D., Pinter, R., Čisar, P. Effectiveness of Program Visualization in Learning Java: a Case Study with Jeliot 3. In: International Journal of Computers, Communications & Control (IJCCC) 6(4):669-682. doi:10.15837/IJCCC.2011.4.2094 (2011).
24. Sorva, J. Notional machines and introductory programming education. In: ACM Trans. Comput. Educ. 13, 2, Article 8, 31 pages. doi:10.1145/2483710.2483713 (2013).
25. Julie, H., Bruno, D., Patrick, H., Tony, L., Object-Oriented Programming: Diagnosis Understanding by Identifying and Describing Novice Perceptions. In: IEEE Frontiers in Education Conference (FIE). Uppsala, Sweden, pp. 1-5. doi:10.1109/FIE44824.2020.9273990 (2020).
26. Chibizova, N., V. The Problems of Programming Teaching. In: IV International Conference on Information Technologies in Engineering Education (Inforino), Moscow, Russia, 2018, pp. 1-4, doi:10.1109/INFORINO.2018.8581834 (2018).
27. Laakso, M.J., Kaila, E. Rajala, T. ViLLE – collaborative education tool: Designing and utilizing an exercise-based learning environment. Educ Inf Technol 23, 1655–1676 (2018). doi:10.1007/s10639-017-9659-1 (2018).
28. Fisher, D., H., Bian, Z., Chen, S. Incorporating Sustainability into Computing Education. In: IEEE Intelligent Systems, vol. 31, no. 5, pp. 93-96. doi:10.1109/MIS.2016.76 (2016).
29. Perera, P., Tennakoon, G., Ahangama, S., Panditharathna, R., Chathuranga, B. A Systematic Mapping of Introductory Programming Languages for Novice Learners. In: IEEE Access, vol. 9, pp. 88121-88136. doi:10.1109/ACCESS.2021.3089560 (2021).
30. Drini, M. Using new methodologies in teaching computer programming. In: IEEE Integrated STEM Education Conference (ISEC), Princeton, NJ, USA, 2018, pp. 120-124, doi:10.1109/ISECon.2018.8340461 (2018).
31. Kesselbacher, M., Bollin, A. Quantifying Patterns and Programming Strategies in Block-Based Programming Environments. In: IEEE/ACM 41st International Conference on Software Engineering: Companion Proceedings (ICSE-Companion), Montreal, QC, Canada, 2019, pp. 254-255, doi:10.1109/ICSE-Companion.2019.00101 (2019).
32. Fisher, D., H., Bian, Z., Chen, S. Incorporating Sustainability into Computing Education. In: IEEE Intelligent Systems, vol. 31, no. 5, pp. 93-96. doi:10.1109/MIS.2016.76 (2016).

Using Design Science Research to Enable Performance Prediction for IS&T Students

Rushil Raghavjee¹[0000-0002-6010-1431], Rontala Prabhakar Subramaniam²[0000-0003-2719-3503]
and Irene Govender³[0000-0002-4499-1091]

¹ University of Kwazulu-Natal, Pietermaritzburg, South Africa, raghavjee@ukzn.ac.za

² University of Kwazulu-Natal, Pietermaritzburg, South Africa, prabhakarr@ukzn.ac.za

³ University of Kwazulu-Natal, Westville, South Africa, govenderi4@ukzn.ac.za

Abstract. The area of learning analytics has generated substantial interest around the world as technology has evolved in academia. The need to track learner performance and development is crucial in the era of growing technological and learning trends. This paper describes part of a larger study that uses design science research in order to implement learning analytics within a specific discipline at a university, specifically the prediction of academic performance for IS&T students. Prediction is performed using machine learning algorithms that are applied to student demographics, past academic performance and student LMS log data. The intention of this research is to describe the use of design science research for the development of an artifact that being a learning analytics process model that guides the application of learning analytics at institutions of higher education. The artifact has the potential to benefit both the areas of teaching and learning as well as practitioners of learning analytics research. From a teaching and learning perspective, the process model lays the foundation for teaching staff to detect students that fall under the risk of failing and plan for strategies to address student problems. From a research perspective, the artifact can guide a researcher or practitioner through all the steps related to a learning analytics project or initiative.

Keywords: Learning analytics, process model, design science

1 Introduction

Over the past decade, the advances in data storage and processing technologies have necessitated the need for better data analysis and action based on this analysis. Higher education institutions (HEIs) in South Africa have been slow to take advantage of this as the focus continues to be on the development and updating of curriculum as well as finding and improving new teaching pedagogies [24]. The advent of learning analytics has resulted in a shift from only focusing on pedagogy development to how the characteristics and actions of students play a role in the academic performance of students. The most common definition of learning analytics is “The measurement, collection, analysis and reporting of data about learners and their contexts, for the purpose of understanding and optimizing learning and the environments in which it

occurs. Learning analytics are concerned with improving learner success” (p.24) [29].

The benefits of learning analytics have been well documented by [28], [10], [21] amongst others. A key benefit is the ability to better understand student learning habits, thus allowing teaching staff to adjust their teaching methods to accommodate individual learning behaviours [22].

Design Science is a research methodology where the objective is to develop and evaluate one or more artifacts to solve a problem. Such artifacts include software applications or processes amongst others. According to [14], the aim of design science is to create novel knowledge and understanding about how artifacts can be designed and used to solve problems in various domains. The design science process involves first identifying a potential problem or opportunity. This is followed by development of a design theory or framework, resulting in the creation of an artifact. Finally, the artifact is evaluated in terms of its effectiveness in meeting its desired outcome [23]. The results of design science research can lead to new theories, models, and best practices that can be used by practitioners and/or researchers to improve their work.

This study applies the steps used in the design science research model proposed by [23] to design an artifact in the form of a learning analytics process model. The process model describes all the stages involved in the process of learning analytics from data acquisition to performance prediction. The process model includes tools and techniques that can be substituted for other tools based on the context of the study or the learning analytics project being conducted. A process model of this type is beneficial for guiding LA researchers through the learning analytics process from the stage of data acquisition until the end stage of developing and validating a prediction model. From a teaching and learning perspective, the end result of the process model can provide insight for teachers as to what features (attributes) are important to consider when understanding student performance as well as how these attributes play a role in the determination or prediction of student performance.

Section 2 looks at the learning analytics concept and previous related work. The section also discusses the area of design science research and its relationship to information systems research. Section 3 outlines the objectives of the study in this paper. Section 4 describes the construction of the process model through design science research. Section 5 is a discussion on the developed artifact for this study in terms of its contribution towards teaching and learning. Finally, in section 6, the conclusion and ideas for future research are provided.

2 Related work

2.1 Information Systems Research

The objective of information systems (IS) within an organization is to improve efficiency and effectiveness of processes continuously [14]. Information systems research, therefore, plays a key role in achieving this objective, which is executed either through behavioural science or design science research [14].

The area of behavioural science arises from natural science research. From the IS point of view, this research paradigm is used for testing as well as providing reasoning for theories to explain the various IS related activities such as analysis and design, implementation and use within a business or related association [15]. This is seen as the more common area of research in IS and provides researchers with relevant information regarding, how and why people act, relationships between technology and organizations, and guidance relating to the management and improvement of IS-related activities [15].

Design science research, on the other hand, is seen as an area of research for computer science and engineering disciplines, and aims to address problems that have unstable requirements, constraints defined within an uncertain environmental context, complex interactions among subcomponents of the problem and its solution, difficulty in adapting to change, and a critical dependence on human cognitive abilities to produce effective solutions [15]. Design science results in the development of new artifact(s) which can then be evaluated using methodologies of behavioural science. Thus, both behavioural and design science share an important association that is required for the development and enhancement of information systems. This relationship is illustrated in Figure 1 [15].

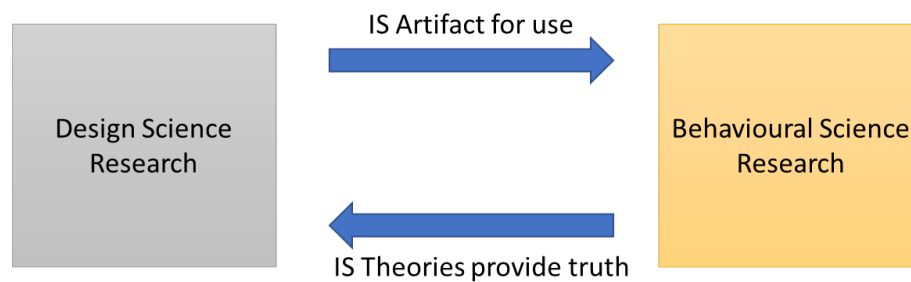


Fig. 1. Research relationship between design science behavioural science [15]

2.2 Design science research

Design science is said to be a pragmatic research paradigm, meaning that the objective of the research would be to attempt to meet the objectives of the research through any viable research philosophy such as positivism, post modernism or critical realism [15].

The design science research model proposed by [23], shown in Figure 2, is one of several models or frameworks found in the literature (such as [32], [3] and [15]). This model is made up of six stages that guides the researcher through the design science research project. The first stage, “identify the problem and motivate”, is the forming of the problem definition and outlining the benefits of the solution. Once this is accomplished, the next stage is followed, “define objectives”, where the goals of the project are outlined. According to [23], the goals should be design and development oriented, resulting in the creation of an artifact. The next stage of “design and development” is the creation of the artifact based on the objectives specified. The artifact can be a model, framework, instance, object or resource that contributes towards enhancing the efficiency and effectiveness of some aspect within the organization [23]. Once the artifact has been created, it is “demonstrated” in the form of case studies, experiments or simulations. This is followed by the penultimate stage of “evaluation”. Once these stages are complete, the final stage is that of “communication” where the results are presented to the relevant stakeholders [23].

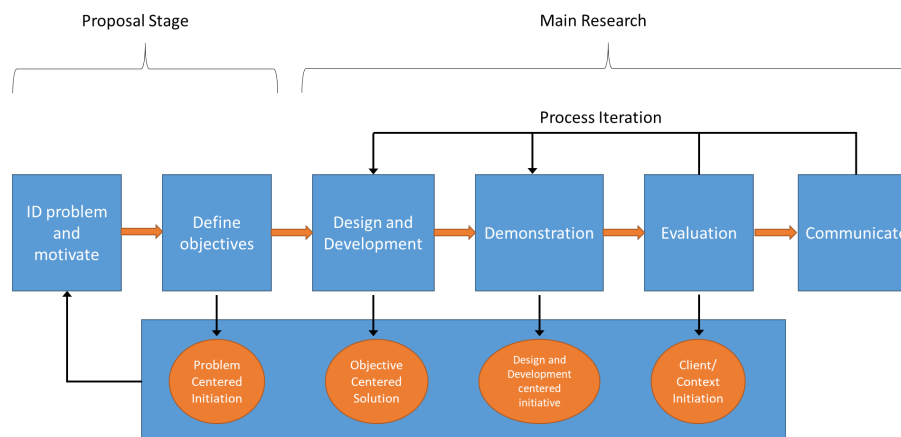


Fig. 2. Design science research model [23]

2.3 Learning analytics

Learning analytics is a bricolage field, inspired from other fields such as statistics, mathematics, data mining, and psychology to name a few [8]. Depending on the context of the learning analytics study, researchers choose from four available paths of

analytics that being descriptive, diagnostic, predictive, and prescriptive [4] to conduct their research. Depending on which path is being followed, researchers can understand various aspects of teaching and learning. Descriptive analytics allows for better understanding of teaching pedagogies through statistics and visualizations [5]. The aim of diagnostic analytics is to better understand the root causes of learning problems through data discovery and pattern identification [4]. As the name states, the objective of predictive analytics is to anticipate what will happen based on the past and current characteristics or activities of students. Finally, prescriptive analytics deals with the issue of identifying the most efficient, effective strategy to achieve the required goals [4].

In the current higher education climate, learning analytics offer a number of benefits. The main beneficiaries of learning analytics are students, educators, administrators, and the research community [28]. One of the main benefits arise from the use of statistical and prediction techniques that allow educators to gain insight into student problems and needs. This allows for not only detecting possible student failure [10], [21] but also to better understand how students learn and get involved in the learning process [20]. Prediction techniques assist in understanding how students learn and allows for the development of early intervention and improvement strategies [7].

From a teaching perspective, learning analytics allows lecturers or teachers to revise learning activities with the objective of improving course content quality [21]. In addition, learning analytics could improve the use and allocation of resources based on prediction of student enrollment and requirements to maximize graduation throughput [18].

The majority of studies relating to learning analytics have focused on implementation studies where a technique or group of techniques have been applied to a variety of datasets. Also common are studies surrounding challenges of learning analytics implementation, learning analytics implementation requirements and learning analytics overview studies.

2.4 Learning analytics frameworks and models

A number of frameworks and process models have been developed surrounding learning analytics. The objectives of these frameworks and models focus on the process of learning analytics or the guidance provided on the implementation and application of learning analytics at an institution.

In terms of implementation of learning analytics initiatives, there are various frameworks such as the learning analytics reference model [6], the ROMA framework adapted by [9], The Let's Talk framework by [33] and the PERLA framework [7]. These frameworks cover institutional requirements for learning analytics implementation such as data source requirements ([6], [12], [7]), individuals and stakeholders ([6], [12], [9], [7]), institutional purpose and objectives ([6], [12], [9], [7]), tech-

niques required ([6] , [12] , [6]), limitations ([12], [33]), strategy and policy development [9], barriers to implementation [9], staff requirements [9], infrastructure [33] and indicators for personalized learning [7].

Process models, on the other hand, focus on the processes required to carry out learning analytics. The five-stage learning analytics model by [5] was established for the extraction of information from large datasets (see Figure 3). The five stages are capture data, report data patterns, develop the prediction model, intervention, and refinement of the model.

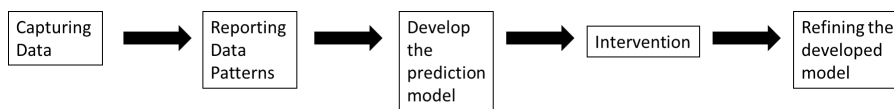


Fig. 3. Five-Stage Learning Analytics Model [5]

While this model appears to be linear with a starting and ending stage, the learning analytics life cycle [7] follows an iterative, cyclical process (see Figure 4). The first stage of data collection is followed by data processing and storage. This is followed by the stages of analysis and visualization. At this point, the necessary stakeholders act and form new learning activities based on the analysis and visualization. The cyclical process then restarts with the first stage of data collection [7].

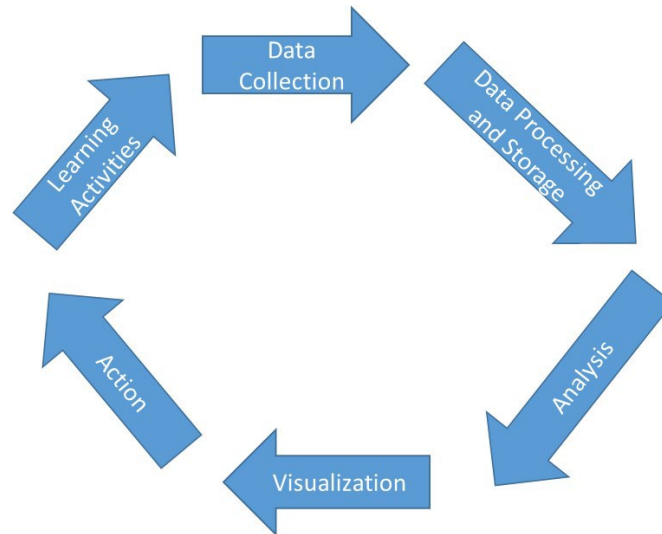


Fig. 4. Learning Analytics Life Cycle [7]

The LA model presented by [30] follows a similar approach and is described as a systematic representation of analytics (Figure 5). The model consists of 7 components, that being collection, storage, cleaning, integration, analysis, representation

(visualization) and action. The model describes the process of learning analytics as circular, with each component requiring a combination of individuals (data team), tools and techniques in order to function effectively [30].

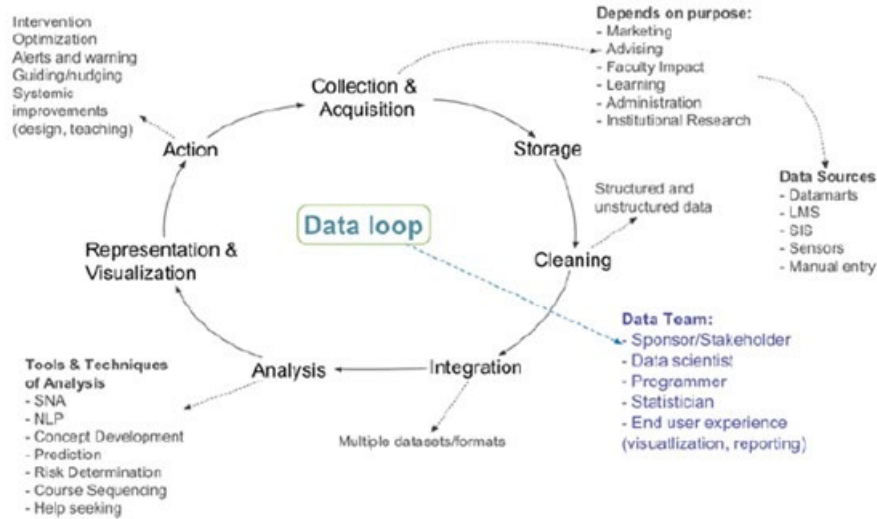


Fig. 4. Learning Analytics Model [30]

From an implementation perspective, all three of the above models show that the first step within learning analytics is the collection of data. As seen in Figure 4 and Figure 5, once data has been collected, it must be cleaned and prepared before any analysis can be conducted. Several approaches are used such as removal of duplicates [2], removal of attributes (features) [28], categorizing data items into groups [26] and dealing with missing or inconsistent data [2]. When conducting analysis, sampling can be applied, especially in cases where there is a dominant class value (imbalanced datasets). Sampling techniques include oversampling [10], undersampling [10] and SMOTE [11]. Feature selection [2] can also be used for analysis and is known to improve algorithm performance and reduce overfitting of prediction models [2], [27]. For prediction and analysis, learning algorithms are used with decision trees [16], naïve bayes [25], random forests [17] and neural networks [1] being the most commonly used algorithms.

3 Objectives of the study

The research presented here is part of a larger ongoing study, where the aim is to predict student academic performance for students within the discipline of Information Systems and Technology.

The objectives of the research presented in this paper is thus as follows:

1. To describe and discuss how the learning analytics study is being presented in terms of design science research.
2. To present and discuss how the artifact generated from this research will contribute towards teaching and learning within the discipline and the greater learning analytics community.

4 Research methodology

This section covers how the design science research methodology was used to carry out the learning analytics study. The design science research model by [23], described in section 2.2 and illustrated in Figure 2, was used for the larger study. The steps described in sections 4.1 to 4.6 represent the steps outlined in the design science research model.

4.1 Identify the problem and motivate

The identification and necessity of the problem stems from the need to make better use of the digital data being captured daily at HEIs, coupled with the continuous advancement of data storage and processing technologies. The need for improved data analysis is also essential in the face of the lack of resources being available to HEIs while student enrollments are on the increase [19]. As learning analytics is fairly new to the African continent, a guide in the form of a process model would be a useful artifact to add to the body of knowledge. Thus, the problem of bringing in learning analytics to HEIs can be addressed by meeting specific objectives which are discussed in 4.2.

4.2 Defining the objectives of the solution

The objectives of the larger study can be conceptualized from the problem stated in 4.1. For the larger study being researched, the objectives were as follows:

1. To merge the relevant HEI data sources in preparation for prediction
2. To perform extraction, cleaning and anonymization of the integrated data
3. To train the integrated dataset with the objective of determining patterns and useful information for predicting the performance of students
4. To determine the effectiveness of the training technique using various performance metrics
5. To evaluate the results of the generated artifact and compare them against the performance of other studies within the domain of educational data mining or learning analytics.
6. To better understand and discuss the potential impact of the predictions in terms of changing how students interact with course content as well as how teaching staff manage content and keep track of student interactions.

4.3 Design and development

During this stage, the artifact(s) is/are created. For the larger study being conducted, two key artifacts are developed. The first artifact is a process model that goes through the required steps to perform learning analytics (the focus of this paper). This includes the steps of data acquisition, anonymization, cleaning and prediction. The second artifact is a dataset consisting of student demographics, assessment and LMS log data.

In the case of the process model, the first step is data acquisition. This step involves not only obtaining the data from the data sources but also seeking permission to obtain this data from the gatekeeper. Permission is obtained through the ethical clearance process as well as agreeing to satisfy POPIA requirements. Once the data has been obtained, it must be anonymized, cleaned and prepared for analysis. The third step in the process model is to apply prediction to the dataset. This is accomplished by experimentation using a variety of machine learning techniques such as decision trees, naïve bayes, neural networks and random forest algorithms. Finally, the prediction models obtained using the selected machine learning techniques are then applied to unseen data in order to test the validity and accuracy of the model. The model performance can also be determined using other performance metrics such as kappa scores, precision, recall, F-measures, and Precision-recall curve values.

4.4 Demonstration

During this activity, the artifact(s) that were designed are applied to solve the identified problem from the first stage. This stage is usually in the form of the testing of the artifact to determine whether or to what extent the problem is solved. Examples of how demonstration is conducted is in the form of experiments, simulations, case studies or any other suitable activity [23]. For the larger study, experiments were executed using random forest algorithms, decision tree algorithms, and an optimized forest algorithm that used genetic algorithms. For all experiments, feature selection was used in order to identify the best features (attributes) to obtain the best performance measures.

4.5 Evaluation

For the evaluation stage, the main objective is to measure to what extent the artifact solves the given problem. This is accomplished by comparing the objectives of the study against observed results that were obtained during the Demonstration stage. The evaluation stage will require knowledge of evaluation techniques and the identification of relevant metrics. Examples of evaluation outputs are document detailing how the objectives of the research were met, the use of statistics or graphs to present quantitative evaluations, or the presentation of quantifiable measures such as response times of product and feedback.

Once all the steps are completed, the accuracy of the model (as well as other performance measures) will be determined in terms of its ability to predict student performance. The performance measures will also be compared to that of other learning analytics or educational data mining studies. In the case of the larger study, the performance measures (such as precision, recall, accuracy, F-measure etc.) obtained from the experiments are compared to that of performance measures obtained from identified learning analytics or educational data mining studies. This comparison is performed to determine if the prediction models generated are of an acceptable quality when compared to other studies performed at other educational institutions.

4.6 Communication

This is the final activity where the artifact details and its significance are communicated to the relevant stakeholders. Some of the details include the results of experiments conducted, a justification of the benefits of the artifact to researchers and other relevant stakeholders. The results described in this research paper forms the communication medium for the artifact, which is one of the major aspects identified in the larger study, its importance and necessity to the use of LA in higher education, the steps taken for the study in order to develop the artifact, and the application of the artifact within the problem domain.

It should be noted that the activities of the design science research model may not necessarily be executed in the order specified. The model does allow for iteration between the stages of development, demonstration, evaluation, and communication [23]. This observation was also noted by other authors that state that design science follows a pragmatic approach [15], specifically in the case of LA perspectives [13].

5 Discussion

As stated in the previous section, the process model artifact illustrates the learning analytics process from the beginning stage of data acquisition to performing prediction. The diagram, shown in Figure 6 below, is an adapted dataflow diagram with parallelograms representing techniques that have been applied where necessary.

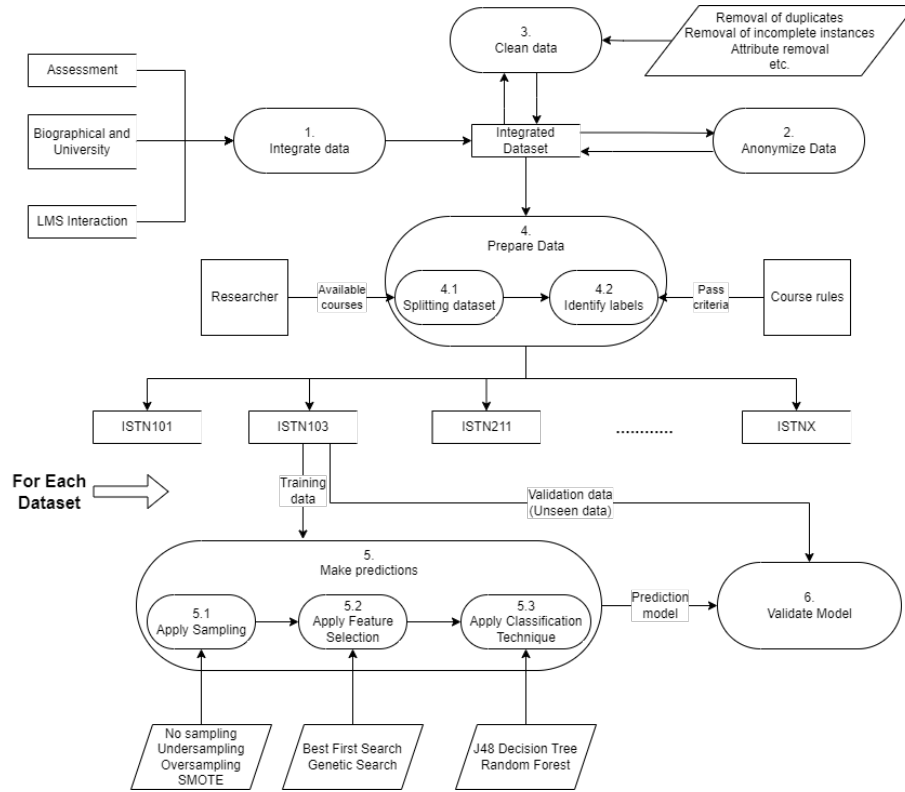


Fig. 5. Learning analytics process model (Artifact)

Starting at process 1 (Integrate data), data from the various data stores (in this case, assessments, biographical and university, LMS interaction) are fed into process 1 where the data is integrated into a single data store. While not illustrated in the model, the acquisition of the data is dependent on meeting the ethical and privacy policies followed by the institution conducting the learning analytics initiative.

Processes 2 and 3 involve the anonymization and cleaning of the integrated dataset. For the larger study, this involved the removal of duplicate rows of student data and attributes, removal of incomplete instances as well as categorizing of data items (data discretization).

The cleaned and anonymized dataset is then passed into process 4, where it is prepared for eventual analysis (prediction). The course rules dictate the pass criteria while the researcher specifies the courses required for prediction. During process 4, the integrated courses are divided based on the required courses resulting in a number of individual course datasets.

For each course dataset, the data is divided into training and validation (or test) datasets. The training data is sent to Process 5 (make predictions). During this process, sampling may be applied if required to deal with data imbalance, followed by feature selection and application of learning algorithms. The output of this process is in the form of a prediction model that forms the input of the final process of validation (process 6). Here the prediction model is applied to the unseen validation dataset. This is done to verify the performance of the prediction model in terms of how it predicts future unseen data instances.

The process model developed contributes to the area of predictive analytics, mainly in terms of better predictive analytics that will enable teaching staff to identify students in danger of performing poorly and try to assist them so that they may pass. Thus, the implementation of the learning analytics system forms the basis for improved identification of student weakness. A further benefit is that the use of feature selection may assist in identification of factors that influence student performance. By identifying these factors, stakeholders within the HEI can focus on these specific areas such as improved quality of learning resources (based on LMS interaction factors) or better resource allocation (such as improved support to students from poorer schools).

When comparing the above model to that of other learning analytics process models, it is noted that the processes are common for this model as well as that of those described in [5], [7] and [30]. The model developed in this study differs from that of other studies in the form of its presentation and additional application components that have been included.

Firstly, the use of a dataflow diagram provides emphasis on the conversion of raw data from the various data sources into the intended final form of information (in this case, predictions). Further to this, researchers or practitioners may use further levels of dataflow diagrams to better describe how each process converts the data input into output.

Secondly, while not a standard part of the dataflow diagram, the parallelograms provide an indication of which techniques have been implemented or fed into each process in order to convert the input data.

Thus, the presented model describes the learning analytics process that includes techniques involved in each process that were used to convert the input data (into the process) into the resultant data or information (out of the process).

6 Conclusion and future research

The study explored the use of design science in learning analytics to predict student academic performance within the information systems and technology discipline.

This was done at a South African university. While the concept of learning analytics was only conceived in 2011, it is now being seen as a necessary step in the advancement of teaching and learning at HEIs. This is crucial in the face of increasing student numbers, less support from governments, and rapid technological advancements.

Two main contributions are identified in this study. Firstly, while the process model described seems similar to that of other models, it differs with others in that it is designed using dataflow diagram notations, thus not only showing the common learning analytics process, but also how data is transformed into predictions. In addition, the model also includes an avenue for techniques based on the researcher requirements. Secondly, the study can formulate the design principles upon which its creation was based. These serve as propositions to be used in future work.

Since it is the first artifact of a larger study, its evaluation using formal methods is adjourned until future research. However, the artifact and the innovation inherent in it is presented so that academic managers and researchers know its capabilities.

References

1. Adejo, O., Connolly, T.: Predicting student academic performance using multi-model heterogeneous ensemble approach. *Predicting student academic performance using multi-model heterogeneous ensemble approach* 10(1), 61-75, (2018).
2. Alexandropoulis, S.N., Kotsiantis, S.B., Vrahatis, M.N.: Data preprocessing in predictive data mining. *The Knowledge Engineering Review* 34, (2019).
3. Baskerville, R., Pries-Heje, J., Venable, J.: Soft design science methodology. In the *Proceedings of the 4th International Conference on Design Science Research in Information Systems and Technology*, 1-11, Association for Computing Machinery, Philadelphia, USA (2009).
4. Bonnin, G., Boyer, A.: Higher Education and the Revolution of Learning Analytics. *International Council for Open and Distance Education*, Oslo, Norway (2017)
5. Campbell, J.P., DeBlois, P.B., Oblinger, D.G.: Academic analytics: A New Tool for a New Era. *EDUCAUSE Review* 42(4), 1-40, (2007).
6. Chatti, M.A., Dyckhoff, A.L., Schroeder, U., Thus, H.: A Reference Model for Learning Analytics. *International Journal of Technology Enhanced Learning* 4(5-6), 318-331, (2012).
7. Chatti, M.A., Muslim, A.: The PERLA Framework: Blending Personalization and Learning Analytics. *International Review of Research in Open and Distributed Learning* 20(1), 244-261, (2019).
8. Dawson, S., Joksimovic, S., Poquet, O., Siemens, G.: Increasing the impact of learning analytics. In *Proceedings of the 9th International Conference on Learning Analytics & Knowledge*, 446-455, Association for Computing Machinery, Tempe, USA (2019).
9. Ferguson, R., Clow, D., Macfadyen, L., Essa, A., Dawson, S., Alexander, S.: Setting Learning Analytics in Context: Overcoming Barriers to Large-State Adoption. *Journal of Learning Analytics* 1(3), 120-144, (2014).
10. Gasevic, D., Jovanovic, J., Pardo, A., Dawson, A.: Detecting Learning Strategies with Analytics: Links with Self-Reported Measures and Academic Performance. *Journal of Learning Analytics* 4(2), 113-128, (2017).

11. Ghorbani, R., Ghousi, R.: Comparing Different Resampling Methods in Predicting Students' Performance using Machine Learning Techniques. *IEEE Access* 8, 67899-67911, (2020).
12. Gibson, A., Lang, C.: The pragmatic maxim as learning analytics research method. In: the Eighth International Conference on Learning Analytics and Knowledge, 461-465, Sydney, Australia (2018).
13. Greller, W., Drachsler, H.: Translating Learning into Numbers: A Generic Framework for Learning Analytics. *Education Technology & Society* 15(3), (2012).
14. Hevner, A., March, S., Park, J., Ram, S.: Design science in information systems research. *MIS Quarterly* 28(1), 75-105 (2004).
15. Hevner, A., Chatterjee, S.: *Design Science in Information Systems: Theory and Practice*, Springer, New York (2010).
16. Kumar, M., Salal, Y.K.: Systematic Review of Predicting Student's Performance in Academics. *International Journal of Engineering and Advanced Technology* 8(3), 54-61, (2019).
17. Madasamy, K., Ramaswami, M.: Data Imbalance and Classifiers: Impact and Solutions from a Big Data Perspective. *International Journal of Computational Intelligence Research* 13(9), 2267-2281, (2017).
18. Mahrooian, H., Daniel, B., Butson, R.: The perceptions of the meaning and value of analytics in New Zealand higher education institutions. *International Journal of Educational Technology in Higher Education* 14(1), 1-17 (2017).
19. Marongwe, N., Kariyana, I., Mbodila, M.: Determinants of student dropout in Rural South African Universities. *Journal of Gender, Information and Development in Africa (JGIDA)*, 109-130, (2020).
20. Muljana, P.S., Placencia, G.: Learning analytics: Translating Data into "Just-in-Time" Interventions. *Scholarship of Teaching and Learning, Innovative Pedagogy* 1(1), 1-6, (2018).
21. Nguyen, A., Tuunanen, T., Gardner, L., Sheridan, D.: Design principles for learning analytics information systems in higher education. *European Journal of Information Systems* 30(5), 541-568 (2021).
22. Patwa, N., Seetharam, A., Sreekumar, K., Phani, S.: Learning Analytics: Enhancing the Quality of Higher Education. *Research Journal of Economics* 2(2), 1-7 (2018).
23. Peffers, K., Tuunanen, T., Rothenberger, M.A., Chatterjee, S.: A design science research methodology for information systems research. *Journal of Management Information Systems* 24(3), 45-77 (2007).
24. Prinsloo, P., Kaliisa, R.: Learning Analytics on the African Continent: An Emerging Research Focus and Practice. *Journal of Learning Analytics* 9(2), 218-235 (2022).
25. Rish, I. : An empirical study of the naive Bayes classifier, In: the International Joint Conference on Artificial Intelligence, 41-46, Seattle, USA (2001).
26. Romero, C., Ventura, S., Garcia, E.: Data mining in course management systems: Moodle case study and tutorial, *Computers & Education* 51(1), 368-384 (2008).
27. Romero, C., Romero, J.R., Ventura, S.: A survey on Pre-Processing Educational Data. *Educational Data Mining: Application and Trends*, Springer, (2014).
28. Romero, C., Ventura, S.: Educational data mining and learning analytics: An updated survey. *Wiley Interdisciplinary Review: Data Mining and Knowledge Discovery* 10(3), e1355 (2020).
29. Siemens, G., Gasevic, D., Haythornwaite, C., Dawson, S., Shum, S.B., Ferguson, R., Duval, E., Verbert, K., Baker, R.S.J.D.: *Open Learning Analytics: An Integrated & Modularized Platform*. Open University Press, Maidenhead (2011).

30. Siemens, G.: Learning Analytics: The emergence of a discipline., *American Behavioral Scientist* 57(10), 1380-1400, (2013).
31. Vaishnavi, V.K.: Design science research methods and patterns: innovating information and communication technology. 2nd edn. CRC Press, New York (2015).
32. Vaishnavi, V., Kuechler, B., Petter, S.: Design Science Research in Information Systems. (2004).
33. West, D., Heath, D., Huijser, H.: Let's Talk Learning Analytics: A Framework for Implementation in Relation to Student Retention., *Online Learning Journal* 20(2), 1-21, (2016).

Beyond the Hype: A Cautionary Tale of ChatGPT in the Programming Classroom

Grant Oosterwyk^[0000-0002-2745-3929], Pitso Tsibolane^[0000-0002-7888-1181],
Popyeni Kautondokwa^[0000-0001-9001-7313], and Ammar Canani

Department of Information Systems, Faculty of Commerce, University of Cape Town,
Cape Town, South Africa
{grant.oosterwyk;pitso.tsibolane}@uct.ac.za,
{KTNPOP001;CNNAMM001}@myuct.ac.za

Abstract. Due to the proliferation of Large Language Models research and the use of various Artificial Intelligence (AI) tools, the field of information systems (IS) and computer science (CS) has evolved. The use of tools such as ChatGPT to complete various student programming exercises (e.g., in Python) and assignments has gained prominence amongst various academic institutions. However, recent literature has suggested that the use of ChatGPT in academia is problematic and the impact on teaching and learning should be further scrutinized. More specifically, little is known about how ChatGPT can be practically used with code (programming) writing to complete programming exercises amongst IS and CS undergraduate university students. Furthermore, the paper provides insights for academics who teach programming to create more challenging exercises and how to engage responsibly in using ChatGPT to promote classroom integrity. In this paper, we used Complex Adaptive Systems (CAS) theory as a guideline to understand the various dynamics through demonstrations. Using ChatGPT, we analyzed various practical programming examples from past IS exercises and compared those with memos created by tutors and lecturers in a university. This paper highlights common ways of assessment, programming errors created by ChatGPT and the potential consideration for IS academics to ensure the development of critical programming skills among students.

Keywords: ChatGPT, Python, CAS, Exercises, Information Systems, Computer Science, University, Undergraduate students, South Africa.

1 Introduction

With the proliferation of Large Language Models (LLM) and their use in research and practice, many organisations are starting to leverage artificial intelligence (AI) tools such as ChatGPT for competitive advantage (Dwivedi et al. 2023). In various cases, the value of these tools gives way to enhancing productivity among practitioners. However, in the context of academia, the use of AI tools for educational purposes, specifically related to problem-solving and critical thinking (Mathee and Turpin, 2019) in software

programming, has become increasingly problematic in the information systems (IS) and computer science (CS) disciplines. LLMs, such as ChatGPT, can be defined as a human-text generator capable of scanning text patterns of large internet datasets using Natural Language Processing (NLP) (Rahman et al. 2023). A recent editorial paper has highlighted that ChatGPT is not limited to natural language (Dwivedi et al. 2023) and that the tool is able to handle various querying languages (e.g., Python, Java, C++ and others). The paper further recommended that tasks should be designed to be more challenging given the current design of many teaching pedagogies. Other studies (Shahrasbi, Jin, and Zheng, 2021) critically analyse traditional pedagogical methodologies employed in programming and application development courses, which tend to emphasize programming exercises while neglecting opportunities for meaningful student interaction. As a countermeasure, the researchers put forth an innovative instructional framework for a mobile application development course, predicated upon a design thinking paradigm that capitalizes on the active involvement of students within the design process. Hence, one of the objectives of this paper is to demonstrate how ChatGPT can be used to write basic programming tasks to complete various university programming assignments (i.e., exercises). The paper further aims to investigate the challenges associated with using ChatGPT for completing programming tasks using an IS foundation course and argues for the need to design more challenging assignments that foster critical programming skills among students. Furthermore, we employed the Complex Adaptive Systems (CAS) theory as a guiding framework to comprehend the multifaceted dynamics present in demonstrations. Using ChatGPT, we scrutinized various practical programming examples derived from past programming exercises and juxtaposed them with memos crafted by tutors and lecturers at a higher education institution. This paper elucidates prevalent assessment methods, programming inaccuracies generated by ChatGPT and potential considerations for IS educators to facilitate the cultivation of critical programming competencies in their students. The remainder of the paper is structured as follows. First, we provide survey-relevant literature related to AI, LLM and ChatGPT. Following this, we present the theoretical foundation that grounds this paper and offer an overview of the methodological approach of how we gathered and analysed our data. In the subsequent section, we provide our findings and explore what they mean in terms of theoretical and practical contributions. We conclude by recognising the limitations of our study and suggesting potential avenues for future research.

2 Related Work

In recent years, AI has received considerable attention from both academia and practitioners with discussions from different vantage points (Berente et al. 2021; Enholm et al. 2021; Chatterjee et al. 2022; Grøder et al. 2022; Dwivedi et al, 2023; Haque et al. 2023). With the increase in LLM placing humans at the core of an AI ecosystem necessitates providing systematic guidance on AI development from developers or engineers, as well as interfaces to elucidate the consequences of their programming decisions (Zhou et al. 2023).

Due to the lack of research regarding using ChatGPT for accomplishing programming tasks among students, this narrative review section will summarise the extant literature and describe the current state of knowledge on the phenomenon. Particularly, the use of LLM in IS and CS education. This will be followed by a discussion on the application of ChatGPT as an intelligent system for completing programming exercises and assignments and will highlight the benefits and limitations of using ChatGPT in IS and CS education. This will be followed by the importance of developing critical programming skills among students and the potential risks associated with relying solely on ChatGPT for completing programming exercises and assignments. The insights that follow are derived from a multitude of studies, collated to provide a narrative review encapsulating previous observations on the subject matter (Schryen et al. 2020). It should be emphasised that the methodological rigour typically associated with a systematic literature review has not been adhered to in this instance.

2.1 Large Language Modules (LLM) in IS and CS Education

A comprehensive analysis of all forms AI models has not been studied in IS and CS literature. Despite this, generative AI has received a decent amount of attention across various disciplines (Dwivedi et al. 2023). The introduction of several generative AI platforms in the past half-decade, such as GPT-3 in 2020 and ChatGPT in 2022, along with the broader category of LLM, has added significant interest from different vantage points (Mariani et al. 2023). Furthermore, others (Shah and Bender, 2022) have introduced a concern regarding the anonymity that comes with the use of search systems that employs machine learning (ML) techniques—specifically, the challenges regarding the origins of the information presented. The implementation of extensive LLM may exacerbate the lack of transparency and accountability within these systems which directly impacts the integrity of student academic tasks.

2.2 ChatGPT as an Intelligent System

A notable characteristic of generative AI platforms is their rapid and widespread adoption; for instance, ChatGPT, launched on November 30, 2022, amassed a user base of one million within the initial five days post-release (Dwivedi et al. 2023; Zhou et al. 2023). The primary goal of ChatGPT is to generate natural language text for various applications (Sobania et al. 2023). The reciprocal relationship between humans and AI fundamentally addresses the challenge of fostering trust in AI, particularly when it has not yet achieved full competence. By employing machines to enhance human cognitive abilities, humans have the potential to surpass their previous intellectual capacities. Additionally, AI can facilitate the expansion of human endeavours and render scarce human resources considerably more attainable. As an alternative, humans may delegate routine tasks to machines, relying on their efficiency in exchange for the opportunity to invest time and effort in more engaging, creative, or specialised activities. In both scenarios, machines can assist humans in amassing information and presenting various options while ultimately deferring to human users for the final decision-making (Dwivedi et al. 2023; Mikalef et al. 2023). Despite this, an overreliance on ChatGPT

could introduce serious limitations to critical thinking and problem-solving, including but not limited to, IS and CS students. This will be discussed in the following section.

2.3 Limitations of ChatGPT

Though the internal mechanisms of ChatGPT remain unclear, OpenAI recognises certain limitations and researchers continue to engage with and scrutinize it to uncover further constraints (Else, 2023). Certainly, the impact on higher education has become a subject of debate, especially among IS and CS academics (Greene et al. 2022; Teubner et al. 2023, Dwivedi et al. 2023). Others argue that such tools can expedite knowledge acquisition, potentially benefiting the learning process. However, these platforms also present concerns regarding plagiarism and academic integrity, as students may employ them to complete assessments and dissertations (Stokel-Walker, 2022). Additionally, long-term consequences may arise if students opt to forgo comprehensive and critical engagement with a topic, relying instead on ChatGPT for quick and potentially superficial learning and coding practices. More technical limitations will be provided in the experimental section.

2.4 Benefits of ChatGPT

As mentioned, ChatGPT is not limited to natural language (Dwivedi et al. 2023) and the tool is able to handle various querying languages. For IS and CS programming students, OpenAI has now introduced various platforms that can suggest code and entire functions in real-time (e.g., GitHub Copilot, Codex). Despite this, it positively influences students and practitioners with low AI literacy or numerical skills (Zhou et al. 2023) as it provides the opportunity to invest time and effort in more engaging, creative, or specialized activities, especially for those students who do not major in these fields (e.g., Humanities). One major benefit of ChatGPT is the interaction with the system that allows for in-depth dialogue conversations to avoid misleading textual descriptions (Sobania et al. 2023). ChatGPT could be programmed to provide hints or suggestions that help programmers break down problems into more manageable parts. Additionally, ChatGPT could be designed to recognize common programming patterns and offer suggestions for applying them in new contexts, similar to how experts use their mental plans to solve problems. More technical benefits will be provided in the experimental section.

2.5 Development of Critical Programming Skills

Various academic institutions have called for more computational thinking to improve the transferability of programming skills in multiple courses (Matthee and Turpin, 2019). Despite Python being widely taught in both IS and CS courses, there is a notable lack of research and teaching guidelines in IS specifically focusing on problem-solving compared to CS. Others recommend that ChatGPT can be used in programming as a knowledge-creation tool for initial brainstorming (Megahed et al. 2023). The authors offered three distinct tasks encompassing (1) *the comprehension of code*, (2) *the*

exploration of alternative methodologies for problem resolution through programming and (3) *the translation of code between disparate programming languages* (Megahed et al. 2023). The study further suggests that ChatGPT would alleviate the challenges among students associated with acquiring proficiency in programming languages, while simultaneously facilitating the sharing of code across various programming languages. A previous paper (De Raadt, Watson, and Toleman, 2006) advocates for the use of the Soloway (1986) model, to develop a tacit body of programming strategies to meet goals. For example, the model suggests that breaking down complex problems into smaller sub-problems is an effective problem-solving strategy.

2.6 Responsible Use of Generative AI (ChatGPT)

A recent paper (Dwivedi et al. 2023, p. 6), explores four themes under the umbrella of “Exploring the ethics of responsible AI: lessons from utilitarianism.”: Theme 1: *a tool view*: the aim to improve the clarity of generative and self-adaptive AI models, to facilitate the explainability of resulting outputs (Haque et al. 2023); Theme 2: *a proxy view*: aims to understand ethical policies across organisations; Theme 3: *a ensemble view*: aims to understand under what conditions ChatGPT should be implemented; Theme 4: *a skills view*: aims to understand the skills and resources required to investigate the key limitations of ChatGPT across various contexts. Further studies suggested that the ongoing responsibility of academics is to foster critical thinking skills in students, a goal that has remained consistent over time. They also highlighted the potential for applications like ChatGPT to serve as companions or tutors in support of this objective (Dwivedi et al. 2023; Matthee and Turpin, 2019).

3 Theoretical approach

Complexity theory emerged within the domain of pure mathematics and was applied within the natural sciences to conceptualise better the non-linear and unpredictable nature of complex systems and the importance of feedback loops and emergence (McMillan, 2008; Stacey, 2003). More recently, the social sciences have increasingly adopted and adapted complexity theory to describe various social systems as complex systems (Moore et al., 2019; Brainard and Hunter, 2016; Hawe et al., 2009; Howarth et al., 2016; Moore and Evans, 2017; Rutter et al., 2017;). Within the field of education, complexity theory has been applied as a framework to understand the behaviour of complex systems such as the classroom environment (Jacobson et al. 2020).

Complex Adaptive Systems (CAS) theory (See Fig. 1) is a branch of complexity theory that provides a novel framework for conceiving systems as constituted by autonomous agents that interact with each other following predetermined rules (Onik, Fieft & Gable, 2017). It facilitates an inquiry into the adaptive nature of complex systems and the emergence of order or properties that stem from the interactions of its constituent elements (Holland, 1995; Vidgen & Wang, 2006). The core components of a CAS include a) agents (individuals or entities that interact with each other), b)

networks or connections between agents, and c) feedback loops or mechanisms that allow agents to learn and adjust their behaviour (Holland, 2006).

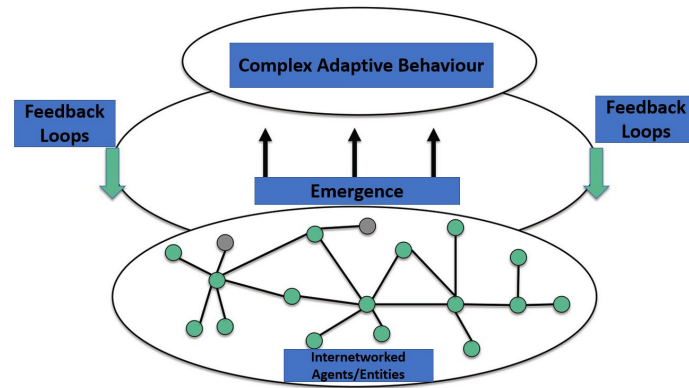


Fig. 1. Complex Adaptive System (adapted from van der Leeuw, 2020)

This research views the Information Systems programming classroom as a teaching and learning practice complex system i.e., students, lecturers/educators, programming tasks and the AI-driven technology (ChatGPT 3.5) as agents and entities interacting with each other within the classroom through the teaching and learning process of executing programming tasks. The interactions between individual entities (e.g., students, programming tasks, ChatGPT3.5) at the lower level create patterns in the classroom that emerge and are observable at the higher level (course outcomes), which, in turn, impact the interactions between individual entities. Onik, Fiel and Gable (2017) identified seven foundational constructs of the CAS theory, namely, 1) co-evolution, 2) emergence, 3) self-organisation, 4) fitness landscape, 5) the edge of chaos, 6) dynamism and non-linearity, 7) adaptation.

4 Methodology

The study adopted a pragmatist approach (Rorty, 2004) drawing on empirical evidence from real-world Python programming tasks from an introductory programming course aimed at first-year university students. Pragmatism emphasizes practicality, usefulness, and effectiveness and posits that ideas, theories, and beliefs should be evaluated based on their practical consequences and how they can be applied in real-world situations (Denzin, 2010; Rorty, 2004). Furthermore, pragmatism highlights the importance of experimentation, observation, and experience in evaluating the effectiveness of different approaches to solving problems.

To ensure rigour, the paper followed the pragmatic qualitative analysis process suggested by Ramanadhan et al. (2021). Firstly, a research question that addresses a practical problem was formulated, and then secondly, empirical evidence derived from our real-world situation was gathered. Thirdly, pragmatism suggests that ideas, theories, and beliefs should be evaluated based on their practical consequences. Fourthly, we

applied our own experiences and the experiences of the participating students to gain insights into the implications of using ChatGPT in a programming classroom. Finally, pragmatism encourages an iterative approach to problem-solving. We conducted multiple iterations of different Python programming exercises to observe, analyse, explore, and refine our understanding of the implications of AI-led programming. Guided by the CAS constructs, we qualitatively and critically examine the dynamic and emergent interactions, pedagogical consequences and feedback loops due to the interaction of students of programming and ChatGPT (an AI-driven technology). We present several programming tasks that students are required to complete as part of learning new programming skills. We model how the freely available AI-driven application, ChatGPT 3.5, processes those tasks and discuss the consequent complex adaptive behaviours produced by this interaction.

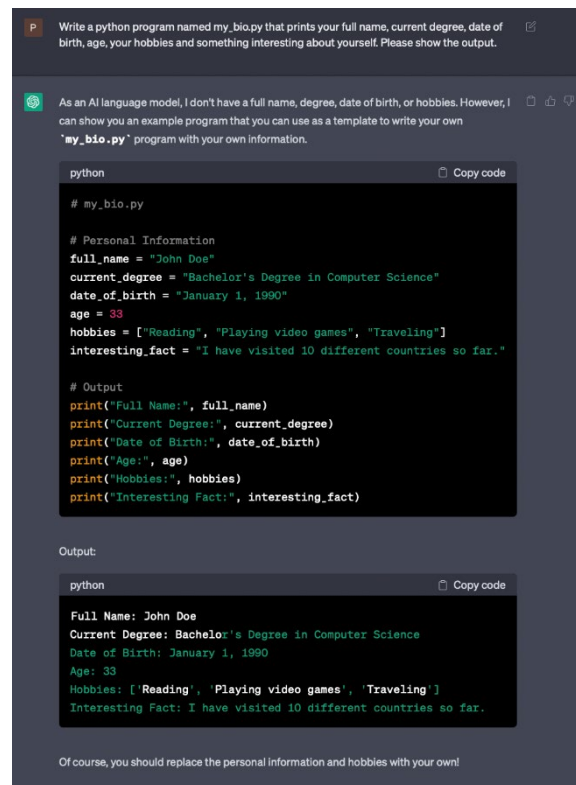
5 Findings

In this section, we requested ChatGPT to complete various programming tasks which are part of the Python exercises typically required for students to complete. The prompt and corresponding output from ChatGPT is displayed. ChatGPT has been asked to complete programming tasks for activity one based on questions in Fig. 2. The prompt and the output from ChatGPT are presented as snapshots (zooming is required). The snapshots and responses represent the practical classroom interaction students can make while completing their programming activities.

No.	Name of Program	Output
1	escape_it.py	Python's syntax is nice. It's like writing in English. It is like they say it isn't 'hectic'
2	product.py	Enter the first number: 2 Enter the second number: 4 Product of 2 and 4 is: 8
3	pass_fail.py	What is your course mark? 50 Passed
4	age.py	What is your age? 20 Adult
5	slicing.py	Enter your first name: Jane Enter your last name: Doe jan.doe
6	biggest.py	Enter the first number: 1 Enter the second number: 3 Enter the third number: 2 3

Fig. 2. The six programming tasks for the first Python class activity.

In the next stage, see Fig. 3, the researchers prompted ChatGPT to complete the fourth programming activity and provided the output (Fig. 3). A Python file “my_bio.py” was created for this task by the chatbot.



The screenshot shows a ChatGPT chat window. The user prompt is: "Write a python program named my_bio.py that prints your full name, current degree, date of birth, age, your hobbies and something interesting about yourself. Please show the output." The ChatGPT response is: "As an AI language model, I don't have a full name, degree, date of birth, or hobbies. However, I can show you an example program that you can use as a template to write your own 'my_bio.py' program with your own information." Below this, there is a code block for a Python program named my_bio.py. The code defines variables for personal information and hobbies, and then prints them out. The output of the program is shown below the code block.

```
python
# my_bio.py

# Personal Information
full_name = "John Doe"
current_degree = "Bachelor's Degree in Computer Science"
date_of_birth = "January 1, 1990"
age = 33
hobbies = ["Reading", "Playing video games", "Traveling"]
interesting_fact = "I have visited 10 different countries so far."

# Output
print("Full Name:", full_name)
print("Current Degree:", current_degree)
print("Date of Birth:", date_of_birth)
print("Age:", age)
print("Hobbies:", hobbies)
print("Interesting Fact:", interesting_fact)
```

Output:

```
python
Full Name: John Doe
Current Degree: Bachelor's Degree in Computer Science
Date of Birth: January 1, 1990
Age: 33
Hobbies: ['Reading', 'Playing video games', 'Traveling']
Interesting Fact: I have visited 10 different countries so far.
```

Of course, you should replace the personal information and hobbies with your own!

Fig. 3. ChatGPT output: prompt for the third Python task of activity one.

In Fig. 4, the researchers asked ChatGPT to complete the first programming activity by creating a “hello world” Python file and to print the output (on the right-hand side of Fig. 4). ChatGPT is prompted to complete the second programming activity, with an output displayed below in the form of an ASCII image (on the left-hand side of Fig. 4). The chatbot created the “ascii_art.py” Python file. ChatGPT presented a random smiley face as an output as it cannot read the web link provided to create the image. This presented a limitation.

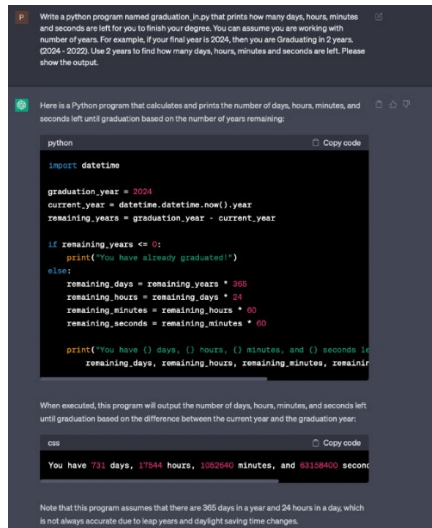


Fig. 5. ChatGPT output: prompt for the fifth Python task of activity one.

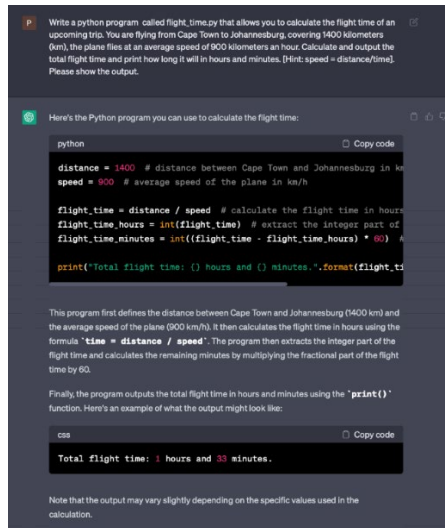


Fig. 6. ChatGPT output: prompt for the sixth Python task of activity one.

In Fig. 7, the expected output for all the tasks for the first programming activity are shown, the prompted task closely resembled the expected outcome for each of these tasks.

No.	Name of Program	Output
1	hello_world.py	Hello World
2	my_bio.py	Name: John Doe Current Degree: Bachelors in Commerce Specializing in Information Systems Date of Birth: 31 December 2000 Age: 22 Hobbies: Espresso and Chill Something interesting about myself: I hate questions such as these!
3	ascii_art.py	
4	my_bio_2.py	Name: John Doe Current Degree: Bachelors in Commerce Specializing in Information Systems Date of Birth: 31 December 2000 Age: 22 Hobbies: Espresso and Chill Something interesting about myself: I hate questions such as these!
5	graduation_in.py	Days Left: 730 Hours Left: 17520 Minutes Left: 1051200 Seconds Left: 63072000
6	flight_time.py	It will take: 1.0 hour(s) and 33.333333333333336 minute(s) to arrive.

Fig. 7. The expected course output for Python activity one.

ChatGPT has been asked to complete programming tasks for activity two based on questions in Fig. 8. The prompts and outputs are displayed below.

No.	Name of Program	Question
1	escape_it.py	Write a program that prints "Python's syntax is nice. It's like writing in English. It is like they say it isn't 'hectic'".
2	product.py	Write a program that asks the user to enter 2 numbers. The program then prints the sum and product of the 2 numbers.
3	pass_fail.py	Write a program that asks the user to enter their course mark. Print "Passed" if the mark is 50 or higher, otherwise print "Failed"
4	age.py	Write a program that asks the user to enter their age. Print "baby" if the age is between 0 and 4 inclusive, print "child" if the age is between 5 and 12 inclusive, print "teenager" if the age is between 13 and 19 inclusive and print "adult" if the age is 20 or older.
5	slicing.py	Write a program that asks the user to enter the first name and then their last name. Print the first 3 characters of the first name, a dot (".") and the last 3 characters of the last name. All of the characters need to be lower-case when you print them. E.g., If the name is John Doe, you will print: joh.doe
6	biggest.py	Write a program that asks the user to enter 3 numbers, print the biggest number.

Fig. 8. Instructions for tasks of the second Python activity.

ChatGPT is prompted to complete the first activity of the second programming activity, the prompt and the output is displayed in Fig. 9. The chatbot created the "escape_it.py" Python file and printed the output.

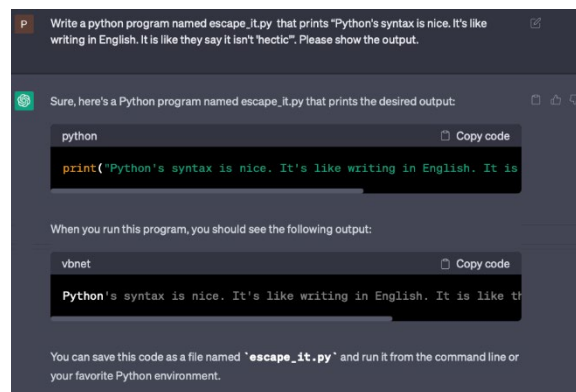


Fig. 9. ChatGPT output: prompt for the first Python task of the second activity.

In the snapshot in Fig. 10, ChatGPT is prompted to complete the second task of programming activity two and provides the output. A Python file for this task was created by the chatbot. ChatGPT also provides context on the functions used in the coding of the program, this provides benefits for students learning the specific programming language, this is an example of a positive feedback loop.

P Write a python program called product.py that asks the user to enter 2 numbers. The program then prints the sum and product of the 2 numbers. Please show the output.

G Sure, here's a Python program called "product.py" that asks the user to enter two numbers, then calculates and prints the sum and product of the two numbers:

```
python
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))

# calculate the sum and product
sum = num1 + num2
product = num1 * num2

# print the results
print("The sum of", num1, "and", num2, "is", sum)
print("The product of", num1, "and", num2, "is", product)
```

When you run this program, you should see output like the following:

```
python
Enter the first number: 3.5
Enter the second number: 4.2
The sum of 3.5 and 4.2 is 7.7
The product of 3.5 and 4.2 is 14.7
```

Note that the program prompts the user to enter two numbers, then converts the input to floating-point values using the "float()" function. It then calculates the sum and product of the two numbers using basic arithmetic operations and stores the results in the "sum" and "product" variables, respectively. Finally, it uses the "print()" function to output the results to the console in a human-readable format.

Fig. 10. ChatGPT output: prompt for the second Python task of the second activity.

Fig. 11 shows a snapshot of the ChatGPT prompt to complete the third task of programming activity two and provides the output. The chatbot created a Python file for this task.

In the snapshot in Fig. 12, ChatGPT is prompted to complete the fourth task of programming activity two and provides the output. The chatbot created a Python file for this task. This output is another demonstration of ChatGPT providing positive feedback as it shows students how it utilised functions in its output.

P Write a python program named pass_fail.py that asks the user to enter their course mark. Print "Passed" if the mark is 50 or higher, otherwise print "Failed". Please show the output.

G Sure! Here's a Python program that asks the user for their course mark and prints "Passed" if the mark is 50 or higher, otherwise "Failed":

```
python
mark = float(input("Enter your course mark: "))

if mark >= 50:
    print("Passed")
else:
    print("Failed")
```

Here's an example output for a mark of 70:

```
css
Enter your course mark: 70
Passed
```

And here's an example output for a mark of 40:

```
css
Enter your course mark: 40
Failed
```

Fig. 11. ChatGPT output: Prompt for the third Python task of the second activity.

P Write a python program named age.py that asks the user to enter their age. Print "baby" if the age is between 0 and 4 inclusive, print "child" if the age is between 5 and 12 inclusive, print "teenager" if the age is between 13 and 19 inclusive and print "adult" if the age is 20 or older. Please show the output.

G Sure, here's a Python program that asks the user to enter their age and then prints a message based on their age:

```
python
age = int(input("Enter your age: "))

if age >= 0 and age <= 4:
    print("baby")
elif age >= 5 and age <= 12:
    print("child")
elif age >= 13 and age <= 19:
    print("teenager")
else:
    print("adult")
```

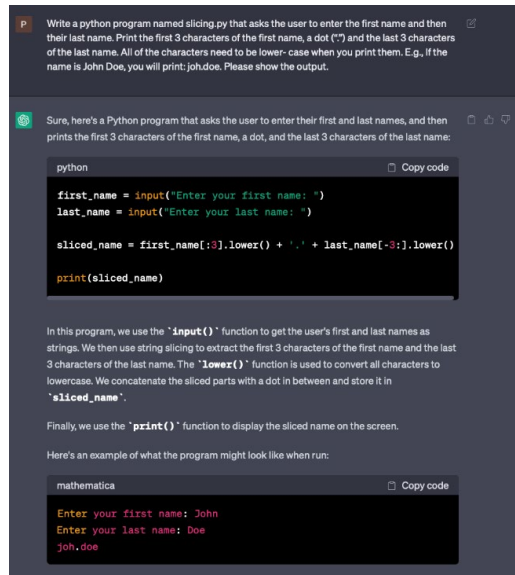
In this program, we use the "input()" function to get the user's age as a string, and then use the "int()" function to convert it to an integer. We then use a series of "if" and "elif" statements to determine the appropriate message to print based on the user's age. Finally, we use the "print()" function to display the message on the screen.

Here's an example of what the program might look like when run:

```
mathematics
Enter your age: 20
adult
```

Fig. 12. ChatGPT output: prompt for fourth Python instruction as part of the second activity.

The snapshot in Fig. 13 displays the prompt in ChatGPT of the second task of programming activity two and provides the output with positive feedback, demonstrating more benefits of ChatGPT to students learning this programming language.



The screenshot shows a ChatGPT conversation. The user prompt (P) asks for a Python program named 'slicing.py' that takes a first and last name, slices the first 3 characters of each, and concatenates them with a dot. The assistant (G) responds with a Python code block and an explanation. The code uses `input()` for user input, string slicing (`[0:3]`), `lower()` for case conversion, and `print()` for output. An example run shows the input 'John Doe' resulting in the output 'joh.doe'.

```
python
first_name = input("Enter your first name: ")
last_name = input("Enter your last name: ")

sliced_name = first_name[:3].lower() + '.' + last_name[-3:].lower()

print(sliced_name)
```

In this program, we use the `input()` function to get the user's first and last names as strings. We then use string slicing to extract the first 3 characters of the first name and the last 3 characters of the last name. The `lower()` function is used to convert all characters to lowercase. We concatenate the sliced parts with a dot in between and store it in `'sliced_name'`.

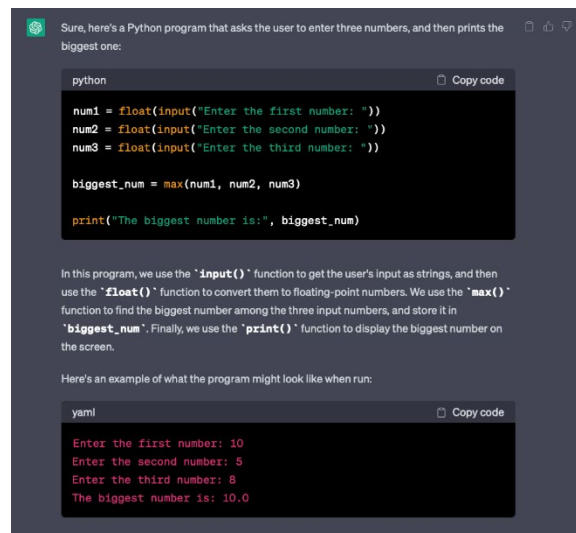
Finally, we use the `print()` function to display the sliced name on the screen.

Here's an example of what the program might look like when run:

```
mathematica
Enter your first name: John
Enter your last name: Doe
joh.doe
```

Fig. 13. ChatGPT output: Prompt for the fifth Python task of the second activity.

In the snapshot in Fig. 14, ChatGPT is prompted to complete the sixth task of programming activity two and provides the output with positive feedback, further demonstrating its benefits to students.



The screenshot shows a ChatGPT conversation. The assistant (G) provides a Python program that asks the user to enter three numbers and prints the biggest one. The code uses `float(input())` to get and convert the numbers, and the `max()` function to find the largest value. An example run shows inputs 10, 5, and 8 resulting in the output 'The biggest number is: 10.0'.

```
python
num1 = float(input("Enter the first number: "))
num2 = float(input("Enter the second number: "))
num3 = float(input("Enter the third number: "))

biggest_num = max(num1, num2, num3)

print("The biggest number is:", biggest_num)
```

In this program, we use the `input()` function to get the user's input as strings, and then use the `float()` function to convert them to floating-point numbers. We use the `max()` function to find the biggest number among the three input numbers, and store it in `'biggest_num'`. Finally, we use the `print()` function to display the biggest number on the screen.

Here's an example of what the program might look like when run:

```
yaml
Enter the first number: 10
Enter the second number: 5
Enter the third number: 8
The biggest number is: 10.0
```

Fig. 14. ChatGPT prompt for the sixth Python task of second activity with the output.

In Fig. 15, the expected output to all the tasks for the second programming activity are displayed, the prompted tasks closely resembled the expected outcome for each task.

No.	Name of Program	Output
1	escape_it.py	Python's syntax is nice. It's like writing in English. It is like they say it isn't 'hectic'
2	product.py	Enter the first number: 2 Enter the second number: 4 Product of 2 and 4 is: 8
3	pass_fail.py	What is your course mark? 50 Passed
4	age.py	What is your age? 20 Adult
5	slicing.py	Enter your first name: Jane Enter your last name: Doe jan.doe
6	biggest.py	Enter the first number: 1 Enter the second number: 3 Enter the third number: 2 3

Fig. 15. Python exercise output: Activity 2.

6 Discussion

A key observation emanating from a typical interaction between the student and ChatGPT is that the AI-driven application provides perfect solutions to the problems presented through merely copying and pasting from the activity sheet. The technology agent, therefore, takes over the expected learning, effort, and experience that the student is expected to undergo. This observation presents an emergent and existential conundrum: If the machine is learning, is the student learning?

In CAS, co-evolution refers to the dynamic process where two or more agents evolve together in response to each other's actions. The co-evolutionary process in a classroom can lead to the development of complex adaptations that allow students to interact with AI technology (ChatGPT) in more complex and sophisticated ways. In our case, while the machine (Agent B) evolves in its better understanding of answering the student, there is a likelihood that the student is developing dependency and disempowerment as they lose out on the opportunity to develop practical programming skills. Emergence in CAS refers to the phenomenon where complex patterns, structures, or behaviours spontaneously arise from the interactions between many simple agents within the system. Emergence in this study refers to new behaviours such as dependency (over-reliance) on the AI tool and loss of competency development that can arise when students rely on technology instead of internalising the learning.

Self-organisation in CAS is the potential to spontaneously form patterns or structures without being directed by any external force or central authority. This emergent behaviour arises from the interactions between the system's individual components. In the interaction with the student and ChaptGPT, while the AI might tend towards self-organising during its interaction with Agent A - the student, it is not evident that the student will grow from this interaction. Due to the pressure and demands of university studies, when students obtain answers for the work assigned to them, they are likely to consider the work to be complete and move on to the next urgent tasks. Tasks such as the ones shown in the previous section are aimed at slowing the student down to spend time engaging with the material. However, this opportunity is lost when they have a powerful answer generator to rely on. The edge of chaos in CAS, applied to our context, is the state where the teaching and learning practice is balanced between order (undisrupted teaching and learning) and chaos (disrupted teaching and learning). In this state of order (the edge of chaos), the system has enough order to be stable and predictable but enough chaos to allow for flexibility and adaptation to changing conditions. We propose that introducing the ChatGPT agent generates a bias towards chaos as the student is deprived of achieving and experiencing the intended learning outcomes. Dynamism in CAS refers to the system's ability to respond to the environment in real time and to continuously adjust behaviour and structure in response to changing conditions. Non-linearity specifies that small changes in one part of a CAS can lead to significant, unpredictable effects throughout the system. Together, dynamism and non-linearity posit the classroom interaction between the students and the AI (ChatGPT) as highly complex and challenging to predict or control. We find that educators (Agent C) will experience high levels of difficulty in monitoring student progress. This presents a significant challenge to the teaching and learning process. The dynamic and non-linear interactions between the three primary agents leads to the emergence of unintended negative complex adaptive behaviors overall.

7 Future Work and Conclusion

As ChatGPT becomes more widely utilised for activities like programming there will be a heightened emphasis on accountability when teaching and learning issues arise. The increasing reliance on ChatGPT has the potential to lead to a shift in educational expectations and outcomes, as pressure to rethink traditional teaching methods and pedagogies develops. This will have profound implications for common assessment practices, marking guidelines and academic recommendations for IS and CS education. From the examples this paper covered, while the introduction of ChatGPT has the potential to provide benefits such as positive feedback loops (providing context to programming solutions), it also generates concerning negative feedback loops of over-reliance and disempowerment. With OpenAI's API being widely used for code generation, students can now produce code without any deep computational knowledge, creating a new wave of ChatGPT-assisted programmers and developers ("prompt engineers") which might lead to unintended consequences such as over-reliance on AI and the decline of computational skills development. This raises questions about the impact

on performative abilities and the value of entering exam locations without prior knowledge, as ChatGPT is mainly used for assignments and tutorials. Consequently, this trend may affect student results and lead to potential demand for "master intelligent prompters" as a new industry skill. However, this reliance on ChatGPT could erode essential IS and CS skills, as ChatGPT may generate incorrect code or negatively impact programming projects. This research contributes a cautious reflection to the emergent discourse regarding the use of LLMs in the classrooms, in particular, the programming classrooms. Future work for this work should involve longitudinal observational studies that would compare, through mixed method inquiry, how the computational thinking skills of students develop over time as various AI-driven programming tools become available to students.

References

1. Berente, N., Bin Gu, Recker, J., Santhanam, R.: Managing Artificial Intelligence. *MIS Quarterly*. 45(3), 1433–1450 (2021).
2. Brainard J. Hunter PR.: Do complexity-informed health interventions work? A scoping review. *Implementation Science*. 11(1) 1–11 (2016).
3. Chatterjee, S., Chaudhuri, R., & Mikalef, P.: Examining the Dimensions of Adopting Natural Language Processing and Big Data Analytics Applications in Firms. *IEEE Transactions on Engineering Management*. (2022).
4. De Raadt, M., Toleman, M., & Watson, R.: Chick sexing and novice programmers: explicit instruction of problem-solving strategies. In *Proceedings of the 8th Australasian Computing Education Conference (ACE)* (52), 55–62 (2006).
5. Denzin, N.: Moments, mixed methods, and paradigm dialogs. *Qualitative Inquiry*. 16, 419–427 (2010).
6. Dwivedi, Y. K., Kshetri, N., Hughes, L., Slade, E. L., Jeyaraj, A., Kar, A. K., Wright, R.: So what if ChatGPT wrote it? Multidisciplinary perspectives on opportunities, challenges and implications of generative conversational AI for research, practice and policy. *International Journal of Information Management*. 71, 102642 (2023).
7. Else, H.: Abstracts written by ChatGPT fool scientists. *Nature*. 613(7944), 423–423 (2023)
8. Enholm, I. M., Papagiannidis, E., Mikalef, P., & Krogstie, J.: Artificial intelligence and business value: A literature review. *Information Systems Frontiers*. 24(5), 1709–1734 (2022).
9. Greene, T., Shmueli, G., Ray, S.: Taking the Person Seriously: Ethically-aware IS Research in the Era of Reinforcement Learning-based Personalization. (2022).
10. Grøder, C. H., Schmager, S., Parmiggiani, E., Vasilakopoulou, P., Pappas, I., & Papavaslopoulou, S.: Educating about Responsible AI in IS: Designing a course based on Experiential Learning. In *Proceedings of ICIS*. (2022).
11. Haque, A. B., Islam, A. N., & Mikalef, P.: Explainable Artificial Intelligence (XAI) from a user perspective: A synthesis of prior literature and problematizing avenues for future research. *Technological Forecasting and Social Change*. (186), 122120 (2023).
12. Hawe P, Shiell A., Riley T.: Theorising interventions as events in systems. *American Journal of Community Psychology*. 43(3), 267–76 (2009).
13. Holland, J.H.: *Emergence: From Chaos to Order*. Addison-Wesley, Boston (1998).
14. Howarth E, Devers K, Moore G.: Contextual issues and qualitative research. In: Raine RFR, Barratt H, Bevan G, et al. (eds) *Challenges, Solutions and Future Directions in the*

Evaluation of Service Innovations in Health Care and Public Health. *Health Service Delivery Research*. 105–20 (2016).

15. Jacobson, M. J., Levin, J. A., & Kapur, M.: Education as a complex system: Conceptual and methodological implications. *Educational researcher*. 48(2), 112–119 (2019).
16. Mariani, M. M., Machado, I., Magrelli, V., & Dwivedi, Y. K.: Artificial intelligence in innovation research: A systematic review, conceptual framework, and future research directions. *Technovation*. 102623 (2022).
17. Matthee, M. & Turpin, M.: Invited Paper: Teaching Critical Thinking, Problem Solving, and Design Thinking: Preparing IS Students for the Future. *Journal of Information Systems Education*. 30(4), 242252 (2019).
18. McMillan, E.: *Complexity, management and the dynamics of change: Challenges for practice*. Routledge. (2008).
19. Megahed, F. M., Chen, Y. J., Ferris, J. A., Knoth, S., & Jones-Farmer, L. A.: How generative ai models such as chatgpt can be (mis) used in spc practice, education, and research? an exploratory study. *arXiv preprint arXiv:2302.10916*. (2023).
20. Mikalef, P., Lemmer, K., Schaefer, C., Ylinen, M., Fjørtoft, S. O., Torvatn, H. Y., ... & Niehaves, B.: Examining how AI capabilities can foster organizational performance in public organizations. *Government Information Quarterly*, 101797 (2023).
21. Mollick, E.: ChatGPT is a tipping point for AI. *Harvard Business Review*. (2022).
22. Moore G, Audrey S, Barker M.: Process evaluation of complex interventions: Medical Research Council guidance. *British Medical Journal*. (350) (2015).
23. Moore, G. F., Evans, R. E., Hawkins, J., Littlecott, H., Melendez-Torres, G. J., Bonell, C., & Murphy, S.: From complex social interventions to interventions in complex social systems: future directions and unresolved questions for intervention development and evaluation. *Evaluation*. 25(1), 23–45 (2019).
24. Onik, M. F. A., Fielt, E., & Gable, G.: Complex adaptive systems theory in information systems research: A systematic literature review. In *Proceedings of the 21st Pacific Asia Conference on Information Systems*. Association for Information Systems (AIS) (2017).
25. Rahman, M., Terano, H. J. R., Rahman, N., Salamzadeh, A., & Rahaman, S.: ChatGPT and Academic Research: A Review and Recommendations Based on Practical Examples. *Journal of Education Management and Development Studies*. 3(1), 1–12 (2023).
26. Ramanadhan S., Revette A.C., Lee R. M., Aveling E.: Pragmatic approaches to analyzing qualitative data for implementation science: an introduction. *Implementation Science Communications*. 2(1),1–10 (2021).
27. Rorty, R., Putnam, H., Conant, J., & Helfrich, G.: What is Pragmatism? *Think: Philosophy for Everyone*. 8(1), 71–88 (2004).
28. Rutter, H., Savona, N., Glonti, K., Bibby, J., Cummins, S., Finegood, D.T., Greaves, F., Harper, L., Hawe, P., Moore, L., Petticrew, M.: The need for a complex systems model of evidence for public health. *The lancet*. 390(10112), 2602–2604 (2017).
29. Schryen, G., Wagner, G., Benlian, A., Paré, G.: A knowledge development perspective on literature reviews: validation of a new typology in the IS field. *Communications of the Association for Information Systems*. 46(7), 134–186 (2020).
30. Shah, C., & Bender, E.M.: Situating Search. In *ACM SIGIR Conference on Human Information Interaction*. 221–232 (2022).
31. Shahrasbi, N., Jin, L., & Zheng, W.-J.: Teaching Tip: Design Thinking and Mobile App Development: A Teaching Protocol. *Journal of Information Systems Education*. 32(2), 92–105 (2021).
32. Sobania, D., Briesch, M., Hanna, C., & Petke, J.: An analysis of the automatic bug fixing performance of ChatGPT. *arXiv* (2023).

33. Soloway, E.: Learning to program = learning to construct mechanisms and explanations. *Communications of the ACM*. 29(9), 850–858 (1986).
34. Stacey, R.: *Strategic Management and Organisational Dynamics: The Challenge of Complexity to Ways of Thinking About Organisations*, (Fifth ed.). Essex, England: Pearson Education (2007).
35. Stokel-Walker, C.: AI bot ChatGPT writes smart essays - should professors worry? *Nature* (London) (2022).
36. Van der Leeuw, S.: *Social sustainability, past and future: undoing unintended consequences for the Earth's survival*. Cambridge University Press (2020).
37. Vidgen, R., Wang, X.: Organizing for Agility: A Complex Adaptive Systems Perspective on Agile Software Development Process, in *Proceedings of the 14th European Conference on Information Systems (ECIS)*, Gothenberg, Sweden (2006).
38. Zhou, L., Rudin, C., Gombolay, M., Spohrer, J., Zhou, M., & Paul, S.: From Artificial Intelligence (AI) to Intelligence Augmentation (IA): Design Principles, Potential Risks, and Emerging Issues. *AIS Transactions on Human-Computer Interaction*. 15(1), 111–135 (2023).

Towards Using African Thought to Decolonise the Curriculum— A Scientific Perspective

Matthew O. Adigun¹[0000-0001-6256-5865] and Skhumbuzo Zwane²[0009-0008-0760-4444]

University of Zululand, Private Bag X1001, KwaDlangezwa 3886, South Africa

¹AdigunM@unizulu.ac.za

²ZwaneSG@unizulu.ac.za

Abstract. The curriculum decolonisation concept is gaining ground in tertiary education circles worldwide. Many attempts in Engineering and Sciences, have demonstrated how to inject hitherto extraneous perspectives into an existing or brand-new curriculum, which in most cases provided the opportunity for students to take personal responsibility for their learning. At the University of Zululand, our node-for-African-thought status requires academics to make room in our curriculum for African epistemologies or indigenous knowledge systems thinking, where possible. Assessing the current literature led to the discovery of a three-fold success factor, the “anchor-approach-impact” (AAI) pattern, found to be a necessary pivot in existing decolonisation practices. A matter arising from this discovery is that AAI needed to be further explored with a view to developing a more generalised framework for taking decoloniality practice further.

In this paper, we propose F-D-E, a three-legged strategy that a Faculty/Discipline can use to get started. The first leg is called Foundation, the starter initiative to establish a shared understanding of what coloniality means to your discipline. Next is the Driver leg, which attempts to find an anchor in your discipline that may be used to bring decoloniality into the curricula. Enabler, being the third leg, is a collaborative, partnership-based but necessary effort to establish a shared community of practice for decolonisation. We conclude the paper with preliminary findings from an experimental attempt at making room for African Thought contributions from students; thereby exploring the AAI pattern in our advanced software techniques course.

Keywords: Decolonisation, African Thought, AAI pattern.

1 INTRODUCTION

Decolonising the curriculum means creating spaces and resources for a dialogue among all members of a university on how to imagine and envision all cultures and knowledge systems in the curriculum, with respect to what is being taught and how it frames the world [1]. Several South African Universities have made numerous attempts with little or no impact. First, there is the challenge of misunderstanding the concept. Decolonisation is not about dismissing or deleting knowledge systems created in the West; it is not to erode academic freedoms or tamper with disciplinary integrity; neither is it to antagonise existing instruments of Knowledge Production for classifying, organising, and representing knowledge that have proven to be effective. Finally, it is also not blind

indigenisation of the curriculum to correct past disadvantages and inequity. To deal with this misconception, thought leaders have endeavoured to spell out what decolonisation is.

Decolonisation is creating spaces for indigenous knowledge within existing and new curricula while allowing the consequences of this centralisation to be interrogated. By holding at its core, a set of questions that are persistently re-visited, discussed, and explored, Curriculum indigenisation challenges the notion of contingency learning, prerequisite knowledge, and linearity. Decolonisation provides the opportunity for students to take personal responsibility for their learning [2, 3, 4, 5]. It entails the genuine desire to make students re-think everything about 'knowledge acquisition' [6]. A move from Epistemic Violence to Epistemic Openness.[7]

Therefore, the goal of this paper is to broadly assess the existing community of decolonisation practices; during which we have discovered that the current state of decoloniality is pivoted on a three-fold pattern that we have chosen to call 'AAI'. The anchor-approach-impact (AAI) architecture, discovered from the literature [2, 3, 4, 5], is a necessary pivot or success factor in current decolonisation practices. A matter arising from this discovery is that the AAI pattern needed to be further explored with a view to developing a more generalised basis for future decoloniality practices.

We, therefore, argue that attempts made so far are yet to become a mainstream practice because future progress will require envisioning a foundational, driving, and enabling set of initiatives which further reinforces the AAI pattern. Foundation: make a discipline-driven effort to strategise around why decolonisation is important and establish a shared understanding of what coloniality means to your discipline (College, Faculty, or School). Driver: Find the anchor in your discipline that can be used to bring decoloniality into the curricula; for instance: chatroom, and design. Enabler: Innovation in collaboration and partnerships as a deliberate effort to establish a shared community of practice.

While other authors have in general presented what they have done in their specific cases, courses, or disciplines, the contributions of this paper are three-fold: (i) we reported as a discovery from existing works, an anchor-approach-impact (AAI) pattern, as the common denominator in some existing decolonisation endeavours; (ii) we then explored AAI further to propose how decolonising the curriculum could become a mainstream practice by envisioning a foundational, driving, and enabling set of initiatives, and (iii) finally we concluded with preliminary results from using the AAI pattern to decolonise the Advanced Programming Course at the University of Zululand (Unizulu).

The rest of the paper consists of the following elements. Section 2 presents the results of assessing four examples of decolonisation practices around the world which revealed the anchor-approach-impact (AAI) pattern. Section 3 discusses the proposed three-legged F-D-E strategy needed to make decolonisation a mainstream Teaching and Learning practice in an academic community. Section 4 shares the preliminary results from our Unizulu attempt at making room for decoloniality in an Advanced Software Techniques course. Finally, Section 5 concludes the paper and recommends how the Unizulu study could be extended in the future.

2 ASSESSING EXAMPLES FROM THE COMMUNITY OF DECOLONISATION PRACTICES

In this section, we present four examples from the existing community of Curriculum Decolonisation practices. Lessons learnt from these four examples have been generalised into the anchor-approach-impact pattern discovered to be a common denominator among the decolonisation examples studied.

Table 1: Evidence that Anchor-Approach-Impact Pattern works for Decolonisation.

Example Studies that used the AAI Pattern	Creating Spaces for alternative canons of knowledge which have been marginalised or dismissed: “The Anchor”	Finding Room and Mechanisms for Indigenisation: “The Approach”	Demonstrate a genuine desire to make students re-think “Knowledge Acquisition”. <i>What can they contribute while enriching their own learning experience?</i> “The Impact”
(A) Using Empathic Approaches in Engineering Capstone Design Projects to Decolonise [2]	Students enriched theories learnt in class with “perspectives of diverse stakeholders “. The process requires student teams to document how they achieved this.	“Translate perspective of diverse stakeholders into empathic skills”. Students are required to reflect on the new skills they have acquired as a team activity.	Student teams document and share how they acquired those empathic skills, reflecting on the role played by newly acquired skills in arriving at the final product design.
(B) Using the interface between ICTs and Indigenous pedagogy to decolonise [3]	The Interface of ICTs and Indigenous Pedagogy was tagged and named ISP (Indigenous Standpoint Pedagogy).	A chatroom was introduced to allow students an additional space for their reflections and learning over the semester.	Impact and efficacy of ICT use in assisting student reflection, provocation, and critique as part of student summative assessment or feedback.
(C) The design & implementation of a new course to decolonise and	The new course “facilitated a small group of indigenous and non-indigenous	A section of the course was designed to “answer the calls to action by the Truth and Reconciliation	Indigenous Knowledge which was shared by Knowledge Elders was discussed via Sharing Circles to

indigenise Engineering [4]	engineering students to think critically about making place and space for indigenous peoples and worldviews in Engineering”	Commission (TRC) of Canada to learn the truth about Canada as coloniser and use education as a tool for reconciliation”.	which they were invited. Students were expected to give feedback via critical reflections as a summative assessment.
(D) Indigenous Knowledges and Pedagogies in Science-based Learning [5]	The Anchor: "The challenge is to ensure strategies used will meaningfully support learning while reflecting local cultural traditions, languages, beliefs, and perspectives."	The Approach: “ Ininiwi-kisk-ntamowin , a model for science and math programming in indigenous settings, is applied to a culturally relevant environmental education program called Bridging the Gap (BTG).”	The impact: “Evaluating BTG within the context of the Ininiwi-kiskntamowin model generates an enlightening illustration of the nature of the model as a process of lifelong learning and suggests the need to consider alternative pedagogies and educational frameworks when developing and evaluating culturally relevant environmental education programs.”
(E) Towards Using African Thought to Decolonise the Curriculum	The Anchor: African Thought as a discussion mechanism for students to make their own contributions to how AI-enabled case studies could be adapted to address African challenges.	The Approach: A set of questions were framed for student groups to discuss. Each question delves into specific African thought in contrast to the theoretical principles on which the case studies were built.	The Impact: Using the D2R process of Figure 2, students are provided the opportunity to contribute responses to questions that can be used as feedback for the decolonisation activity.

The Anchor principle emanates from customising the ‘disciplinary anchor’ for your specific use. The next principle is to craft your indigenisation approach around the customised anchor to decolonise your course. Finally, engage the students, preferably in groups, such that they rethink how they learn via robust discussion as their own contribution and feedback to the decolonisation activity. In the four examples captured in

Table 1, we demonstrate how these three principles became the evidence of success factors on which the examples depend.

For each of the example studies, A to D in Table 1, the following observations confirm how our Anchor-Approach-Impact pattern was evidenced:

“Using Empathic Approaches in Engineering Capstone Design Projects to Decolonise” [2], is an example in the Engineering discipline; hence it was anchored in getting students to consider stakeholders' perspectives as an intentional activity during Capstone Design projects. Ordinarily, students think relying on classroom-espoused theories is enough to complete a project. However, the study shows that empathic skills were acquired by students who paid the necessary attention to stakeholder perspectives. This example demonstrates how to create space for empathic skills to be acquired in a particular way and for the students to discuss its role in or value added to their learning.

“Using the interface between ICTs and indigenous pedagogy to decolonise” [3], this example falls under indigenous education anchored in a special pedagogy called Indigenous Standpoint Pedagogy (ISP). Students meet in chatrooms to discuss various aspects of ISP. Their reflections were based on the impact and efficacy of ICT use on their learning process.

In “The Design & Implementation of a New Course to Decolonize and Indigenize Engineering” [4], another Engineering example is presented. Moreover, this is an optional course that any Engineering student can take. It is different from the earlier Capstone Design Example in that Knowledge Elders were identified and invited to participate as resource lecturers for the course. Small Groups of indigenous and non-indigenous students gave feedback on how their knowledge was enriched by learning the truth about Canada as a coloniser and the use of education as a tool for reconciliation.

“Indigenous Knowledges and Pedagogies in Science-based Learning” [5], this example is rooted in science-based environmental education. A model for science and math programming in indigenous settings was created as the anchor to be used and given an indigenous name (BTG). Students engaged in reflecting on local cultural traditions, languages, beliefs, and perspectives which were incorporated into the Bridging the Gap model. Students also explored the opportunity to discuss how much they think the model adequately supports the process of lifelong learning.

Moreover, all the cited studies, including this one, clearly demonstrate that decolonisation of the curriculum can be achieved through different approaches and more importantly through collaboration between academics and the students in that specific discipline. The next section builds on the AAI pattern to propose a practical strategy for bringing decolonisation into everyday practice.

3 THE THREE-LEGGED STRATEGY FOR DECOLONISING THE CURRICULUM

Arising from the foregoing is the need to find a practical origin for the Anchor principle in the AAI pattern. It is not desirable that interested academics or departments simply thumb-suck the Anchor principle, hence the need for a strategy that gives rise to a credible origin for the anchor. Therefore, we propose F-D-E, a three-legged strategy. The

first leg is called Foundation, being the root initiative to establish a shared understanding of what coloniality means to your discipline. Next is the Driver leg, which attempts to find the anchor in your discipline that can be used to bring decoloniality into the curricula. Enabler, being the third leg, is a collaborative, partnership-based innovation that is necessary to establish a shared community of practice for decolonisation, illustrated in Figure 1.

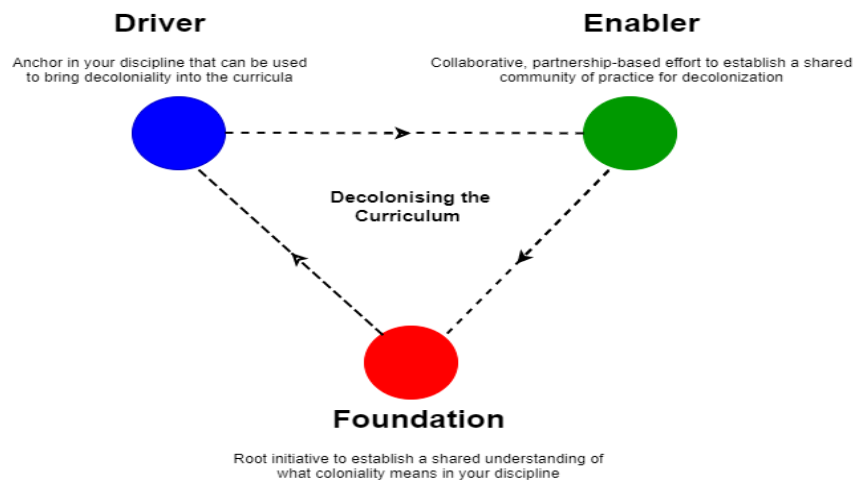


Fig 1. Illustration of F-D-E strategy

3.1 A Decolonisation plus Indigenisation Strategic Objective

The first leg of the initiative is developed as a crucial Foundation for the Decolonisation project. This should be done at a higher level than a department, usually at Faculty, College, or School, where a Teaching and Learning Goal has already been set as a strategic objective. The Foundation leg requires that you add the Decolonisation project as a new strategic objective to address the following issues:

- Start by discussing why Decolonisation is important to your College, Faculty, or Discipline.
- Then, establish a shared understanding of what coloniality means in your discipline. This step is complete only when the definition of terms has been agreed upon for this minimal list of terminologies: Indigenous Innovation, Coloniality, Africanisation, Afrocentric pedagogy, African epistemologies and transformation in your particular context, the value of empathy, epistemological openness, etc.

At a recent conference, the First Unizulu Conference on African Thought 2022, the resolution of the conference, in part, recommended that Decolonisation itself is not enough without transformation brought about by African-thought-based indigenisation. In other words, a valid indicator that a successful foundation has been established would

probably be that some transformation target or deliverable has been set. A typical target at Unizulu would be that a glossary of terms is developed for Decolonisation in the Faculty of Science, Agriculture, and Engineering for our academic colleagues to use.

3.2 The Disciplinary Anchor that can be used to bring decoloniality into the Curricula

The **Driver** leg is based on the assumption that the foundation is already laid by your Faculty, Discipline, or Institution. Several authors alluded to this step as exemplified in the following works:

- Rowena Arshad calls these: Getting Started ideas [8]
 - Consider the ‘diversity’ of your student groups: “to ensure learning content moves beyond Western to global frameworks”.
 - Find an anchor: “alternative canons of knowledge which have been marginalised or dismissed as a result of colonialism that should be included and discussed with students”.
- Engineering examples from Canada have successfully used Indigenous Knowledge discussed in chatrooms to influence how Engineering Design is conceptualised. [3, 9]

Borrowing an idea emanating from the World Economic Forum of 2022 [WEF, 2022], we suggest Creating Data Product research projects with a view to experimenting with AI-enabled Models of Curriculum Decolonisation extracted from Indigenous Knowledge Systems (IKSs). This will require that IKS databases become sources of datasets that can be used to furnish ICT students with alternative canons of knowledge.

3.3 Sustainability Derived from Community of Decolonisation Practices

We need strong enabling engagements which are sustainable before decolonisation can become mainstream practice. It is common knowledge that there are pockets of excellence around the world in this practice. Therefore, the Enabler leg of this initiative will require:

- Innovative curricula that ensure a range of voices and perspectives are represented.
- Partnerships that lead to re-conceptualising the curriculum to reflect wider global and historical perspectives.
- Community of emerging academics and experts with a common purpose of gathering African epistemologies and ways of transforming them into Indigenous Knowledge Systems (IKS).

Sustainability requires, we should innovatively create an African Thought environment for the science, technology, engineering, and mathematics (STEM) community where:

- New Afrocentric STEM pedagogies are developed.
- Capacity-building and reorientation of academics takes place.
- Indigenous Knowledge Systems (IKS) are explored to solve problems.

4 THE UNIZULU TRIAL: DECOLONISING AN ADVANCED SOFTWARE TECHNIQUE COURSE

This experiment on decoloniality explores the AAI principles discussed in Section 2, such that the Anchor is: African Thought (alternative perspective based on African traditions and cultural beliefs), being a discussion mechanism for students to make their own contributions to how AI-enabled Case studies could be adapted to address African challenges. The Approach is a set of questions framed for student groups to discuss. Each question delves into specific African thought in contrast to the theoretical principles on which the Case Studies were built. The Impact is the feedback received from analysing the outcome collected as data from the D2R process in Figure 2. The response activity in D2R produces survey data that are subsequently analysed thereby providing feedback on the decolonisation activity. But first, Section 4.1 provides some preliminary insight into what STEM decoloniality means, followed by an experimental data collection described in Section 4.2. The results from the student group discussion and individual survey data analysis are discussed in Section 4.3.

4.1 Role of African Epistemologies and Indigenous Knowledge Systems in STEM Decoloniality

The study assumes the STEM discipline knows very little or nothing about the role that African epistemologies and Indigenous Knowledge Systems, IKS should play in Decoloniality. If a discipline has already put in place how anchors will be selected for its decolonisation practices, then the next issue to be resolved is what diversity means to that discipline. In the case of Unizulu, every discipline is expected to adopt IKS-based African Thought as a theme that designated courses must subscribe to. Therefore, diversity (or student groupings) can be used to determine which aspect of IKS each student group might be interested in. What does this mean for an interested academic who is just getting started?

Students need to have a say in the choice of IKS that constitutes a relevant indigenous or alternative perspective in their discipline. The challenge of decoloniality is overcoming student resistance. Academics must be willing to let students say how they want to be assessed, especially if that helps to evaluate Knowing in contrast to Learning. To what extent will a discipline be willing to go, probably, as far as replacing Final Examination with Summative Assessment at least in designated decolonised courses?

These are the reality checks if a discipline is genuinely committed to finding out what students know not just what they have learnt.

We argue in this paper that modern African-Thought-driven STEM should be about letting African Culture and Indigenous Knowledge Systems influence Technological Sciences Pedagogies. By making room for global perspectives or otherwise neglected indigenous values in teaching content; we can ensure students are not left out of the decolonisation process. For instance, a credible mechanism that has been used successfully is to let students deliberate on how separating learning from knowing could be achieved. Learning has been oversimplified into Formative and Summative activities. Pedagogies cannot just be either Mediation or Facilitation, there should be a middle ground where African ways of knowing are given the prominence they deserve by letting knowledge elders bring African values and philosophical thinking to enrich University teaching and learning. How else will STEM make room for students to learn conceptual values such as Yoruba ‘Omoluwabi’ and South African ‘Ubuntu’ being African values that have made corruption a no-go area to African Leaders who imbibed these values from childhood? ‘Omoluwabi’ is about preserving the good name of your ‘family’. Ubuntu moves away from Descartes’ *cogito*, ‘I think therefore I am’ to ‘I am because we are’ [11].

The foregoing argument gave rise to an AAI-based experimental study, described in the next section, which collects data from our attempt to allow students to contribute to our decolonisation effort.

4.2 Student Contribution to Decoloniality – Data Collection and Analysis

In our endeavour to indigenise or bring decoloniality to bear on STEM subjects, we present this experiment within an Advanced Software Techniques course context; where group projects are assigned to teach students how to implement software products with real impact on a community of their choice. Leading questions are constructed to assist students to discuss how cultural beliefs, language communication constructs, and African communal practices can influence the choice of decoloniality perspectives that are explored to enrich a course like ours in Computer Science.

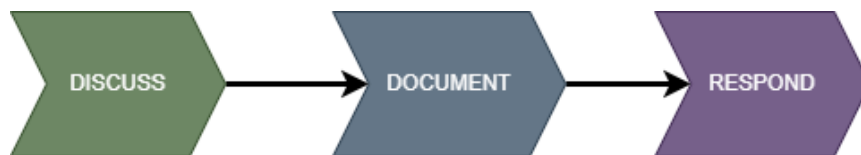


Fig 2. The D2R process for capturing student contributions to decoloniality.

A third-year course (Advanced Software Techniques) offered by the Department of Computer Sciences was used in this study. The class consisted of 71 students, divided into 10 groups. To gather data, we created a simple feedback activity (see Figure 2) called Discuss, Document, and Respond (D2R) integrated into our Learning Management System (LMS) that drives student group participation in the decolonisation

process. As a preliminary measure, a set of five questions were crafted as the basis of the discussion in our D2R process. These questions were first presented in a tutorial class setting with hints on how to reason with local cultural traditions, languages, beliefs, and perspectives as they affect a specific AI-enabled Case Study embedded in Group 1 Project. The students were given two weeks to reflect (or discuss) and summarise (document) their reflections in groups. Then, each student was asked to participate (respond) in a survey that captures their individual perception of the ideas emanating from their group discussion. The group feedback was subsequently visually (not statistically) correlated with their individual feedback responses, see Table 2.

Table 2: Student Perspective on Decoloniality Using D2R

Basis of the D2R Discuss and Summary of Documented Student Perspectives
1. What are the African Perspectives that can be incorporated into how animals are compared? • Animals in African cultures are not accorded human dignity. However, their value as important co-inhabitants in a wider ecosystem is recognised. Therefore, African communities keep animals for security, hunting, or livestock purposes. • Classification was based on animal behaviours or superstitious beliefs. For example, Baboons and Owls are considered evil while deadly animals like lions and tigers are regarded as dangerous. • African animal names reflect physical features (physical appearance, skin colour, spots, body structure, size, etc.). • Livestock animals are considered more important in African cultures and therefore a measure of family wealth.
2. Why is Western Culture most dominant in Computer Technology? • Early designers of computer technology were Westerners. Therefore, uptake was very slow outside of those societies. • Western countries are associated with a highly developed economy, extensive investment in computer technology research and development, and advanced manufacturing processes and skills. • In Western education, STEM subjects were esteemed above literary subjects, but still, some non-western countries were left behind. • Apple, Google, and Amazon emanated from a culture of entrepreneurship and risk-taking. • The English language dominated in technical documentation, programming languages, and software development tools. It is possible some other cultures lost their technical know-how due to little or no documentation. • Finally, intellectual property laws in computer technology are based on Western legal systems, which makes them more favourable to Western companies and individuals.
3. Why do students play down or fail to consider at all the role of their own culture in Software Solutions reasoning or thinking?

• The lack of knowledge on how culture influences the way we think and solve problems is a factor. • Students may surrender to pressure to adhere to prevailing cultural standards or expectations. • Restricted exposure or lack of diversity promotion: people not exposed to a variety of cultural viewpoints may not understand the influence that culture has on how they think and behave. • Students are only exposed to Western notions of software development and believe their cultural values and practices are inappropriate in this context. • Education system: not enough support or resources to explore their cultural perspectives in software development.
4. Can you reason out African Kingdom structures that can be used to illustrate Design Patterns such as Strategy, Observer, or any other pattern? • Polygamy marriage • Stokvel • Kingdom of Benin • Hierarchical Structures of Chieftaincy • Chain Tale • Ubuntu
5. What are the stumbling blocks in the way of using African Community Practices and Beliefs to teach technology in the classroom? • Technology in Western countries uses tried and tested concepts that are documented. These do not exist from African perspectives. • Misconceptions about African cultures, lead to stereotypes and prejudices, making it difficult to integrate African community practices and beliefs into technical education. • Language barriers: it is difficult to explain technological concepts in a way that all can understand. • Lack of teachers with the required expertise, such as knowledge to demonstrate the technology with African cultural examples.

4.3 Discussion of Results

In this subsection, we discuss the foundation laid for decolonisation via Group Project 1 in our Advanced Software Techniques course. Without going into details of the project, we set the aim of the experiment to appraise possible students' resistance to the idea of updating the Computer Science curriculum with African epistemologies or ways of knowing. This must be done most subtly, hence the crafting of questions that allows student groups to reflect on aspects such as Plausibility, Impediments, Western Dominance, African Perspectives, and Resistance, see Table 2.

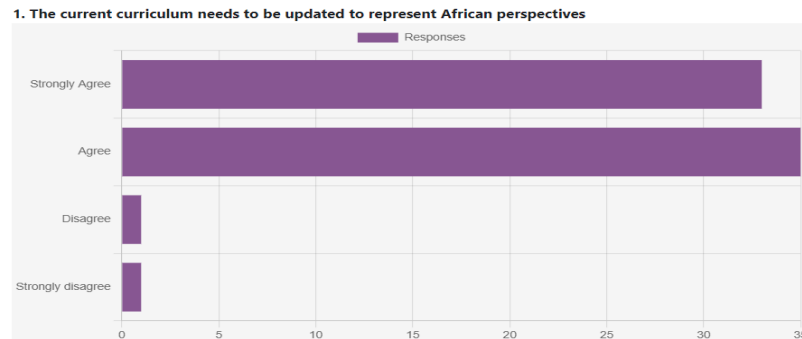


Fig 3. Survey item 1 responses.

The *plausibility* discussion helped to open the eyes of most students to the need to update the current curriculum with African perspectives. Approximately, 50% of the students agreed, while 47% strongly agreed and only 1% disagreed or strongly disagreed with survey item 1, as shown in Fig 3. Although it does not correlate directly with Question 1 it shows how cultural beliefs about pets (animals in general) assisted the students to make up their minds overwhelmingly in favour of making room for African perspectives in the curriculum.

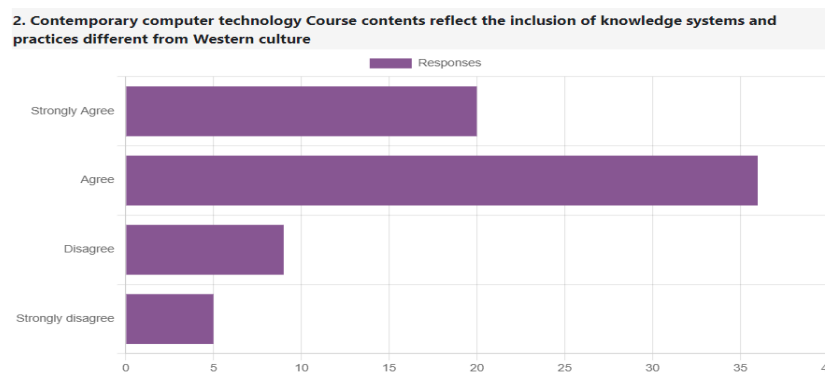


Fig 4. Survey item 2 responses.

The dominance of Western culture is what students were asked to comment about next. They cited entrepreneurship and risk-taking as the main factors for the dominance of Apple, Google, and Amazon Web Services. The English language's advantage over all others factors highly for Western dominance in Computer Technology. Survey item 2 explores whether computer technology courses' content reflects the inclusion of knowledge systems and practices different from Western culture. About, 52% agreed and 29% strongly agreed, while 13% disagreed and 7% strongly disagree, as presented in Fig 4. There is a strong correlation between the answer and the survey response in that some cultural values and knowledge systems can be similar, such as risk-taking,

while language barriers can be addressed with time. One gets a sense that it is better to start bringing African thought into Technological Sciences to preserve the values that African cultures have to offer.

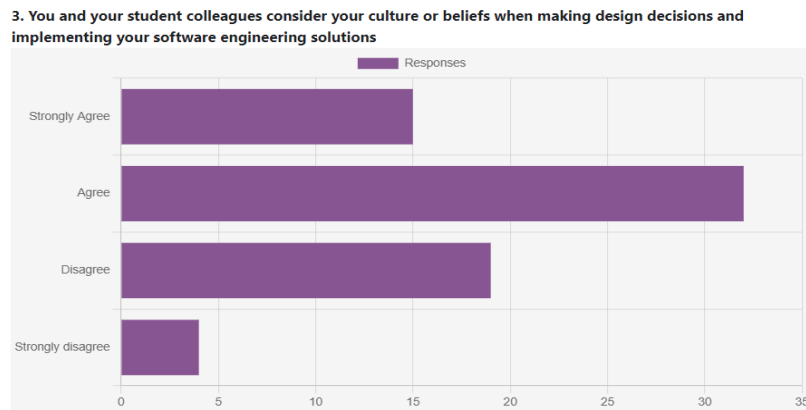


Fig 5. Survey item 3 responses.

Students attributed their lack of interest (or *resistance*) in considering what their culture can contribute to Software design to little or no African Language conceptualisation of Concepts like Object Orientation, Abstraction, and Architectural Design to name a few. However, their survey responses did not attest to this majority opinion which was quite impressively made during the discussion. For example, approximately 46% agreed and 21% strongly agreed, while about 27% and 6% disagreed and strongly disagreed that they consider culture or beliefs when making design decisions for software engineering solutions, Figure 5 illustrates the responses for item 3.

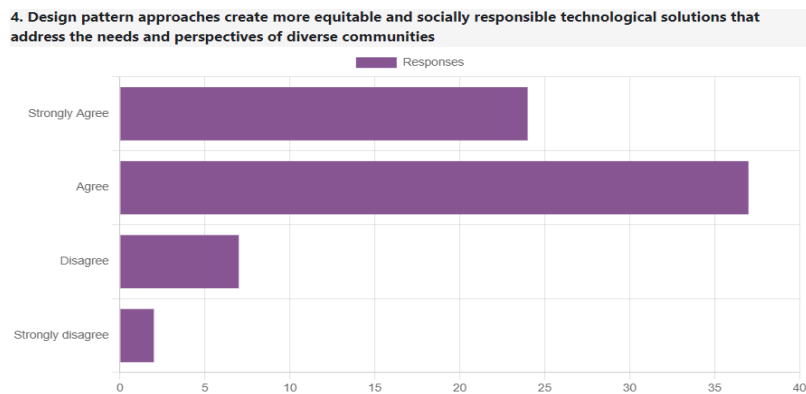


Fig 6. Survey item 4 responses.

Design patterns are ready-made solution concepts that may or may not have any bearing on African kingdom structures. Design patterns will probably be found in narratives on solutions and victory stories found in cultures and African traditional storytelling. This feedback shows that student discussions centred around famous African kingdoms. From the survey, as shown in Fig 6, approximately 53% agreed and 34% strongly agreed, while about 10% disagreed and 3% strongly disagreed. Therefore, this correlation between the group responses and the survey responses was evident as almost all respondents agreed that patterns can produce more equitable and socially responsible technological solutions. However, there are no research findings yet to support this response.

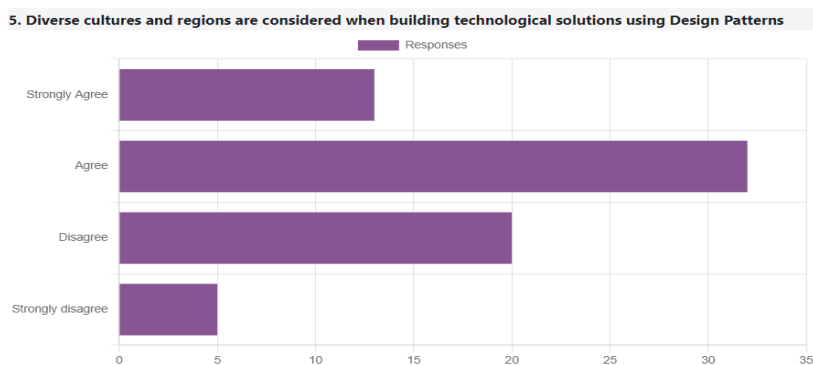


Fig 7. Survey item 5 responses.

Finally, the question of *impediments* in the way of decoloniality was addressed. Students identified reasonable impediments such as limited African perspectives in current technology design, misconceptions about African culture, language barriers, and a lack of trained teaching personnel. However, they think design patterns already capture the diverse perspectives that can result from diversity as nearly 46% and 19% agreed and strongly agreed and roughly 29% and 7% disagreed or strongly disagreed (see Fig 7) but there is no literature evidence that it is the case.

5 CONCLUSION AND FUTURE WORK

This study established the emergence of a dominant pattern of Anchor-Approach-Impact (AAI) in existing decolonisation attempts which can be practiced by interested academics. What was not clear is how you can arrive at the anchor such that there is uniformity in a particular discipline. Our Foundation-Driver-Enabler (FDE) strategy provides the solution to that challenge. Finally, we can conclude that current results show that Decoloniality is not unpopular, impediments to Decoloniality exists, the dominance of Western STEM education needs to be addressed, there is little or no African Perspective in our current curriculum, and student resistance to introducing new ideas can be overcome with the adequate scholarship of teaching and learning in STEM.

Scientific Teaching and Learning is not complete without the Scholarship of Teaching and Learning (SOTL). This is because research is needed to understand students' perspectives on what they are learning. Incorporating the inherent philosophies about being and living in harmony in your community is one way of enriching what students are learning. Interacting with communities in proximity or far away from your institution will result in identifying Knowledge Elders and the role they can play in decolonising the curriculum. Don't decolonise a curriculum for its sake, rather do it to influence or transform your community. We need a multidisciplinary approach: start from your discipline and move systematically to epistemic openness that is informed by evidence from research.

The limitation of this study is demographics are not well represented as our class only consisted of native African students. In the future, and arising from the foregoing, the second half of this study will investigate how students can become empathic designers who use what they are going to study in Group Project 2 to solve community challenges. Additionally, other stakeholders, such as Lecturers and University management personnel will be given the same exercise to get broader views and perspectives as this will also address the demographics limitations.

References

1. Abdi, A.A. Decolonizing Philosophies of Education. In: Abdi, A.A. (eds) Decolonizing Philosophies of Education. Sense Publishers (2012)
2. Guanés, G., Wang, L., Delaine, D., Dringenberg, E.: Empathic approaches in engineering capstone design projects: student beliefs and reported behaviour, *European Journal of Engineering Education*, 47:3, pp 429-445 (2022)
3. Winslett, G., Philips, J.: ICTs and Indigenous pedagogy: Techniques of resistance in chat rooms. In: *ascilite*: pp 729-734 (2005)
4. Cicek, J., Herrmann, R., Forrest, R., Monkman, K.: Decolonizing and Indigenizing Engineering: The Design & Implementation of a New Course. In: 2022 Canadian Engineering Education Association (CEEAA22) Conf., York University; June 19 – 22 (2022).
5. Sutherland, D., Swayze, N.: Including Indigenous Knowledges and Pedagogies in Science-Based Environmental Education Programs. In: *Canadian Journal of Environmental Education*, 17, pp 80-96 (2012)
6. Phillips, J., Whatman, S., Hart, V., Winslett, V.: Decolonising university curricula. reforming the colonised spaces within which we operate. In: *Indigenous knowledges conference, Reconciling academic priorities with Indigenous realities.*, June, University of Wellington, (2005)
7. Ammon, L.: Decolonising the University Curriculum in South Africa: A Case Study of the University of the Free State, A Masters Thesis submitted to DEPARTMENT FOR SOCIAL STUDIES Linnaeus University, Sweden, (2019)
8. Decolonising the curriculum - how do I get started?, *Times Higher education, Campus Newsletter*, <https://www.timeshighereducation.com/campus/decolonising-curriculum-how-do-i-get-started>, last accessed on 2022/11/22
9. Pete, S., Schneider, B., O'Reilly, K.: Decolonizing Our Practice - Indigenizing Our Teaching. In: *First Nations Perspectives* 5, 1: pp 99-115 (2013)
10. Mignolo, W., *Coloniality Is Far from Over, and So Must Be Decoloniality*, In: *A Journal of Arts, Context and enquiry*, Vol 43: pp 38-45 (2017)

11. Said, E.: Orientalism, 1st edition, Vintage Books, New York (1979)
12. Jansen, J.: Introduction – Part II. Decolonising the curriculum given a dysfunctional school system? *Journal of Education* (University of KwaZulu-Natal). 68, pp 3–14 (2017)
13. Savo, H.: Decolonizing Knowledge in South Africa: Dismantling the ‘pedagogy of big lies’
In: *Ufahamu, A Journal of African Studies*, 40(2), pp 47–65 (2018)

The Impact of Social Media Use on Digital Well-being of University Students

Enwill Isaks¹, Zane Davids^{1,2} [0009-0008-2760-5567], Lisa Seymour^{1,2} [0000-0001-6704-0021] and Sharon Geeling^{1,2} [0000-0003-0171-9634]

¹ Department of Information Systems, University of Cape Town, Cape Town, South Africa

ISKENW001@myuct.ac.za, mo.davids@uct.ac.za,
lisa.seymour@uct.ac.za, sharon.geeling@uct.ac.za

² CITANDA, University of Cape Town, Cape Town, South Africa

Abstract. As within our everyday lives, social media use and online interactions continue to increase their significance within our schools and universities. Educators are increasingly using social media platforms to teach their classes, and students use these platforms to facilitate their learning. Our increased dependency has raised real concerns about the many hours spent by students using these digital technologies and what the possible impacts could be. Research focused on understanding the psychological consequences stemming from the daily use of social media has only emerged recently and several critical aspects of social media use amongst students are still unanswered. The purpose of this research is thus to understand the influence of social media use on the digital well-being of university students. The digital well-being theoretical model was used as a theoretical lens for this research study. Students from a leading South African university were interviewed. The results show that social media use can harm and benefit students' digital well-being. The use benefits students by connecting and building supportive relationships amongst classmates, facilitating the sharing of learning resources which supports collaborative learning resulting in a better understanding of coursework. Harms associated with social media use include being distracted, long periods of procrastination, negatively comparing oneself with your peers, and the fear of missing out on activities your peers are involved in without you. Universities providing supportive student online strategies and students having a healthy relationship with social media also aide student digital well-being. These findings extend Büchi's digital well-being theoretical model.

Keywords: Social Media, Digital Well-being, University Students.

1 Introduction

Social media use and online interactions play a significant role in our everyday lives. These same social media platforms have also found a prominent place within our schools and universities, in how educators teach their classes, and in how students learn [48]. University students use social media to improve their learning outputs

through collaborative activities in virtual learning environments [16]. The foundation that supports these learning outputs is the social media networks students belong to. These online networks have become the primary means students use for online interaction and communication [36]. Our increased dependency on social media use has raised several questions about the many hours' students spend using digital technology and what the possible impacts could be [1]. Consequently, there has been a recent increase in research focusing on the psychological consequences stemming from the daily use of social media [5, 50, 56].

Our study connects with contemporary investigations explaining psychological consequences stemming from the daily use of social media. We took an African perspective and focused on students attending a South African university and how social media use impacts their digital well-being. In this context, digital well-being is defined as the self-evaluation of one's degree of happiness in relation to the effects resulting from one's use of social media [8-9, 59].

Our central research question is "How does social media use impact the digital well-being of university students?" The objectives of this research study are thus to explain: (1) how university students use social media, and (2) how these online interactions with their peers and the higher education institution influences their digital well-being. Section 2 presents literature related to the research problem. Section 3 discusses the methodological approach used in the study, while Section 4 discusses the harms and benefits students experience when using social media and their subsequent impact on student digital well-being. Section 5 concludes by discussing the contributions made by the study, the limitations and future research opportunities.

2 Literature Review

The emergency lockdowns brought on by the COVID-19 pandemic highlighted the important role social media can play in online teaching [41, 53]. Subsequently, the use of social media by higher educational institutions during the COVID-19 pandemic positively impacted the educational process by motivating students to become active participants in their own learning [43]. Using social media allows students to share information like videos, text, images, documents, links, discussion boards, and social bookmarking. Students are also able to complete online presentations, assignments, and can join online forums, blogs, and wikis [20, 42, 46, 57, 65]. Social media has benefits yet it also presents challenges.

2.1 Benefits Associated with Social Media Use

One of the main benefits of social media use is allowing people to stay connected. Far distances and geographical barriers between individuals are not a problem and thus we are able to connect with each other on a regular basis [12, 56]. The act of communication plays a central role in the successes we've achieved as a human civilisation. Likes, retweets, and online comments have become an integral part of the tapestry of our lives. Online users now use social media to share their varied perspectives on

topics of interest. Respecting these shared sentiments leads to an increase in robust online engagement that is enriching to all involved [28].

Social anxiety suffers fear social interactions or being judged by others. These individuals find face to face engagements with other individuals significantly challenging. The benefit social media provides to social anxiety sufferers is the ability to have online conversations with others. Having these conversations through a computer / mobile screen alleviates the anxiety and stress associated with partaking in social interactions. Social media support these individuals in making social connections, building friendships, improving their self-esteem and overall well-being [21-22, 44].

In education, social media use offers increased engagement opportunities that strengthens the bond between educators, their students, and the course material [2, 17, 37, 62]. Another benefit of social media use is collaborative learning where students work together to solve a problem and complete their tasks [2, 25-26]. By extension, Chaijum [11] states the combining of social media technology with group brainstorming exercises enhances the teamwork, analytical, and problem-solving skills of students.

2.2 Negative Impacts of Social Media Use

Social media has a dark side and in the wrong hands can be used in malicious ways that affects unsuspecting users across all age groups including students. Social bullying is a significant problem amongst students. Whether these attacks are made directly or indirectly, the victim tends to experience increased levels of anxiety, depression, and loneliness which is harmful to their overall health and state of well-being [64].

The pervasive use of social media has also raised concerns about the marked increase in two psychological health concerns, namely, the Fear of Missing Out (FOMO) and telepressure which have been noted amongst university students who are regular social media users [47]. FOMO refers to an alarming feeling of missing out on rewarding experiences [45]. Telepressure describes the need to respond quickly to text or email messages received. An individual would find it difficult to concentrate of work unless he / she has responded to the message [6]. These negative experiences of social media use are harmful to their overall health and state of well-being [47].

Certain social media users tie their self-image to their online activities. Engaging in negative social comparisons and not receiving enough **likes** on your online posts leads to a reduction in one's self-esteem [10]. This worrying behaviour further reinforces negative self-evaluations which harms their overall health and state of well-being [29, 56].

A further downside of continuous online engagement with online media is students being distracted from their academic work [60, 66]. Many students struggle with self-regulation of their media use when confronted by these distractions [30]. The result is the procrastination in their academic work [51]. Le Roux et al., [31] studied university students from South Africa, Botswana, and Namibia. The study interestingly shows that the extended use of online media does not appear to have a harmful impact on the academic performance of university students. Hence more research is needed to better understand the impacts of social media use on academic performance.

2.3 Social Media Use Amongst University Students in Southern Africa

Le Roux and Parry [32] in their study highlight that students at a South African university tend to participate in several academic instant messaging groups (AIMGs) at the same time. A distinction is made between formal AIMGs created and administered by the teaching staff and organic (class and study) AIMGs created and administered by the students [40, 54, 66]. Students seem to prefer engaging more on the study AIMGs which is made up of a smaller group of students who personally know each other. There is a level of trust amongst these students who tend to collaborate on group assignments and share solutions. While acknowledging the limitations associated with self-reported data about media use behaviour [18], the le Roux and Parry [32] survey interestingly finds that neither formal nor organic AIMGs seem to be academically important to the performance, stress, or course perception amongst university students. Additionally, they appear to be neither useful nor harmful within higher education. Their study does suggest the need for more qualitative studies to further unpack the role of AIMGs in students' well-being and academic development.

Smartphone companies are finding ways to help their customers effectively manage their smartphone use. The result is the creation of digital well-being applications. The purpose of these applications is to help smartphone users become more mindful of how they use digital media [39]. Le Roux et al., [33] conducted a study focused on university students' perceptions, adoption, and use of digital well-being applications. Their study highlights the importance of understanding individual's motivations when adopting and using digital well-being applications. Adding that digital well-being is much more than simply using an application to manage smartphone use and needs to be a conscious reflection on digital media and its place within one's life.

2.4 The Digital Well-being Model

This study uses the digital well-being theoretical model from Büchi [9] to investigate the influence of social media use on university students. The strength of this theoretical model is that it offers a view on how the harms and benefits associated with digital practice impact digital well-being [9]. The original theory consists of both the macro (the society) and micro (the individual) levels of social media use linked to digital well-being with relationships connecting the two levels. Since the focus of our study was at the micro level, we've adapted the theory by removing the macro level and all the associated relationships. This left us with the micro level that presents the social media use of an individual linked to digital well-being.

Subjective well-being also referred to as **personal well-being** is a self-evaluation of one's quality of life. The measurement is the degree of one's happiness which is influenced by the levels in pleasure and satisfaction one experiences [15, 24, 27, 38]. By extension **digital well-being** is how the use of digital media influences the subjective / personal well-being of the online users. All behaviours related to digital media is referred to as **digital practice**. Individuals can either experience **beneficial** or **harmful** consequences resulting from their digital practice. Harms tend to decrease digital well-being whereas benefits increase digital well-being [8-9, 59].

3 Research Method

The research philosophy we chose was interpretive and qualitative in nature since our study attempts to develop a deeper understanding of digital practice amongst university students and its influence on their digital well-being. The higher education institution which the students attend was selected as the base for our qualitative study. The appropriateness of using a qualitative study method is based on it allowing us to carry out an in-depth exploration of the research phenomenon with the purpose of better understanding digital well-being amongst university students [13-14]. Our approach to theory was deductive in nature since we've applied predetermined codes derived from the Büchi [9] model to our findings. In this way we were also able to test the model.

Our data collection sampling approach was purposeful in nature [13-14]. We targeted university students aged between 18 and 25 years that are social media users. The student participants commonly identified with using WhatsApp as their preferred social media application. This platform is also mainly used by the university as the chosen teaching and learning social media support tool. The use of WhatsApp as a teaching and learning support tool stands out as a simple solution that both students and lecturers find easy and convenient [52]. Students and lecturers are already familiar with using WhatsApp which increases acceptance and integration of the social media platform into teaching and learning [55]. WhatsApp also enables inquiry learning that supports student creativity, peer-to-peer collaboration, and critical thinking [63]. However, it can also lead to student distraction and procrastination [30, 51, 58].

Students mentioned Facebook, Twitter, Instagram, and Snapchat but they played a much lesser role in the daily social media activities amongst students and their engagement with their university. Additionally, we interviewed university staff who are the official gatekeepers on the university's social media accounts. Their perceptions as the moderators on the university's social media accounts provide valuable insights into the digital practice of students and its influence of their digital well-being. A total of four university students and two university staff members were interviewed. Tables 1 and 2 represent the research participants with their unique ids.

Table 1. University Student Participants

Participant ID	Faculty / Dept	Course of Study	Year of Study	Age	Gender
I1	Biological Sciences	Bachelor of Science	2 nd Year	23	Female
I2	Humanities	Bachelor of Social Science	2 nd Year	19	Male
I4	Engineering & the Built Environment	Bachelor of Engineering	3 rd Year	23	Male
I6	Commerce & Information Technology	Bachelor of Commerce	2 nd Year	25	Male

Table 2. University Staff Participants

Staff Members		
Participant ID	Faculty / Dept	Job Title
I3SM	Law Faculty	Student / Course Administrator
I5SM	Marketing and Communications	Senior Manager (Social Media)

Our interview schedule was derived from the theoretical model and used to collect research data through semi-structured interviews. The interviews took between 45 minutes to an hour and the qualitative data analysis software programme Nvivo was used to store and analyse the data. We used thematic analysis to code and categorise the data into themes allowing us to gain a deep understanding of the phenomenon being studied [13-14, 49]. Figure 1 is a summary of the steps we've followed during our coding process.

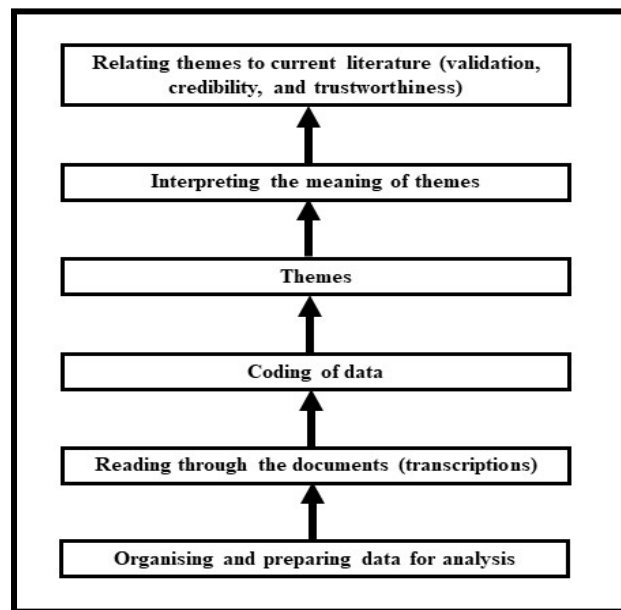


Fig. 1. The steps followed during thematic analysis (Adapted from Creswell [13-14]).

4 Research Findings and Discussion

This section elucidates the sentiments shared by students and staff on digital practice, social media harms, and social media benefits. Furthermore, this section focuses on how the benefits and harms associated with digital practice influences the digital well-

being of university students. Finally, we present our extension to the Büchi [9] model with the new categories and themes that emerged from this study.

4.1 Digital Practice

Social media is becoming the preferred mode of communication amongst students and between students and their faculty. Staff member I5SM states that *“social media has become an immediate communication platform for students. Anything and everything that is of concern or interest to students, we tend to see that increasingly as students go more and more to social media than any other platform. Students run to social media first (to engage amongst themselves and with faculty).”* Staff member I3SM noted that social media has become an important communication platform and adds that in the law faculty *“students will go to our social media posts that we put out and see what law courses we offer, and then they will either sign up or they will ask us to join our mailing list”*.

Student I6 further explains that student communication is facilitated by the creation of online student groups which forms the basis for their engagement. These online groups help *“students connect with students on social media. We used to be in a group during our first year at university. I used the online groups to connect with other students”* The findings confirm the current academic literature that online networks like social media platforms have become the primary channel used by students for online communication and interaction [36]. Additionally, South African university students readily participate in several online academic instant messaging groups [32]. Comparing our findings with the Büchi [9] model we noted the following new concepts the *preferred student-to-student communication platform* and *preferred student-to-faculty communication platform* that extends the model and helps better explain student digital well-being.

4.2 Social Media Harms Reducing Digital Well-being

The participants express social media harms as being FOMO, negative comparisons, distraction, procrastination, and increased anxiety and depression.

Procrastination and distraction are mentioned as harmful unintended consequences of social media use. Student I4 says that social media use *“aids our procrastination and distracts us often. I will admit that social media takes up many hours of my day. It strays me away from my coursework which adds to the list of things I need to worry about.”* Student I1 further explains that the ease of access to social media further compounds the negative impacts of procrastination and distraction on student life. It is *“so easy to go onto social media and spend a long time on it, which can hurt our performance in our courses. I probably spend a total of three hours a day on it, at times it's worse.”* The findings confirm the current academic literature where students being distracted from their academic work is another downside highlighted when engaging on online media [60, 66]. Struggling with self-regulation of their online media use leads to procrastination in their academic work [30, 51]. Comparing our findings with

the Büchi [9] model we noted *student procrastination and student distraction* as new concepts that extends the model and helps better explain student digital well-being.

Negatively comparing oneself with one's peers is concerning. This harm is highlighted by Student I1 as *"another downside of social media use. So, if someone already has anxiety, they could get worse. There are beauty standards and especially for a lot of women, you compare yourself to them and that can hurt. That can worsen one's mental health."* Student I6 adds that social media *"gives up this image that life's perfect. And so that would cause depression and anxiety when using social media to compare yourself with others."* The findings confirm the current academic literature that self-evaluations through social media comparisons negatively impacts one's state of mind [10, 29, 56]. Comparing our findings with the Büchi [9] model we noted that the model speaks about social comparison which aligns with our research findings.

Student I1 asserts that FOMO *"stands for fear of missing out. If you hear about your friends doing something interesting on social media and if you are not able to join them, you'll be sad. Worst still if you weren't invited."* Student I2 adds that having peers' comment on your social media posts is important, *"I would expect at least some peers to interact with the social media post in some way, either commenting or even liking it, that's what I'd expect."* The feelings of being ignored by your peers increases if these expected interactions do not happen. FOMO then has the tendency to increase anxiety and depression levels. Online activity and its impact on one's anxiety is further emphasised by student I4 who says *"I think it would hurt anxious people. The procrastination that is popular when using social media may cause more stress or anxiety in an anxious person's life."* The findings confirm the current academic literature in that students' anxiety and depression levels tend to increase when social media is used in a negative way. University students use social media to make online connections and share rewarding experiences amongst themselves. Where these online connections do not satisfy this need FOMO sets in [45, 47]. Comparing our findings with the Büchi [9] model we noted that the model speaks about FOMO, anxiety, and depression which aligns with our research findings.

4.3 Students Dealing with the Harms Associated with Social Media Use

These identified harms have a negative impact by decreasing the digital well-being of university students. It is important for students to be aware of the harms social media use can cause to their well-being and find ways to protect themselves from these harms. Student I4 provides insights into social media use and says *"well-being to me means being happy and finding a good balance in life. Balance of work, social media, socializing, and exercise. Well-being in terms of social media to me means not taking it too seriously and not letting it affect your mental health and opinions of yourself. But overall, my social media makes me happy. I've found a healthy relationship with social media. I don't let jealousy of other people's lives I see on social media affect me and I use it effectively to keep me entertained and in touch with my friends."* The student's mindset is thus important when using social media and not letting the harms associated with social media use affect one's digital well-being. This way of thinking

is captured by student I1 who says *“it's about being content with all the activities you do and the people you surround yourself with. Feeling secure in your relationships and the goals you have and what you're like.”*

The findings confirm the current academic literature where individuals who perceive themselves as being vulnerable with little control against the impacts of social media use experiences lower levels of well-being [4, 23, 34]. However, those individuals who take active control of their social media use tend to not show the same vulnerability and experiences higher levels of well-being [19, 35, 61]. One's perception and positively managing one's relationship with social media is thus important. Comparing our findings with the Büchi [9] model we noted the following new category *student's mindset* and associated new theme *healthy relationship with social media* that extends the model and helps better explain student digital well-being.

4.4 Social Media Benefits Improving Digital Well-being

The participants explained social media benefits as information (learning resources) sharing, collaborative learning, staying connected, and getting online support from peers.

The sharing of information (learning resources) through social media is highlighted as an important benefit. Staff I3SM affirms that *“social media is a very important tool to relay information from faculty to our students. If we didn't have the social media tool, students and faculty would use email to share information and it is not an efficient tool to distribute large amounts of information regularly. So, social media is, a very, very important tool to everybody.”* This sharing of learning resources also forms the basis of collaborative learning. Student I6 shares that the online study groups would *“share learning resources we've found to help better our understanding of our coursework.”* The findings confirm the current academic literature that collaborative learning where students share learning resources and work together to solve a problem and complete their tasks as important benefits of social media use [2-3, 11, 25-26, 32]. Comparing our findings with the Büchi [9] model we noted the following new concepts *sharing of learning resource* and *collaborative student learning* that extends the model and helps better explain student digital well-being.

Additionally, convenience and being connected are also mentioned as benefits associated with social media use. Student I4 feels that *“social media just makes things easier in the sense that you can efficiently communicate with peers and network(connect) in general with people around the world.”*

Student I1 adds that social media use is *“generally a good thing that allows you to connect with other people thus staying involved in what's going on and learning about other people as well.”* The findings confirm the current academic literature in that social media use offers opportunities for better engagement and connection between students and their educators [17, 37, 62]. Comparing our findings with the Büchi [9] model we noted that the model speaks about convenience and connectedness which aligns with our research findings.

4.5 Supportive Strategies to Facilitate Improved Student Digital Well-being

Social media has also become an effective tool when providing support to students who are going through challenges related to their academic life which then affects their well-being. Students who are struggling academically or financially are using social media to get assistance through online support campaigns. Staff I5SM explains that he sees *“many examples of students who use social media to ask for help. They can't afford to pay their fees and people (online community) will rally behind and come on board, and they will support the students' financially.”* Supportive strategies also include empowerment and mitigation steps taken by the university to facilitate improved student digital well-being. Educating students on how to make better use of social media platforms is an important university intervention and empowers students to better safeguard themselves against adverse online experiences. Staff I5SM outlines that the university *“does have a variety of resources as well as measures. The first instance is our Student Wellness services, for anyone who might have had a negative social media experience. So, through our Student Wellness services, we have psychological or psychosocial support offerings.”* Good online behaviour is essential, and the university provide students with practical resources like *“in 2021 we hosted two online Emma seminars with one of the top social media practitioners in the country, Emma Sadleir she's one of the top social media practitioners in the country. She shares the legal part of social media and social implications, everything around social media use, we hosted two of those seminars last year, open to students, to empower students and make them aware.”*

When students use social media in an incorrect manner it can lead to both the students and the university suffering from reputational damage. Staff I5SM says that *“students who use social media in negative ways will have an impact across the institution. It goes back to the responsible versus irresponsible usage of the platforms. Reckless social media use and posting of false accusations are insensitive to those targeted resulting in a negative impact on those involved.”* Staff member I3SM provides an actual situation in their dept where students used social media in an incorrect manner to target students and faculty. I3SM used the following approach to mitigate the negative impacts of this unfortunate social media encounter. *“We had our social media platforms closed for a while ago. It was on Facebook where they (students) were speaking badly of our unit (students, management, and academics), but it wasn't true. They(students) were making up stories. I had to quickly go in and try and shut them down. So, we blocked the students, and we deleted all the posts that wasn't true (insensitive and false accusations).”*

The benefit resulting from university staff proactively using the university's social media accounts to help empower students to successfully navigate their academic journey and mitigate reckless / inappropriate use on these online accounts would be a reduction in student anxiety and depression levels. This benefit would have an overall positive moderating effect and leads to an increase in student digital well-being.

The findings confirm the current academic literature. Screenshots is an educational initiative aimed at developing positive digital social skills within students. The goal is to improve the digital well-being of students through foster respectful online interac-

tions amongst peers, promote the safe use of communication technologies, and prosocial conflict resolution. The skills students acquire will help them use social media in a positive way which will also addresses potential harms [7]. Comparing our findings with the Büchi [9] model we noted the following new category *student online supportive strategies* and the associated themes as *student empowerment* and *risk mitigation* that extends the model and helps better explain student digital well-being.

4.6 The Resultant Model with New Categories and Themes

Figure 2 presents the Büchi [9] model extended to include the new categories and themes that emerged during our study. The original model is shown in black with the new categories shown in green and new themes shown in blue.

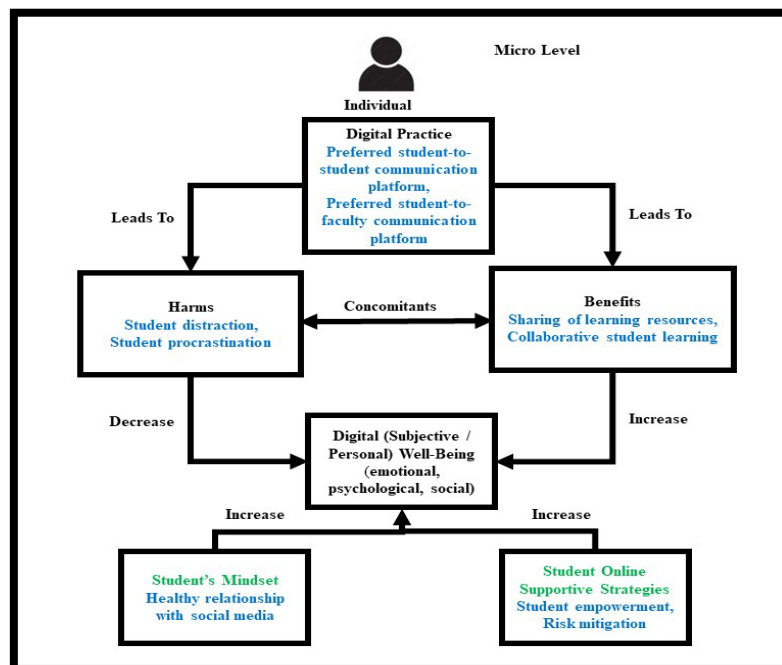


Fig. 2. The Büchi [9] model extended with the New Categories and Themes that emerged during this study.

5 Conclusion

Our core research question is “How does social media use impact the digital well-being of university students?” The objectives are thus to: (1) explain how university students use social media, and (2) understand how these online interactions with their peers and their higher education institution influence their digital well-being.

The research results support our decision to use the Digital well-being model [9] as a theoretical lens for our research study. The study shows that students tend to use social media platforms that are popular amongst their peers and family. They use social media for communication, entertainment, to unwind from their studies, and as a distraction when bored. Students also tend to spend several hours on social media.

Student digital well-being is affected by both the benefits and harms associated with social media use. The research highlights connecting and building supportive relationships amongst classmates, students sharing learning resources within their study groups which supports collaborative learning resulting in a better understanding of coursework, and the ease of using social media as the preferred mode of communication as social media use benefits. Additionally, knowing that important communication from their departments will be posted on the university's social media accounts, and getting the support from their peers and university staff when they post their concerns and requests for help are also benefits associated with social media use. Collectively, these benefits would bring about an increase in student digital well-being.

Alternatively, being distracted, long periods of procrastination, negatively comparing oneself with your peers, FOMO, and not getting positive validation when posting images and statuses are harms associated with social media use which in turn bring about a decrease in student digital well-being.

In summary, it is important that university students are aware of both the benefits and harms associated with social media use and the resulting impact on their well-being. Students are encouraged to ask for help when their digital practice negatively affects their digital well-being. Similarly, university institutions can monitor online student engagement for possible signs of students struggling with the negative impacts associated with social media use. Offering Student Wellness Services that provide psychological or psychosocial support to students who've had a negative social media experience are proactive steps that universities can take to address the risks associated with social media use. Additionally, hosting online seminars to help share the legal aspects of social media use and its social implications and identifying good and bad social media use. This will empower students and make them more aware and able to mitigate the risks. Having a positive mindset that aides a healthy relationship with social media also improves the digital well-being of students. These interventions and personal behaviour can help students successfully navigate their academic life and hence increase their digital well-being.

A limitation of this research study is its focus on university students aged between 18 and 25. Future research should include different groups of students like those in primary and high school. In this way, a more holistic view on the influence of social media use on the digital well-being of students will be obtained thus extending our current findings. Additionally, future research ought to investigate which social media platforms are more harmful to the digital well-being of students.

Acknowledgements. Funding support to write this study is acknowledged from the National Institute for the Humanities and Social Sciences. This research was conducted as part of a BCom Honours in Information Systems empirical research.

References

1. Akram, W., Kumar, R.: A Study on Positive and Negative Effects of Social Media on Society. *International Journal of Computer Sciences and Engineering*. 5(10), 351-354 (2017). <https://doi.org/10.26438/ijcse/v5i10.351354>
2. Alenezi, W., Brinthaup, T. M.: The Use of Social Media as a Tool for Learning: Perspectives of Students in the Faculty of Education at Kuwait University. *Contemporary Educational Technology*. 14(1), (2022). <https://doi.org/10.30935/cedtech/11476>
3. Ansari, J. A., Khan, N. A.: Exploring the role of social media in collaborative learning the new domain of learning. *Smart Learning Environments*. 7(1), 1–16 (2020). <https://doi.org/10.1186/s40561-020-00118-7>
4. Bánya, F., Zsila, Á., Király, O., Maraz, A., Elekes, Z., Griffiths, M. D., ..., Demetrovics, Z.: Problematic social media use: Results from a large-scale nationally representative adolescent sample. *PloS one*. 12(1), (2017). <https://doi.org/10.1371/journal.pone.0169839>
5. Bayer, J. B., Trieu, P., Ellison, N. B.: Social Media Elements, Ecologies, and Effects. *Annu Rev Psychol*. 71, 471-497 (2020). <https://doi.org/10.1146/annurev-psych-010419-050944>
6. Barber, L. K., Santuzzi, A. M.: Please respond ASAP: workplace telepressure and employee recovery. *Journal of Occupational Health Psychology*. 20(2), 172-189 (2015). <https://doi.org/10.1037/a0038278>
7. Bickham, D. S., Moukalled, S., Inyart, H. K., Zlokower, R.: Evaluating a middle-school digital citizenship curriculum (Screenshots): Quasi-Experimental study. *JMIR Mental Health*. 8(9), (2021). <https://doi.org/10.2196/26197>
8. Blank, G., Lutz, C.: Benefits and harms from internet use: a differentiated analysis of Great Britain. *New Media & Society*. 20(2), 618–640 (2018). <https://doi.org/10.1177/1461444816667135>
9. Büchi, M.: Digital well-being theory and research. *New Media & Society*. (2021). <https://doi.org/10.1177/14614448211056851>
10. Burrow, A. L., Rainone, N.: How many likes did I get? Purpose moderates links between positive social media feedback and self-esteem. *Journal of Experimental Social Psychology*. 69, 232-236 (2017). <https://doi.org/10.1016/j.jesp.2016.09.005>
11. Chaijum, N.: Using brainstorming through social media to promote engineering students' teamwork skills. *European Journal of Science and Mathematics Education*. 8(4), 170-176 (2020). <https://doi.org/10.30935/scimath/9555>
12. Chen, H. T., Li, X.: The contribution of mobile social media to social capital and psychological well-being: Examining the role of communicative use, friending and self-disclosure. *Computers in Human Behavior*. 75, 958-965 (2017). <https://doi.org/10.1016/j.chb.2017.06.011>
13. Creswell, J. W.: *Research design: Qualitative, quantitative, and mixed methods approaches*. Sage Publications, Los Angeles, CA (2009)
14. Creswell, J. W.: *Educational research Planning, conducting, and evaluating quantitative and qualitative research* (4th ed.). Boston, MA Pearson (2012).
15. Diener, E., Oishi, S., Tay, L.: Advances in subjective well-being research. *Nature Human Behaviour*. 2(4), 253–260 (2018). <https://doi.org/10.1038/s41562-018-0307-6>
16. Dongke, P., Sannusi, S. N.: Social Media Use among University Students in Malaysia during the COVID-19 Pandemic. *Higher Education and Oriental Studies*. 1, 17-25 (2021). <https://doi.org/10.54435/heos.v1i4.28>
17. Dragseth, M. R.: Building student engagement through social media. *Journal of Political Science Education*. 14, 243-256 (2018). <https://doi.org/10.1080/15512169.2018.1550421>

18. Ellis, D. A., Davidson, B. I., Shaw, H., Geyer, K.: Do smartphone usage scales predict behavior? *International Journal of Human-Computer Studies*. 130, 86–92 (2019). <https://doi.org/10.1016/j.ijhcs.2019.05.004>
19. Escobar-Viera, C. G., Shensa, A., Bowman, N. D., Sidani, J. E., Knight, J., James, A. E., Primack, B. A.: Passive and active social media use and depressive symptoms among United States adults. *Cyberpsychology, Behavior, and Social Networking*. 21(7), 437–443 (2018). <https://doi.org/10.1089/cyber.2017.0668>
20. Giannetto, D., Chao, J., Fontana, A.: Gamification in a social learning environment. *Issues in Informing Science and Information Technology*. 10, 195–207 (2013). <https://doi.org/10.28945/1806>
21. Hardy, B. W., Castonguay, J.: The moderating role of age in the relationship between social media use and mental well-being: An analysis of the 2016 General Social Survey. *Computers in Human Behavior*. 85, 282–290 (2018). <https://doi.org/10.1016/j.chb.2018.04.005>
22. Hatchel, T., Negri, S., Subrahmanyam, K.: The relation between media multitasking, intensity of use, and well-being in a sample of ethnically diverse emerging adults. *Computers in Human Behavior*. 81, 115–123 (2018). <https://doi.org/10.1016/j.chb.2017.12.012>
23. Hawi, N. S., Samaha, M.: The relations among social media addiction, self-esteem, and life satisfaction in university students. *Social Science Computer Review*. 35(5), 576–586 (2017). <https://doi.org/10.1177/0894439316660340>
24. Helliwell, J.F., Aknin, L.B.: Expanding the social science of happiness. *Nature Human Behaviour*. 2(4), 248–252 (2018). <https://doi.org/10.1038/s41562-018-0308-5>
25. Junco, R., Elavasky, M., Heiberger, G.: Putting Twitter to the test: Assessing outcomes for student collaboration, engagement, and success. *British Journal of Educational Technology*. 44(2), 273–287 (2012). <https://doi.org/10.1111/j.1467-8535.2012.01284.x>
26. Karahoca, A., Yengin, I.: Understanding the potentials of social media in collaborative learning. *Encyclopedia of Information Science and Technology*. 14, 7168–7180 (2018). <https://doi.org/10.4018/978-1-5225-2255-3.ch623>
27. Keyes, C. L. M.: Happiness, flourishing, and life satisfaction. In: Cockerham, W. C., Dingwall, R., Quah, S. (Eds.), *The Wiley Blackwell Encyclopedia of Health, Illness, Behavior, and Society*. pp. 747–751. John Wiley & Sons, Chichester (2014). <https://doi.org/10.1002/9781118410868.wbehibs454>
28. Kim, B., Kim, Y.: College students' social media use and communication network heterogeneity: Implications for social capital and subjective well-being. *Computers in Human Behavior*. 73, 620–628 (2017). <https://doi.org/10.1016/j.chb.2017.03.033>
29. Kross, E., Verduyn, P., Sheppes, G., Costello, C. K., Jonides, J., Ybarra, O.: Social Media and Well-Being: Pitfalls, Progress, and Next Steps. *Trends Cogn Sci*. 25(1), 55–66 (2021). <https://doi.org/10.1016/j.tics.2020.10.005>
30. le Roux, D. B., Parry, D. A.: Media multitasking and cognitive control: A systematic review of interventions. *Computers in Human Behavior*. 92, 316–327 (2019). <https://doi.org/10.1016/j.chb.2018.11.031>
31. le Roux, D. B., Parry, D. A., Totolo, A., Iyawa, G., Holloway, J., Prenter, A., Botha, L.: Media multitasking, online vigilance, and academic performance among students in three Southern African countries. *Computers & Education*. 160, 104056 (2021). <https://doi.org/10.1016/j.compedu.2020.104056>
32. le Roux, D. B., Parry, D. A.: An exploratory investigation of the use and effects of academic instant messaging groups among university students. *Education and Information Technologies*. 27(1), 1055–1080 (2022). <https://doi.org/10.1007/s10639-021-10631-y>

33. le Roux, D. B., Parry, D. A., Morton, J., Pons, R., Pretorius, R., Schoeman, A.: Digital wellbeing applications: adoption, use and perceived effects. *Computers in Human Behavior*. 139, 107542 (2023). <https://doi.org/10.1016/j.chb.2022.107542>
34. Lanette, S., Chua, P. K., Hayes, G., Mazmanian, M.: How much is 'too much'? The role of a smartphone addiction narrative in individuals' experience of use. In: *Proceedings of the ACM on Human-Computer Interaction*, 2(CSCW), pp. 1-22. (2018). <https://doi.org/10.1145/3274370>
35. Lewin, K. M., Meshi, D., Schuster, A. M., Cotten, S. R.: Active and passive social media use are differentially related to depressive symptoms in older adults. *Aging & Mental Health*, 1-8 (2022). <https://doi.org/10.1080/13607863.2022.2068133>
36. Leyrer-Jackson, J. M., Wilson, A. K.: The associations between social-media use and academic performance among undergraduate students in biology. *Journal of Biological Education*, 52, 221-230 (2018). <https://doi.org/10.1080/00219266.2017.1307246>
37. Mehdinezhad, V.: First-year students' engagement at the university. *International Online Journal of Educational Sciences*. 3(1), 47-66 (2011).
38. Miao, F.F., Koo, M., Oishi, S.: Subjective well-being. In: Boniwell, I., David, S.A., Ayers, A. C. (Eds.), *Oxford Handbook of Happiness*. pp. 174-184. Oxford University Press, Oxford (2013). <https://doi.org/10.1093/oxfordhb/9780199557257.013.0013>
39. Monge Roffarello, A., De Russis, L.: The race towards digital wellbeing: Issues and opportunities. In: *Proceedings of the 2019 CHI conference on human factors in computing systems*, pp. 1-14. (2019). <https://dl.acm.org/doi/abs/10.1145/3290605.3300616>
40. Muktar, B., Sagagi, F. M., Gumel, M. F., Hussein, A.: An application of self-regulated learning theory in assessing effect of ICT mediated academic communication amongst Federal University dutse undergraduate students. *JCCR — Journal of Community & Communication Research*. 5(2), 139-143 (2020). <https://jccr.secdcr.org/index.php/jccr/article/view/81>
41. Naik, G. L., Deshpande, M., Shivananda, D. C., Ajey, C. P., Manjunath Patel, G. C.: Online teaching and learning of higher education in India during COVID-19 emergency lockdown. *Pedagogical Research*. 6(1), 1-14 (2021). <https://doi.org/10.29333/pr/9665>
42. Neier, S., Zayer, L.T.: Students' perceptions and experiences of social media in higher education. *Journal of Marketing Education*. 37(3), 133-143 (2015). <https://doi.org/10.1177/0273475315583748>
43. Papademetriou, C., Anastasiadou, S., Konteos, G., Papalexandris, S.: COVID-19 pandemic: the impact of the social media technology on higher education. *Education Sciences*. 12(4), (2022). <https://doi.org/10.3390/educsci12040261>
44. Phu, B., Gow, A. J.: Facebook use and its association with subjective happiness and loneliness. *Computers in Human Behavior*. 92, 151-159 (2019). <https://doi.org/10.1016/j.chb.2018.11.020>
45. Przybylski, A. K., Murayama, K., DeHaan, C. R., Gladwell, V.: Motivational, emotional, and behavioral correlates of fear of missing out. *Computers in human behavior*. 29(4), 1841-1848 (2013). <https://doi.org/10.1016/j.chb.2013.02.014>
46. Putnik, G., Costa, E., Alves, C., Castro, H., Varela, L., Shahl, V.: Analysing the correlation between social network analysis measures and performance of students in social network-based engineering education. *International Journal of Technology and Design Education*. 26, 413-437 (2015). <https://doi.org/10.1007/s10798-015-9318-z>
47. Rogers, A. P., Barber, L. K.: Addressing FoMO and telepressure among university students: Could a technology intervention help with social media use and sleep disruption? *Computers in Human Behavior*. 93, 192-199 (2019). <https://doi.org/10.1016/j.chb.2018.12.016>

48. Romero, S., Fardoun, H. M., Penichet, V. M. R., Lozano, M. D., Gallud, J. A.: Analysis of online social interactions based on positive reinforcement social networks in a k-12 geometry class. *Applied Sciences* (Switzerland). 11(23), (2021). <https://doi.org/10.3390/app112311545>
49. Saldana, J.: *The Coding Manual for Qualitative Researchers* (2nd ed.). London, Sage (2013).
50. Schodt, K. B., Quiroz, S. I., Wheeler, B., Hall, D. L., Silva, Y. N.: Cyberbullying and mental health in adults: the moderating role of social media use and gender. *Frontiers in psychiatry*. 12, (2021). <https://doi.org/10.3389/fpsyt.2021.674298>
51. Sternberg, N., Luria, R., Chandhok, S., Vickers, B., Kross, E., Sheppes, G.: When Facebook and finals collide-procrastinatory social media usage predicts enhanced anxiety. *Computers in Human Behavior*. 109, 106358 (2020). <https://doi.org/10.1016/j.chb.2020.106358>
52. Tarisayi, K.S., Munyaradzi, E.: A simple solution adopted during the Covid-19 pandemic: Using WhatsApp at a university in Zimbabwe. *Issues in Educational Research*. 31(2), 644-659 (2021). <https://search.informit.org/doi/10.3316/informit.053265997263512>
53. Tesfamicael, S. A., Ayalew, Y.: Mathematics education in Ethiopia in the era of COVID-19: Boosting equitable access for all learners via opportunity to learning. *Contemporary Mathematics and Science Education*. 2(1), ep21005 (2021). <https://doi.org/10.30935/conmaths/9680>
54. Udenze, S., Oshionebo, B.: Investigating Whatsapp for collaborative learning among undergraduates. *Etkileşim*. (5), 24-50 (2020). <https://doi.org/10.32739/etkilesim.2020.5.92>
55. Usman, Y.D., Bukar, A.: Students' Usage of WhatsApp Instant Messenger as a Supporting Tool for Learning in Kaduna State, Nigeria. *International Journal of Education and Development using Information and Communication Technology*. 18(3), 147-158 (2022).
56. Vannucci, A., Flannery, K. M., Ohannessian, C. M.: Social media use and anxiety in emerging adults. *Journal of affective disorders*, 207, 163-166 (2017). <https://doi.org/10.1016/j.jad.2016.08.040>
57. Van Den Beemt, A., Thurlings, M., Willems, M.: Towards an understanding of social media use in the classroom: a literature review. *Technology, Pedagogy and Education*. 29(1), 35-55 (2020). <https://doi.org/10.1080/1475939X.2019.1695657>
58. Van Den Berg, G., Mudau, P.K.: Postgraduate students' views on the use of WhatsApp groups as an online communication tool to support teaching and learning during COVID-19. *Perspectives in Education*. 40(1), 112-128 (2022). <https://doi.org/10.18820/2519593X/pie.v40.i1.7>
59. Van Dijk, J.: *The Digital Divide*. Cambridge: Polity Press (2020)
60. Van Koningsbruggen, G. M., Hartmann, T., Du, J.: Always on? Explicating impulsive influences on media use. In: *Permanently online, permanently connected*, pp. 51-60. Routledge, (2017)
61. Wang, H. Z., Yang, T. T., Gaskin, J., Wang, J. L.: The longitudinal association between passive social networking site usage and depressive symptoms: The mediating role of envy and moderating role of life satisfaction. *Journal of Social and Clinical Psychology*. 38(3), 181-199 (2019). <https://doi.org/10.1521/jscp.2019.38.3.181>
62. Welch, B., Bonnan-White, J.: Twittering to increase student engagement in the university classroom. *Knowledge Management E-Learning: An International Journal*. 4(3), 325-345 (2012). <https://doi.org/10.34105/j.kmel.2012.04.026>
63. Yeboah, D., Nyagorme, P.: Students' acceptance of WhatsApp as teaching and learning tool in distance higher education in sub-Saharan Africa. *Cogent Education*. 9(1), p.2077045 (2022). <https://doi.org/10.1080/2331186X.2022.2077045>

64. Yirci, R., Karakose, T., Malkoç, N.: Examining the Influence of Cyberbullying Perpetration and Victimization among High School Adolescents—Associations with Gender and Grade Level. *Educational Process International Journal*. 10(4), (2021). <https://doi.org/10.22521/edupij.2021.104.4>
65. Zachos, G., Paraskevopoulou-Kollia E-A., Anagnostopoulos, I.: Social media usage in higher education: A review. *Education Sciences*. 8(4), 1-13 (2018). <https://doi.org/10.3390/educsci8040194>
66. Zulkanain, N. A., Miskon, S., Syed Abdullah, N.: An adapted pedagogical framework in utilizing WhatsApp for learning purpose. *Education and Information Technologies*. 25(4), 2811–2822 (2020). <https://doi.org/10.1007/s10639-019-10096-0>

Increasingly Forgetful and Distracted? A cross-temporal meta-analysis of everyday cognitive failures among university students

Daniel B le Roux¹[0000–0001–9682–0377], B Finsterbusch¹, J Saunders¹, and B De Agrela

Stellenbosch University
dbleroux@sun.ac.za

Abstract. High levels of online media use have become the norm on university campuses across the world, engendering behavioural patterns characterised by habitual, rapid switching between media and non-media activities. In a growing body of literature, scholars from various fields have expressed concern about the potential negative effects of this form of behaviour on attention control and academic performance. Instructors, accordingly, perceive the current generation of students as distracted and absent-minded. In the present study, we investigate whether this is the case - Does the current cohort of university students display higher levels of distractibility than the previous cohorts? To address this question, we conduct a cross-temporal meta-analysis of the Cognitive Failures Questionnaire (CFQ) administered among a total sample of 7294 healthy university students across eight countries between 1999 and 2020. Our findings indicate a small, positive effect of time on CFQ scores, indicating higher distractibility in more recent years. Although this finding may reflect the negative effects of media use patterns on executive functioning, it may also be the result of a variety of other factors.

Keywords: digital distraction · cognitive failures · cross-temporal meta-analysis · media multitasking · students

1 Introduction

The omnipresence of mobile computing devices, such as smartphones, tablets, and laptops, has become a defining feature of higher education contexts across both developed and developing countries [12]. These devices enable students to maintain permanent online connectivity in different life contexts and engender an always-on culture characterised by continuous engagement with online content and communication [30]. An important dimension of always-on culture is that it involves frequent switches between different streams of attentional stimuli as students juggle diverse online and offline activities. Under the banner of *media multitasking*, a rapidly growing body of research has investigated the antecedents, nature and effects of this form of behaviour [22], with evidence indicating that

it is common among university students both in general and academic settings, such as lectures and study sessions [12].

While the ability to multitask effectively is often framed as a strength of digital natives, there is concern among scholars that a habit of constant task switching may harm learning outcomes in various ways [9]. Central to these concerns is the argument that when students multitask with media, their attention is repeatedly distracted from their ongoing academic tasks. This line of reasoning is supported by studies that have found negative associations between media multitasking (both within and outside the classroom) and academic performance [11, 41, 12]. Furthermore, there is evidence of a negative association between media multitasking and general cognitive control ability [21, 42, 22]. The exact nature of these effects is a matter of ongoing debate among scholars, but there is broad agreement that frequent media multitaskers find it more difficult than their peers to remain focused on a particular task for an extended period of time (see, e.g., Le Roux and Parry [13]). Consistent with these findings, qualitative studies suggest that instructors and lecturers perceive the current generation of university students as inattentive and distracted as a result of the constant presence of online media in academic settings [4].

Although these findings paint a somewhat alarmist picture of a highly distractible student population, two key points of uncertainty characterise the current body of knowledge in the domain. First, in all existing evidence, the observed effect sizes are typically small and scholars have expressed doubts about their practical significance [22]. Second, there is no strong evidence to suggest that the association between media multitasking and distractibility is causal in nature; it may merely indicate that distractible students tend to multitask with media more frequently than their less distractible peers. Given these uncertainties, the perception that the current generation of university students display, on average, higher levels of distractibility, forgetfulness and inattention than earlier generations is called into question.

In the present study we investigate this question by conducting a cross-temporal meta-analysis to determine whether there has been an increase in distractibility, forgetfulness and attentional lapses among university students in the period between 2000 and 2020. To this end, we used the concept of cognitive failures and its associated measurement instrument, the Cognitive Failures Questionnaire (CFQ) introduced by Broadbent et al. [3], and analysed the changes in this measure among healthy university students throughout the relevant period of time.

1.1 Digital Distraction, Media Multitasking, and Cognitive Control

To make sense of the interaction between digital media use patterns and cognitive control, it is useful to distinguish between two levels of analysis. At the basic level, cognitive control is affected by digital media when ongoing tasks are interrupted by the use of a digital device. This phenomenon is often referred to as digital distraction and involves the switching of attention away from one stream of stimuli (e.g., a homework assignment) towards another (e.g., social

media). Agrawal et al. [1] define digital distraction as “distraction due to electronic devices and media that breaks the concentration from the main piece of work that is being done” (p. 91) and argues that users of digital devices are “constantly distracted by emails, pings, texts, etc. which prevents them from completing their work-related tasks” (p. 91). In their study of this phenomenon in college classrooms, Flanigan and Babchuk [4] report that instructors typically share the belief that “digital distraction decreases student engagement and comprehension of lecture content which, subsequently, reduces student learning and achievement” (p. 8). Their findings further indicate that digital distraction is a common phenomenon among students across various world regions; that it is linked to diminished learning and achievement; and that it has the potential to erode the student-instructor relationship if it is not adequately managed in the classroom.

While digital distraction describes *in-the-moment* instances of task interruption, there is also, on a second level, concern about the possibility that a habit of continuous, rapid switching among different tasks may lead to deficits in executive function. Executive function describes the human ability to modulate attention in a top-down manner (i.e., concentrate on something) and is based on a combination of important mental processes. While different conceptualisations exist, there is broad agreement that three key functions combine to enable attention modulation: shifting, working memory and inhibition [19]. Shifting refers to the ability to switch from one task or mental set to another, while working memory enables updating and monitoring information in memory. Lastly, inhibition “concerns one’s ability to deliberately inhibit dominant, automatic, or prepotent responses when necessary” [19, p. 57]. Across studies of the interaction between media multitasking and executive function, findings generally indicate a small, negative association [38]. Parry and le Roux [22], in their recent meta-analysis of findings in this domain over the past decade, report, firstly, that the observed association is generally stronger in studies utilising self-report measures of executive function than those utilising task-based measures taken in laboratory settings (e.g., stroop, flanker); and, secondly, that there remains a lack of evidence that this association is causal in nature.

Interpretations of these findings can generally be grouped into two categories. The first posits that the association reflects the attentional distribution strategies of people that frequently multitask with media. It is argued, consequently, that individuals with a general tendency to distribute their attention more broadly (i.e., pay attention to a greater range of stimuli in their environment) are more likely to become aware of or orientate themselves towards events in their online spheres as they are continuously on the lookout for new or potentially interesting cues [13, 28]. From this perspective the observed association between media multitasking and executive function is not causal, but rather explained by personal traits of individuals — i.e., broader attention distribution in media-saturated environments leads to more media multitasking.

An alternative interpretation posits that there may be some form of causal mechanism by which high levels of media multitasking, over an extended period,

deteriorate executive function. Unlike the first interpretation, which concerns attention distribution strategy, the second considers the potential that our ability to effectively control attention may be harmed by a pattern of behaviour characterised by frequent attentional shifting rather than focussed concentration [28]. However, there is a lack of evidence to support this interpretation. In one of few studies which have aimed to test it, Parry et al. [23] limited a group of university students to a maximum of 90 minutes of smartphone use per day over the course of a month, down from the four-hour average observed in the target population. In follow-up cognitive tests participants displayed no improvement in cognitive control ability. However, the authors note that a longer period of manipulation and/or greater restriction of use volumes may be required to rule out test for causal effects.

While interpretations in the first category focus on attention distribution strategy, those in the second category focus on the deterioration of the ability to focus attention. A more recent theorisation of the relationship between media multitasking and executive function partly straddles these two categories. Reinecke et al. [30] propose that, over time, our media use patterns cultivate *online vigilance* — a “trait-like” (p. 7) variable that describes an individual’s “chronic attention for and a permanent proclivity to respond to incoming cues from the online sphere” (p. 5) to prioritise these over other activities. Although the notion of online vigilance does not imply deterioration of executive function ability, it acknowledges the causal relationship between media use behaviour and attention distribution. Reinecke et al. [30] argue that the multitude of gratifications offered by online media represent a reliable source of intrinsic need satisfaction. Consequently, high levels of media use, over time, lead to “automatic stimulus-response reactions as well as deliberately controlled, goal-directed behaviour based on the learned causal relationship between action and desired outcomes” (p. 4). Unlike a general tendency for broad attention distribution, which may not be determined by media use behaviour, online vigilance presents an attentional strategy which favours the online sphere specifically. Among young adults, online vigilance has been shown to negatively predict academic performance, affective well-being and mindfulness [12, 8, 30, 32].

1.2 The Present Study

Given the extant uncertainty about the nature of the relationship between media use behaviour and cognitive control, questions remain over the general levels of distractibility among the current generation of university students and, specifically, whether these are significantly different from those of earlier generations. Cross-sectional research designs that have been adopted in the majority studies in this domain are poorly suited to address this question. It requires, rather, a method that enables the comparison of generations on the basis of some indicator of general distractibility.

A widely adopted indicator of distractibility in everyday, real-world settings is the frequency of experiences of “cognitive failures” - episodes experienced by normal, healthy individuals which involve “perceptual failures, or failures of

memory, or actions which are misdirected” [3, p. 1]. These failures are typically characterised as attentional lapses or instances of absent-mindedness and often result in mistakes made in basic tasks. To measure the frequency at which individuals experience such failures, Broadbent et al. [3] developed the Cognitive Failures Questionnaire (CFQ) consisting of 25 question with Likert scale responses. Each question asks the respondent how frequently they experience a particular type of attentional lapse. For example, “Do you fail to notice signposts on the road?”; or “Do you find you forget appointments?” Response options include very often, quite often, occasionally, very rarely and never on a scale ranging from 0 to 4. To calculate a total score for the CFQ the responses to all items are summed to produce a scale ranging from 0 to 100. A number of subsequent studies have investigated the dimensions and correlates of the CFQ with partially divergent results [15, 26, 10, 40]. However, the four factors identified by Wallace et al. [40] have been widely adopted over the past two decades. These include memory (“memory failures and forgetfulness”), distractibility (“perceptual aspects of divided-attention tasks”), blunders (failures of motor function) and names (memory failures of names or proper nouns).

In the present study, we adopt the proposition that any potential impact that online media use may have had on general levels of distractibility among university students, would be reflected in a change in CFQ measures in this population over the time period during which high levels of online media use became the norm. The rapid and widespread adoption of mobile media devices, and smartphones in particular, have occurred in a fairly short and well-demarcated timeframe. Although comparable devices were available earlier, it is generally acknowledged that the launch of the iPhone in 2007 marks the beginning of broad smartphone adoption. In 2011 around 35% of American adults owned a smartphone, a number which has increased to 85% more recently [25]. Additionally, a strong body evidence indicates that the volumes of online media and smartphone use have steadily increased during this period, particularly among adolescents and university students. In a 2011 study, US students ($n = 1026$) reported spending an average of just over 30 minutes e-mailing, approximately 10 minutes chatting or instant messaging, and approximately 45 minutes talking on the phone or text messaging [7]. In a more recent study that used tracking data rather than self-reported measures, Andrews et al. [2] report that “participants used their phones a mean of 84.68 times each day ($SD = 55.23$) and spent 5.05 hours each day using their smartphone ($SD = 2.73$)”.

2 Materials and Methods

Cross-temporal meta-analysis (CTMA) adopts the perspective that an individual’s birth cohort can be used as “a proxy for the larger sociocultural environment”, allowing exploration of its impacts on aspects such as personality and behaviour [37, p. 1008]. Consequently, CTMAs often involve the investigation of temporal changes in a particular measure among a particular age group. While CTMAs have been utilised across various domains, important questions concern-

ing the method’s validity have been raised. Central among these is the problem of measurement equivalence [35]. Due to changes in cultural norms over time, measurement items may be interpreted differently by respondents from different birth cohorts. Consequently, the instrument may fail to measure the same latent construct at different points in time. Because the CFQ measures the frequency of everyday events that are largely personal in nature, it is arguably less sensitive to changes in cultural norms than, for example, measures for personality constructs like conscientiousness. Nevertheless, when considering the 25 items in the CFQ scale, there is at least one which refers to an event which is less likely to occur in current times than in the 1980s. Item 11 asks the respondent to indicate how frequently they “leave important letters unanswered for days” [3]. Respondents who interpret the word “letters” narrowly (i.e., a paper-based letter sent via traditional post) are likely to indicate a very low frequency for this item. With the exception of this item, we argue that the CFQ is likely to resist measurement inequivalence between the 1980s and the present day.

In a more recent critique of CTMAs, Rudolph et al. [31, p. 734] argue that a primary challenge for the method is to clearly separate the effects of age, period and cohort. Age effects refer to “biological or social differences attributable to physical or psychological maturation, life stage, or development”, while period effects refer to “the influence of contemporaneous time and thus reflect variation among individuals based on the impact of historical events that affect people across ages”. Finally, cohort effects refer to “differences between groups of individuals that are attributable to their membership in that group”, with groups defined by properties like birth year or nationality.

Considering the distinction between these effect types, we argue that the adoption and increased use of smartphones (and online media in general) have not been limited to particular age groups [34]. Although younger generations may have been exposed to digital devices at an earlier age, any potential effects that the use of these devices may have had on cognitive functioning should be observable across age cohorts in which adoption occurred. It should be noted, however, that studies of the interaction between age and CFQ outcomes generally support the premise that aging influences cognitive failures, but that its effect is not the same across the different factors. For example, Rast et al. [29] found that while forgetfulness increases with age, distractibility remains relatively stable between 24 and 63 years, but drops sharply thereafter, and false triggering remains stable throughout the lifespan. Given these findings, the identification of any potential effects of media use patterns on cognitive failures should control for the effects of aging.

In addition to controlling for age-related effects, it is important to acknowledge that the adoption and use of digital media technologies have not occurred simultaneously across world regions. Hence, while it can be framed as a historical event which has affected people across age groups, different populations have experienced (and continue to experience) this event at different times. For example, data from the World Bank [43] indicate that in 2010 71.7% of the US population, 85% of the UK population and 80.3% of the Canadian population used the

Internet. By contrast, in the same year, only 47.9% of the Chinese population and 7.5% of the Indian population used the internet. Given these differences, any potential effects of the use of online media on individual outcomes would be observable at different points in time in different countries.

2.1 Literature Search

To find studies in which the CFQ was used among student populations, query-based searches were conducted in four research databases: Scopus, EBSCOHost, ProQuest and Web Of Science. Keywords used included “cognitive failures questionnaire” and “students”. To target the period characterised by the rapid rise in the volumes of online media use, searches were limited to dates of publication no earlier than 1 January 2000. While the year 2000 predates broad adoption of smartphones, internet penetration rates were already around or over 30% in many developed countries [43] with prominent social media platforms like Facebook and Twitter emerging in 2004 and 2006 respectively. Our searches produced a combined total of 992 published reports (see Figure 1).

2.2 Inclusion Criteria

To identify studies for inclusion in the analysis, the following criteria were applied. Studies were only included if they were a) published in English; b) reported primary data collected among a sample of university students with a mean age for the sample in the range of 18 to 25 years; c) utilised the CFQ as a measure and reported its mean, standard deviation, sample size and female representation. In cases where the relevant data points were not reported or there was a lack of clarity regarding some aspect of the measurements instrument utilised (e.g., scale indicator range not specified), the authors were contacted to obtain the relevant information. To control for selection bias due to small sample sizes [14], studies were only included if the CFQ mean was based on a sample of at least 50 students. Finally, we only included studies if data were collected in countries that displayed heterogeneity in terms of internet penetration rates. In this regard we utilised internet penetration data from the World Bank and used 2010 as a point of comparison. Using the United States, United Kingdom and Canada as baseline markers, studies were excluded if the internet penetration rate in the country of data collection was below 60% in 2010.

2.3 Extraction

Data extraction procedures involved recording the following variables for each included study: CFQ mean, CFQ standard deviation, sample size, proportions of females in the sample, year of data collection, and the country in which the sample population resided. In the cases where measures were taken as part of an experimental study, we extracted baseline scores taken prior to treatment or manipulation. In the cases where CFQ means and/or standard deviations were

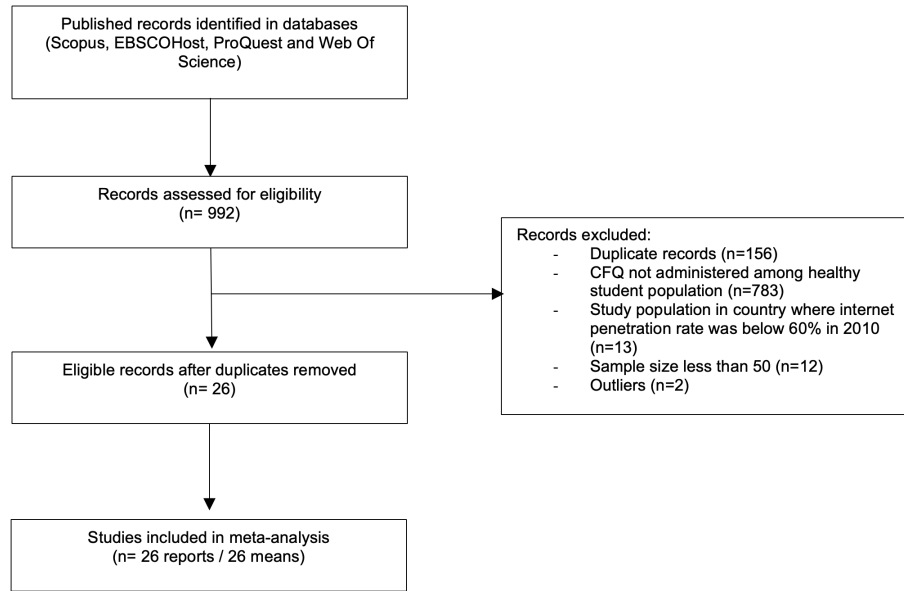


Fig. 1. Literature search and inclusion process.

reported for subgroups of the sample rather than the entire sample, the total means and standard deviations were calculated using the formula prescribed by the Cochrane Handbook for Systematic Reviews of Interventions [6].

2.4 Coding

After completion of data extraction from eligible studies there were 28 CFQ means from 27 reports included in the set. Most studies reported the means of overall CFQ scores calculated by adding the 25 items in the instrument, each measured on a scale ranging from 0 to 4. However, in some cases a scale ranging from 1 to 5 was used (e.g., Wilkerson et al., 2012), in which case the mean was adjusted by subtracting 25. Other studies reported the means of the item responses rather than the sum of the responses (e.g., Horvat and Tement., 2020), in which case the reported mean and standard deviation were multiplied by 25. In cases where the study did not provide a clear indication of how the means were determined, the lead author was contacted for more information. Although we managed to obtain the relevant information in a majority of cases, some authors did not respond or could not confirm the scale ranges used. To mitigate the potential impact of such cases on our results, we utilised the method proposed by Seo [33] to remove outliers. Consequently, studies where the reported CFQ mean differed by more than two standard deviations from the mean across all studies in the sample were removed ($n = 2$). The final dataset included 26 CFQ

means from 25 reports. If not reported in the text, the year of data collection was coded as two years prior to the year of publication.

3 Results

Of the 26 means in the final sample, 14 were obtained from samples in the United States, four from Italy, two each from Australia and the United Kingdom, and one each from Canada, Ireland, Slovenia and The Netherlands. Figure 2 presents a graph of the CFQ means for the 26 studies included in the final sample, with the country and sample size indicated, and data collection years ranging from 1999 to 2020. Across the 26 included studies, the mean CFQ score (non-weighted) is 42.56 ($SD = 4.41$) with the highest (51.46, $n = 106$) measure taken in 2016 and the lowest (31.07, $n=59$) in 2003.

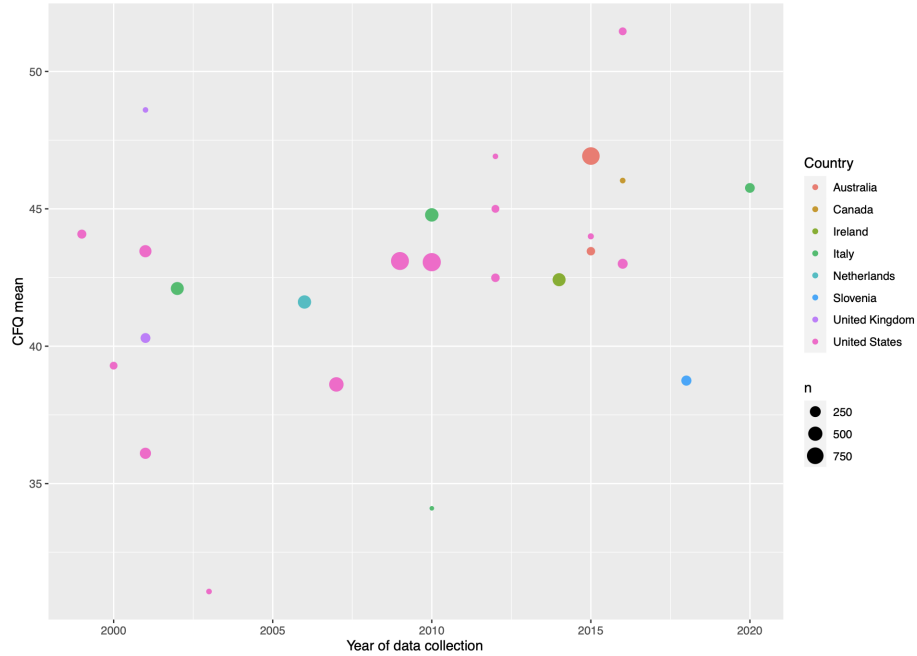


Fig. 2. A scatter plot of the results from included studies.

For each of the years covered in the sample, with the exception of those for which no data was available (2004, 2005, 2008, 2011, 2013, 2017, 2019), weighted means for the CFQ scores were calculated based on sample size, and the associated proportion of female representation was determined. The results of these calculations are presented in Table 1. The weighted means range from 31.1 ($n = 59$) in 2003 to 46.3 ($n = 1054$) in 2015.

Table 1. Summary of data by year of data collection

Year of data collection	CFQ (weighted mean)	% female	n
1999	44.1	70.2	151
2000	39.3	69.9	103
2001	40.8	52.1	839
2002	42.1	36.7	390
2003	31.1	69.9	59
2006	41.6	69.0	413
2007	38.6	62.5	523
2009	43.1	64.9	909
2010	43.2	60.8	1423
2012	44.3	48.5	293
2014	42.4	73.0	396
2015	46.3	71.3	1054
2016	46.0	71.1	359
2018	38.8	84.0	201
2020	45.8	86.7	181

Following the method used by [5] we conducted a robust regression with M-estimation using R package MASS (UCLA: Statistical Consulting Group, n.d.; [39]) and used unstandardised regression coefficients to interpret the results. Our model predicted CFQ means (weighted by sample size) using data collection year and proportion of females as predictor variables. Table 2 summarises the results of the model, indicating that data collection year is a positive predictor of CFQ scores ($b = 0.34, p < 0.01, 95\%CI[0.16, 0.5]$), while proportion of females in the sample does not significantly predict CFQ scores ($b = -0.06, p = 0.23, 95\%CI[-0.15, 0.03]$).

Our data did not include enough measures from countries other the US to enable meaningful comparison across countries. Nonetheless, to test whether the non-US data impacted our results, we tested the same model with US data only. Not surprisingly, the results changed very little with year of data collection remaining a positive predictor of CFQ scores ($b = 0.37, p < 0.01, 95\%CI[0.22, 0.51]$), and proportion of females remaining non-significant ($b = -0.03, p = 0.6, 95\%CI[-0.13, 0.07]$).

Table 2. Results of a robust regression model predicting CFQ means.

	b	SE	p	95% CI
Intercept	-629.46	177.87	<0.01	-978.07, -280.85
Year	0.34	0.10	<0.01	0.16, 0.5
Proportion female	-0.06	0.05	0.23	-0.15, 0.03

4 Discussion

In the present study we tested the proposition that self-reported cognitive failures in everyday life increased among university students in the period between 2000 and 2020. To this end, we conducted a cross-temporal meta-analysis of Broadbent et al. [3] cognitive failures questionnaire for the relevant time period. While a variety of factors may potentially underlie changes in the frequency with which cognitive failures are experienced, we base the proposition on extant evidence of the negative association between media multitasking behaviour and executive function [38, 22]. We propose, accordingly, that, given this association, the increased levels of media use and media multitasking observed within the relevant time period may have resulted in an increase in experiences of everyday cognitive failures. Our findings indicate, firstly, that the average CFQ score among healthy university students during this period was 42.56 and secondly, that scores increased by 0.34 per year over the time period. Moreover, weighted averages higher than 45 were observed in 2015, 2016 and 2020. While relatively small, this effect was statistically significant in our model.

To make sense of the potential practical significance of the observed effect, it is useful to consider our findings in relation to the CFQ scores observed among students who have been diagnosed with attention-related problems. Mawjee et al. [16], in their investigation of working memory among Canadian students diagnosed with ADHD (Attention-Deficit/Hyperactivity Disorder), report mean CFQ scores ranging between 57 and 59. This suggests, firstly, that, while there may have been an increase in cognitive failures among healthy university students, their average scores are still substantially lower than those of students diagnosed with ADHD. Secondly, it suggests that, if the rate of change observed in our findings remained unchanged, healthy students would experience cognitive failures as frequently as students with ADHD in around 15 years. The present study, however, provides little grounds to argue whether or not the observed trend would persist. It is possible that students will, over time, become aware of the negative effects of media multitasking on their cognitive performance and adapt their behaviour accordingly. Alternatively, they may adapt in other ways, both behaviourally and/or cognitively, to cope with or counter these effects.

Notwithstanding these uncertainties, it is important to recognise the implications of the increasing frequency of cognitive failures for the higher education environment. For students, these implications may involve reduced performance in a range of everyday task domains, including academic performance. In line with extant evidence of the negative association between media multitasking and academic performance observed in recent studies [12], higher frequency of cognitive failures may explain the difficulties students report when trying to maintain attentional focus during classes or study sessions [12]. A meaningful way to investigate this possibility in future work would be to test the hypothesis that cognitive failures mediate the relationship between media use behaviour and academic performance.

For instructors, increased levels of distractibility among students may present a range of pedagogical challenges. Central to these is the management of digital

device use in formal academic settings like lectures or tutorial sessions. While there have been interventions aimed at banning or limiting the use of digital devices in classrooms (see, e.g., [24]), the general trend has been to accept a gradual shift toward the presence of more digital technologies in class. Rather than seeking ways to limit the use of digital devices, lecturers are looking for innovative ways to harness their potential to enhance the learning environment [27]. Torres [36], accordingly, argues that “promoting the integration of supportive technologies that aid learning is imperative to avoid disruptions and to provide opportunities focused on the endorsement of technology enhanced environments that result in student engagement and success” (p. 223).

While acknowledging the potential value of strategies aimed at embracing technology as part of the learning process, it is important not to conflate anti-technology and pro-attention perspectives, nor to see these as mutually exclusive. The guiding principle that effective cognitive control is an essential precondition for learning should dictate that any pedagogical strategy, whether embracing or limiting technology use, should foster student’s ability to exercise top-down attentional control. In this sense, interventions such as mindfulness practises at the start of classes (see, e.g., Miller et al. [18]) may be useful strategies to cultivate attitudes and behaviours that promote effective attention management among students.

4.1 Limitations and future work

Although we based our proposition that CFQ scores would increase on evidence of the association between media multitasking and executive function, we are mindful of the wide range of other factors that may explain our findings. For example, a positive association has been observed between depressive symptoms and cognitive failures [17], and a growing body of literature has reported an increase in stress, anxiety, and depression among university students [20]. We are hesitant, accordingly, to single out the increased volume of online media use and media multitasking as drivers and acknowledge that it may be one of a range of interrelated factors at play in the relevant time period. Additionally, our findings do not allow for distinction between the strategic and deficit-producing hypotheses, nor do they provide support for causality between media multitasking and cognitive failures.

The small number of reports included in the final sample and the over representation of data from the US limit the quality of our results. While we searched extensively for reports that met the inclusion criteria, it is possible that we may have inadvertently excluded reports. Lastly, by controlling for age effects and focussing only on university students, our findings do not speak to the possibility that the same increase would be observable within younger or older age groups, or non-student populations in the same age group. We propose that, if media use patterns do indeed influence cognitive failures, the same increase should be observable in other groups, and we call for future studies to test this proposition.

4.2 Conclusion

Findings of the present study indicate that, in the period between 2000 and 2020, self-reported cognitive failures in everyday life increased among university students. Based on the arguments set out in this article, we conclude that, while a range of factors may underlie the observed increase, the role of increased volumes of online media use and media multitasking among students should be considered as one of the key drivers of this increase. This conclusion is based on extant evidence of negative associations between, firstly, media multitasking and executive function and, secondly, media multitasking and academic performance.

References

1. Agrawal, P., Sahana, H.S., De', R.: Digital Distraction. In: Proceedings of the 10th International Conference on Theory and Practice of Electronic Governance. pp. 191–194. ICEGOV '17, Association for Computing Machinery, New York, NY, USA (Mar 2017)
2. Andrews, S., Ellis, D.A., Shaw, H., Piwek, L.: Beyond Self-Report: Tools to Compare Estimated and Real-World Smartphone Use. *PLOS ONE* 10(10), e0139004 (Oct 2015)
3. Broadbent, D.E., Cooper, P.F., FitzGerald, P., Parkes, K.R.: The Cognitive Failures Questionnaire (CFQ) and its correlates. *British Journal of Clinical Psychology* 21(1), 1–16 (1982)
4. Flanigan, A.E., Babchuk, W.A.: Digital distraction in the classroom: exploring instructor perceptions and reactions. *Teaching in Higher Education* 27(3), 352–370 (Apr 2022), publisher: Routledge
5. Hamamura, T., Johnson, C.A., Stankovic, M.: Narcissism over time in Australia and Canada: A cross-temporal meta-analysis. *Personality and Individual Differences* 155, 109707 (Mar 2020)
6. Higgins, J.P.T., Thomas, J., Chandler, J., Cumpston, M., Li, T., Page, M.J., Welch, V.A.: *Cochrane Handbook for Systematic Reviews of Interventions*. John Wiley & Sons (Sep 2019), Google-Books-ID: cTqyDwAAQBAJ
7. Jacobsen, W.C., Forste, R.: The wired generation: Academic and social outcomes of electronic media use among university students. *Cyberpsychology, Behavior, and Social Networking* 14, 275–280 (2011)
8. Johannes, N., Meier, A., Reinecke, L., Ehlert, S., Setiawan, D.N., Walasek, N., Dienlin, T., Buijzen, M., Veling, H.: The relationship between online vigilance and affective well-being in everyday life: Combining smartphone logging with experience sampling. *Media Psychology* 24(5), 581–605 (May 2020), publisher: Routledge
9. Kirschner, P.A., De Bruyckere, P.: The myths of the digital native and the multitasker. *Teaching and Teacher Education* 67, 135–142 (Oct 2017)
10. Larson, G.E., Alderton, D.L., Neideffer, M., Underhill, E.: Further evidence on dimensionality and correlates of the Cognitive Failures Questionnaire. *British Journal of Psychology* 88(1), 29–38 (1997)

11. Le Roux, D.B., Parry, D.A.: In-lecture media use and academic performance: Does subject area matter? *Computers in Human Behavior* 77, 86–94 (Dec 2017)
12. Le Roux, D.B., Parry, D.A., Totolo, A., Iyawa, G., Holloway, J., Prenter, A., Botha, L.: Media multitasking, online vigilance and academic performance among students in three Southern African countries. *Computers & Education* 160, 104056 (Jan 2021)
13. Le Roux, D., Parry, D.: Investigating differences in the attention distribution strategies of high and low media multitaskers through a two-dimensional game. *Cyberpsychology* 13(3) (2019)
14. Leuffen, D.: Case Selection and Selection Bias in Small-n Research. In: Gschwend, T., Schimmelfennig, F. (eds.) *Research Design in Political Science: How to Practice What They Preach*, pp. 145–160. Palgrave Macmillan UK, London (2007)
15. Matthews, G., Coyle, K., Craig, A.: Multiple factors of cognitive failure and their relationships with stress vulnerability. *Journal of Psychopathology and Behavioral Assessment* 12(1), 49–65 (Mar 1990)
16. Mawjee, K., Woltering, S., Tannock, R.: Working Memory Training in Post-Secondary Students with ADHD: A Randomized Controlled Study. *PLoS One* 10(9) (Sep 2015), place: San Francisco Publisher: Public Library of Science
17. Merckelbach, H., Muris, P., Nijman, H., deJong, P.: Self-reported cognitive failures and neurotic symptomatology. *Personality and Individual Differences* 20(6), 715–724 (Jun 1996)
18. Miller, C.J., Elder, K., Scavone, A.: The feasibility of bringing brief mindfulness-based training to the university classroom. *Mindfulness* 8(4), 1047–1054 (2017), place: Germany Publisher: Springer
19. Miyake, A., Friedman, N.P., Emerson, M.J., Witzki, A.H., Howerter, A., Wager, T.D.: The Unity and Diversity of Executive Functions and Their Contributions to Complex “Frontal Lobe” Tasks: A Latent Variable Analysis. *Cognitive Psychology* 41(1), 49–100 (Aug 2000)
20. Mofatteh, M.: Risk factors associated with stress, anxiety, and depression among university undergraduate students. *AIMS Public Health* 8(1), 36–65 (Dec 2020)
21. Ophir, E., Nass, C., Wagner, A.D.: Cognitive control in media multitaskers. *Proceedings of the National Academy of Sciences* 106(37), 15583–15587 (Sep 2009), publisher: National Academy of Sciences Section: Social Sciences
22. Parry, D.A., le Roux, D.B.: “Cognitive control in media multitaskers” ten years on: A meta-analysis. *Cyberpsychology: Journal of Psychosocial Research on Cyberspace* 15(2) (Apr 2021), number: 2
23. Parry, D.A., le Roux, D.B., Bantjes, J.R.: Testing the feasibility of a media multitasking self-regulation intervention for students: Behaviour change, attention, and self-perception. *Computers in Human Behavior* 104, 106182 (Mar 2020)
24. Parry, D.A., le Roux, D.B., Cornelissen, L.A.: Managing in-lecture media use: the feasibility and value of a split-class policy. *Journal of Computing in Higher Education* 32(2), 261–281 (Aug 2020)

25. Pew Research: *Demographics of Mobile Device Ownership and Adoption in the United States* (2021), <https://www.pewresearch.org/internet/fact-sheet/mobile/> [Accessed: 12 May 2022]
26. Pollina, L.K., Greene, A.L., Tunick, R.H., Puckett, J.M.: Dimensions of everyday memory in young adulthood. *British Journal of Psychology* 83(3), 305–321 (1992)
27. Pérez-Juárez, M.A., Aguiar-Pérez, J.M., Del-Pozo-Velázquez, J., Alonso-Felipe, M., Rozada-Raneros, S., Conde, M.B.: *From Digital Distraction to Digital Motivation: Utopia or Reality* (2022), ISBN: 9781799892434 Pages: 205–222 Publisher: IGI Global
28. Ralph, B.C.W., Thomson, D.R., Cheyne, J.A., Smilek, D.: Media multitasking and failures of attention in everyday life. *Psychological Research* 78(5), 661–669 (Sep 2014)
29. Rast, P., Zimprich, D., Van Boxtel, M., Jolles, J.: Factor Structure and Measurement Invariance of the Cognitive Failures Questionnaire Across the Adult Life Span. *Assessment* 16(2), 145–158 (Jun 2009), publisher: SAGE Publications Inc
30. Reinecke, L., Klimmt, C., Meier, A., Reich, S., Hefner, D., Knop-Huelss, K., Rieger, D., Vorderer, P.: Permanently online and permanently connected: Development and validation of the Online Vigilance Scale. *PLOS ONE* 13(10), e0205384 (Oct 2018)
31. Rudolph, C.W., Costanza, D.P., Wright, C., Zacher, H.: Cross-Temporal Meta-Analysis: A Conceptual and Empirical Critique. *Journal of Business and Psychology* 35(6), 733–750 (Dec 2020)
32. Schneider, F.M., Hitzfeld, S.: I Ought to Put Down That Phone but I Phub Nevertheless: Examining the Predictors of Phubbing Behavior. *Social Science Computer Review* (Oct 2019), publisher: SAGE PublicationsSage CA: Los Angeles, CA
33. Seo, S.: A Review and Comparison of Methods for Detecting Outliers in Univariate Data Sets. Master’s thesis, University of Pittsburgh (2002)
34. Smith, A.: *Smartphone Ownership — 2013 Update*, vol. 12. Pew Research Centre, Washington D.C. (2013)
35. Smits, I.A.M., Dolan, C.V., Vorst, H.C.M., Wicherts, J.M., Timmerman, M.E.: Cohort differences in Big Five personality factors over a period of 25 years. *Journal of Personality and Social Psychology* 100(6), 1124–1138 (Jun 2011)
36. Torres, K.M.: *Disrupting the College Classroom Experience: Avoiding Technology Pitfalls* (2022), ISBN: 9781799892434 Pages: 223–242 Publisher: IGI Global
37. Twenge, J.M.: The age of anxiety? The birth cohort change in anxiety and neuroticism, 1952–1993. *Journal of Personality and Social Psychology* 79(6), 1007 (2000), publisher: US: American Psychological Association
38. Uncapher, M., Wagner, A.: Minds and brains of media multitaskers: Current findings and future directions. *Proceedings of the National Academy of Sciences of the United States of America* 115(40), 9889–9896 (2018)

39. Venables, W.N., Ripley, B.D.: Modern Applied Statistics with S. Springer, New York, fourth edn. (2002)
40. Wallace, J.C., Kass, S.J., Stanny, C.J.: The Cognitive Failures Questionnaire Revisited: Dimensions and Correlates. *The Journal of General Psychology* 129(3), 238–256 (Jul 2002), publisher: Routledge
41. Wammes, J.D., Ralph, B.C., Mills, C., Bosch, N., Duncan, T.L., Smilek, D.: Disengagement during lectures: Media multitasking and mind wandering in university classrooms. *Computers & Education* 132, 76–89 (Apr 2019)
42. Wiradhany, W., Nieuwenstein, M.R.: Cognitive control in media multitaskers: Two replication studies and a meta-Analysis. *Attention, Perception, & Psychophysics* 79(8), 2620–2641 (Nov 2017)
43. World Bank: World Bank Development Indicators (2021), <https://databank.worldbank.org>[Accessed: 12 May 2022]

CS/IS/IT Programme Accreditation and SACAB support at Higher Education Institutions in Southern Africa

Estelle Taylor¹[0000-0003-2848-7829], Andre P. Calitz²[0000-0002-2555-9041], and
Sue Petratos³[0000-0002-2934-5198]

¹School of Computer Science and Information Systems, North-West University, Potchefstroom, S.A.

²Department of Computing Sciences, Nelson Mandela University, Port Elizabeth, S.A.

³School of IT, Nelson Mandela University, Port Elizabeth, S.A.

¹Estelle.Taylor@nwu.ac.za; ²Andre.Calitz@Mandela.ac.za;
³Sue.Petratos@Mandela.ac.za

Abstract. The accreditation of degree programmes by professional and statutory bodies ensures that education institutions adhere to set standards. In South Africa, only a small number of Computer Science (CS), Information Systems (IS), Information Technology (IT) and other related computing degree programmes are accredited by international accreditation bodies. The British Computer Society (BCS) accredits degree programmes in the United Kingdom and internationally. The Australian Computer Society and the IT Professionals New Zealand perform similar accreditation in their countries. In South Africa, the South African Computing Accreditation Board (SACAB) was established under the control of the Institution of IT Professionals of South Africa (IITPSA). In addition, a South African IT Diploma Advisory Board (SAITDAB) was proposed that would operate in collaboration with the SACAB.

The aim of the study was to investigate the accreditation of CS/IS/IT degree programmes at universities in Southern Africa by international bodies and whether universities would support the SACAB degree and diploma programme accreditation in South Africa. An accreditation questionnaire was compiled and distributed to the Head of Departments (HODs) of 32 universities in Southern Africa. A total of 27 HODs or representatives at 14 universities completed the survey over a two-week period. The data were statistically analysed and the results of the study indicate that three of the 14 institutions had degree programmes accredited by the BCS or the Accreditation Board for Engineering and Technology (ABET). The majority of the respondents would support the SACAB accreditation initiative in order to maintain high academic standards. This research study could assist CS/IS/IT academic departments to prepare for programme accreditation. The study also reports on the importance of different components during the accreditation process and site visit and specific graduate abilities.

Keywords: Accreditation, SACAB, Academic standards.

1 Introduction

Academic departments in Computer Science (CS), Information Systems (IS), Information Technology (IT) and other related departments at Higher Education Institutions (HEIs) should offer programmes and curricula that are relevant in a fast-changing environment, as students should be prepared for a constantly changing workplace. Organisations, for example the IEEE Computer Society and the ACM, have specified requirements and content for programmes, such as CS (CS2023) [1], IS (IS2020) [2] and IT (IT2017) [3].

Computing accreditation boards are responsible for evaluating and accrediting academic programmes in computing and related fields to ensure they meet certain academic quality standards. The main purpose of accreditation is to provide assurance to students, employers and other stakeholders that an academic programme at an institution meets specified minimum standards in terms of curriculum content, faculty qualifications, infrastructure and other criteria. Accreditation can also help ensure that graduates of accredited programmes have the knowledge and skills needed to be successful in industry.

In the United States of America, the two primary computing accreditation boards are the Accreditation Board for Engineering and Technology (ABET) [4] and the Computing Accreditation Commission (CAC) [5] of the Accreditation Board for Engineering and Technology. In South Africa, the South African Computer Accreditation Board (SACAB) [6] was established under the administration of the Institute of IT Professionals of South African (IITPSA) [7], with the South African Computing Lecturers Association (SACLA) and the South African Institute of Computer Scientist and Information Technologists (SAICSIT) [9] representatives serving on the board. In addition, the South African IT Diploma Advisory Board (SAITDAB) was proposed in 2020 at the SACLA conference and is associated with the SACAB [10].

This paper provides the current status of computing programme accreditation at different universities in South Africa and the support for the SACAB going forward. The study also reports on the different components during the accreditation process and site visit and specific graduate abilities respondents rated as important. The research problem, research questions and the accreditation survey are discussed in Section 2. Literature on accreditation are discussed in Section 3. The accreditation survey results are presented in Section 4. Conclusions and recommendations of this study, relevant to Departments making use accreditation bodies at HEIs and future work are discussed in Section 5.

2 The Research Problem and Research Methodology

Accreditation has increasingly become a valuable tool for quality assurance in Higher Education, both to improve processes and to use as a marketing tool to attract students and funding [11]. Topi [12] highlights accreditation as a mechanism for the improvement of quality in information systems. There is pressure on institutions to continuously improve academic programmes [13]. According to literature sources, accreditation

serves as an excellent tool for programmes to continuously improve and enhance its visibility and reputation [12, 14]. Although benefits from accreditation are clear, there are also some reservations. Preparing for an accreditation review, managing the process, requires planning, coordination, implementation, monitoring, feedback and evaluation seems like a daunting task [15]. The accreditation review process is an intensive team effort [16], requiring multiple measurement cycles and approaches [11].

Some research reports scepticism about the effectiveness and fairness of some of the accreditation criteria [14]. Andreani *et al.* [17] refer to an ongoing debate on whether accreditation systems really improve the quality of higher education. Previous studies report a scarcity of available literature on the implementation of accreditation approaches for computing programmes [14]. Another issue, mentioned by Rahmed *et al.* [16], is limited studies focused on specific countries to highlight their specific issues and problems, and finding solutions for the requirement of these countries according to their own education policies and procedures. This is especially important for this study, as little research has been done on accreditation, specifically in the South African context. Prospective national and international students wanting to study in South Africa are enquiring about degree programme accreditation to provide articulation of qualifications internationally [10]. In South Africa, computing degree programmes are generally not accredited with international accreditation authorities and currently governmental regulations do not stipulate accreditation requirements.

The research problem investigated in this study is based on the realisation that only a limited number of CS/IS/IT degree programmes in South Africa are accredited. The research questions addressed in this study were:

- *Which CS/IS/IT degree programmes at HEIs in Southern Africa are accredited by international accreditation bodies?*
- *Would HEIs support the SACAB in accrediting CS/IS/IT degree and diploma programmes at HEIs in Southern Africa?*
- *How important are the different components during the accreditation process and site visit?*
- *What are the levels of agreement regarding the importance of graduate abilities that must be considered during accreditation?*

In order to determine departmental accreditation practices and perceptions, a questionnaire was compiled that included the following sections:

- Biographical details;
- Open-ended questions relating to the accreditation of degree programmes; and
- Likert scale statements relating to the accreditation process and specific graduate abilities (as identified during the literature review).

The questionnaire was captured using the online survey tool, QuestionPro. The Heads of Departments (HODs) from all 32 universities in Southern Africa were requested to complete the survey. The first call for participation was distributed via e-mail to the SACLA HOD list and later followed by a call on the SACLA Google mailing list.

A total of 27 HODs or representatives completed the survey over a two-week period. The data were statistically analysed and the qualitative results were thematically analysed.

3 Literature Review

Advantages of accreditation will be described, followed by the characteristics an accreditation system should have, as well as a discussion of the accreditation process.

3.1 Accreditation of programmes by accreditation bodies

Accreditation bodies are responsible for standard and procedures, giving assurance that a programme has been determined of high quality. The aim of accreditation is to ensure that graduates have the necessary skills to contribute to a profession [4]. Graduates from accredited programmes have the assurance that the community will recognise their achievement. Accreditation of programmes in computing is conducted by different organisations, ranging from government bodies to private organisations. In some areas, accreditation is regulated by laws, while in other areas it is a voluntary process [8].

The advantages of accreditation were summarised from several sources (ABET [4]; Accreditation.org [8]; ACS [18]; ASIIN [19], BCS [20]; Blair *et al.* [21]; Calitz, *et al.* [22], CIPS [23]; IPTNZ [24]; Mohiuddin *et al.* [25]; Oudshoorn *et al.* [26]; Rahmed *et al.* [16]; Topi [12]) and include:

- Promoting quality in higher education by setting standards for graduates;
- Giving recognition to quality programmes;
- Providing independent recognition for institutions;
- Contributing to meeting future industry needs;
- Giving graduates a stamp of approval, producing graduates who have the knowledge and skills required of an ICT professional;
- Providing opportunity for self-reflection and supporting improvement of programmes;
- Protecting the interest of students, the academic institutions, potential employers, other stakeholders and the public;
- Confirming that a course of study meets the high-level requirements of science and professional practice of the involved disciplines, and that programmes are industry-aligned and of value to the industry;
- Documenting that conditions for good teaching and successful learning are in place and highlighting areas of best practice across institutions;
- Benchmarking programmes against those of other institutions both nationally and internationally;

- Assist with attracting top students, as students are increasingly searching for degrees that have been accredited by a professional body; and
- Promoting public welfare through the development of quality computer professionals and fostering a cooperative approach between industry, government, and educators to meet the changing needs of society.

3.2 Characteristics of accreditation processes

The characteristics of different accreditation processes were compiled from several sources [4, 8, 19, 27] and this section provides a summary of some of these characteristics. Participation in the accreditation process should be voluntary. The process is not controlled by governmental bodies (government inspection is not accreditation), but preferably by local educators and practitioners. Programmes have their choice of an accrediting body (there is more than one accreditation agency that a programme may use). The process is more evaluative than regulatory.

Information of accredited programmes must be provided, as well as when it was accredited, which accrediting body issued the accreditation and what the expiration date is of the current accreditation credential. The process must include a comprehensive self-evaluation and standards used in the evaluation must be clear. The quality is also determined by the success that graduates achieve in their profession.

In cases where the programme is offered in several campus or modalities, it will be accredited only if all the campuses where the programme is offered are evaluated and satisfy the accreditation criteria and policies. In case the evaluation of a single programme is required and it is offered in another campus, the academic degree must identify the campus.

3.3 International CS/IS/IT programme accreditation

Information on the major accreditation bodies for computing are summarised in Table 1. The accreditation bodies included in this list were identified using the comprehensive list available at <https://accreditation.org/accreditation-processes/accrediting-bodies>, excluding only those not in the field of CS, IS or IT. The further information in Table 1 were compiled from ABET [4], Accreditation.org [8], ASIIN [19], BCS [20], CIPS [23], CONAIC [28], ICADIT [27], ITPNZ [24], MBOT [30] and PICAB [31]. The countries the accreditation bodies have conducted accreditation reviews in are specified, as well as the fields of accreditation.

3.4 Seoul Accord participants documentation

The Seoul Accord [32] is a mechanism for mutual recognition of graduates of accredited programmes among the signatories of the Accord. It is based on the principle of equivalence of educational preparation for entry to a computing profession, rather than on exact correspondence of content and outcomes of accredited programmes.

The Seoul Accord is an international agreement among organisations responsible for accrediting computing and related programmes [32]. It was signed in 2008 in Seoul,

South Korea, and its purpose is to establish mutual recognition of accredited computing programmes among signatory countries. The Accord is based on the idea that computing is a global profession and that there should be a common set of standards for accreditation that can be recognised around the world. The Accord specifies consensus on the generally accepted attributes of graduates [32].

Table 1. Accreditation Bodies for Computing

Organisation	Accrediting in	Fields
ABET	United States, Austria, Bahrain, Brazil, Brunei, Chile, China, Colombia, Ecuador, Egypt, India, Indonesia, Jamaica, Jordan, Kazakhstan, Korea, Kuwait, Lebanon, Mexico, Morocco, Oman, Palestine, Peru, Philippines, Poland, Portugal, Qatar, Russia, Saudi Arabia, South Africa, Spain, Turkey, UAE, Vietnam	Non-profit, non-governmental agency that accredits programs in applied natural science, computing, engineering, and engineering technology.
ASIIN	Armenia, Australia, Azerbaijan, Bosnia and Herzegovina, China, Croatia, Cyprus, Finland, Indonesia, Germany, Kazakhstan, Mongolia, Namibia, Peru, Serbia, Slovenia, Switzerland, Tunisia, Turkey	Accrediting degree programs from the fields of engineering, informatics/computer science, the natural sciences and mathematics at bachelor, Master and PhD levels.
Australian Computer Society (ACS)	Australia, Fiji, Oman, Singapore, Sri Lanka, UAE, Vietnam	Provide accreditation for Computer Science, Information systems, Computer Systems Engineering, and Software Engineering.
British Computer Society (BCS)	UK and International	Recognition of accredited academic computing programmes that prepare graduates for professional practice.
Canada's Association of Information Technology Professionals (CIPS)	Canada, United Arab Emirates	Provides accreditation of Computer Science Degree Programs and Software Engineering Degree Programs
Institute of IT Professionals New Zealand (ITPNZ)	New Zealand	Operates the IT industry's independent degree accreditation process, recognising and endorsing degree programmes and supporting international portability of qualifications.
Instituto de Calidad y Acreditación de Programas de Computación, Ingeniería y Tecnología (ICACIT)	Peru	An international accrediting agency specialized in undergraduate and postgraduate Engineering, Computer Science, and Computational Engineering.
Malaysia Board of Technologists (MBOT)	Malaysia	Gives Professional Recognition to Technologists and Technicians in related technology and technical fields.
Mexico National Council for Accreditation in Informatics and Computing (CONAIC)	Mexico	Accreditation is conducted in the following areas: Computer Engineering, Software Engineering, Computer Science, and Computational Engineering.
Philippine Information and Computing Accreditation Board (PICAB)	Philippines	Responsible for accreditation of computing programs in the Philippines using an outcomes-based system for computing and information technology education at the bachelor degree level.

3.5 SACAB

In South Africa, all qualifications offered by Higher Education Institutions must be approved by the Department of Education (DOE), the South African Qualifications Authority (SAQA) and the Higher Education Quality Committee (HEQC) before the academic institution can offer the qualification. All programmes must be Higher Education Qualifications Framework (HEQF) compliant. The purpose of the Higher Education Qualification Sub Framework (HEQSF) [33] is to define the relationships between qualification types.

The SACAB maintains the accreditation standards for CS/IS/IT degree programmes at HEIs in SA and IT diploma programmes offered by Comprehensive Universities and Universities of Technology in South Africa [10]. Figure 1 shows a visual representation of the SACAB structure, linking the SACAB with the Seoul Accord and four South African stakeholders. The IITPSA maintains the link to the Seoul Accord. An administrative department at the IITPSA office in Midrand SA, manages the daily operations of the SACAB and the various assessment panels responsible for the accreditation visits and reviews.

The SACAB composition consists of two SAICSIT members, the SAICSIT Chairperson and SAICSIT nominated representative, three IITPSA representatives, two SACLA representatives, two HOD's representatives nominated by HODs and the SACAB Chairman.

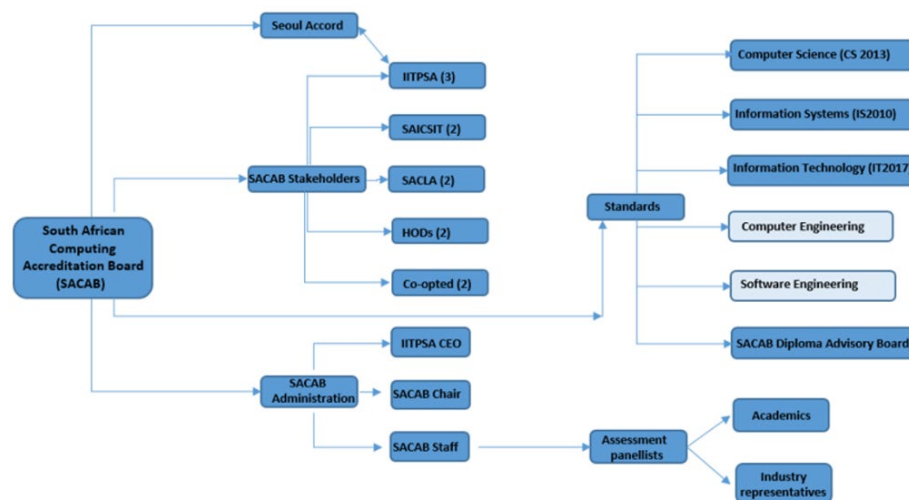


Fig. 1. SACAB Structure

3.6 SACAB documentation set

The SACAB documentation set provides guidelines for universities seeking SACAB accreditation. The SACAB Degree and Diploma Accreditation Document Set consists of the following nine documents:

1. SACAB General Information (Document 1);
2. SACAB Administrative Guidelines (Document 2);
3. SACAB Application Guidelines (Document 3);
4. SACAB Guidelines for Submissions (Document 4);
5. SACAB Submission Forms (Document 5);
6. SACAB Graduate Attributes (Document 6)
7. SACAB CS/IS/IT Degree Programme Body of Knowledge (Document 7);
8. SACAB IT Diploma Programme Body of Knowledge (Document 8); and
9. SACAB/IITPSA Professional Body of Knowledge (Document 9).

3.7 The South African IT Diploma Advisory Board (SAITDAB)

The SAITDAB was proposed in 2020 at the SACLA conference and is associated with the SACAB [10]. A proposed rationale and process for drafting an IT Diplomas' and Advanced Diplomas' programme Body of Knowledge (BOK) was recommended. The establishment of the SAITDAB was fully supported by representatives from Comprehensive Universities and Universities of Technology [10]. It was concluded that the SAITDAB must further establish administrative guidelines for IT diploma programmes and draft a core BOK for the different IT diploma programmes. Evaluation criteria, a standards document for accreditation purposes must be compiled and published by the SACAB, based on the work done by the SAITDAB.

3.8 The accreditation process and specific graduate abilities

The most important components to include during the self-study and review, compiled from several sources (ABET [4]; Accreditation.org [8]; ACS [18]; ASIIN [19]; BCS [20]; ICACIT [27]; PICAB [31]; Rahmed *et al.* [16]):

1. Developing a mission and vision statement;
2. Setting or re-articulating the program's goals and standards;
3. Developing and maintaining accreditation standards and criteria;
4. Describing specific and measurable objectives for teaching and learning;
5. Describing strategies and processes in place for quality control;
6. Conducting a comprehensive data collection and data analysis;
7. Demonstrate a continuous improvement loop;
8. Preparing examples.

Compiled from the sources mentioned above, examples may include a curricular overview for each programme, with objectives and learning outcomes of degree programmes, and outcomes of modules; explanation and proof of the teaching and learning

process and assessment strategies, including samples of student work, syllabi, textbooks and sample assignments, teaching methodology, assessments and procedures surrounding assessments, support available to students; the human resources used to support the programme, for example staff, staff development and leadership; physical and material resources, for example classrooms, offices, laboratories, libraries and associated equipment; institutional support and leadership from top management must be adequate to ensure the quality and continuity of the programme; external engagements, for example information about industry involvement in programme design and industry advisory boards; funds and equipment; the HEI's internal regulations regarding academic progression and awarding of degrees; interviews with students, faculty, staff, administrators and review of display materials; alumni records; and Faculty / School / department workload.

The British Computer Society provides specific abilities that graduates must be assessed on [20]. This can be valuable in the evaluation of programmes:

- *“Essential facts, concepts, principles, and theories relating to computing and computer applications as appropriate to the programme of study, and the use of such knowledge and understanding in the modelling and design of computer-based systems.*
- *Management techniques which may be used to achieve objectives within a computing context.*
- *Information security issues in relation to the design, development, and the use of information systems.*
- *Methods, techniques and tools for information modelling, management, and security.*
- *Systems architecture and related technologies for developing information systems.*
- *Mathematical and/or statistical principles appropriate to the nature of the programme.*
- *Graduates must have the ability to:*
 - *Recognise and analyse criteria and specifications appropriate to specific problems, and plan strategies for their solution;*
 - *Analyse the extent to which a computer-based system meets the criteria defined for its current use and future development;*
 - *Deploy appropriate theory, practices and tools for the specification, design, implementation, and evaluation of computer-based systems; and*
 - *Recognise the legal, social, ethical, and professional issues involved in the exploitation of computer technology and be guided by the adoption of appropriate professional, ethical and legal practices.”*

ICACIT [27] also mention specific requirements for accreditation in computing sciences:

- *“Computer Knowledge: Apply knowledge of mathematics, science and computing appropriate to the solution of defined problems and their requirements in the discipline of the programme.*
- *Problem Analysis: Identify, formulate, research literature and solve complex computing problems and other relevant disciplines in the domain.*
- *Design and Development of Solutions: Design, implement and evaluate solutions to complex computing problems and design and evaluate systems, components or processes that satisfy specific needs.*
- *Use of Modern Tools: Create, select, adapt and apply modern techniques, resources and tools for the practice of computing and understand their limitations.*
- *Individual and Teamwork: Function effectively as an individual, as a member or leader of diverse teams.*
- *Communication: Communicate effectively orally and in writing, in a variety of professional contexts.*
- *Professionalism and Society: Analyse and assess the local and global impact of computing on people, organizations and society.*
- *Ethics: Understand and commit to ethics, professional responsibilities, and standards of professional computing practice.*
- *Continuous Learning: Recognize the need and can engage in autonomous learning for continuous professional development.”*

PICAB [31] refer to the same requirements mentioned by ICACIT [27], but adds some specific Computer Science requirements, namely coverage of fundamentals of algorithms and data structures; A variety of programming languages and systems; At least one high level language; Advanced course work that provides depth by building on fundamental course work; and Mathematics (including discrete mathematics such as calculus, linear algebra, numerical methods, probability, statistics, number theory, geometry or symbolic logic).

These issues were further investigated by adding specific questions in the *Accreditation survey 2023*.

4 Accreditation Survey Results

The *Accreditation survey 2023* was initially completed by 11 HOD's and 8 HOD representatives from 19 academic departments at 13 universities in South Africa and one from Eswatini (Fig. 2). The respondents included departments of Computer Science ($n=4$), CS&IS ($n=3$), Information Systems / Informatics ($n=4$) and other related IT departments. The survey link was emailed to HODs on the SACLA HOD list. The survey link was later also distributed on the SACLA Google mailing list, requesting HODs or departmental representatives to complete the survey. An additional 8 responses were received, however none from additional universities already included in the study.

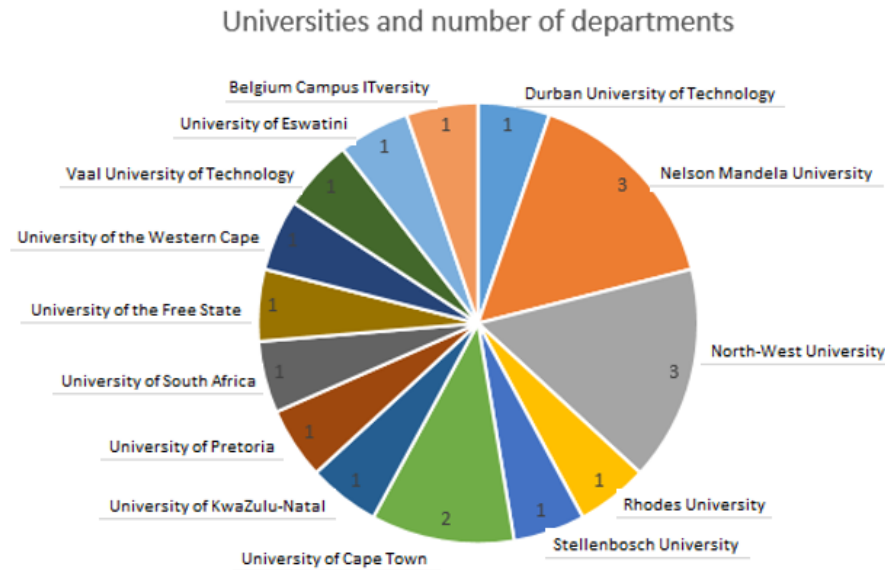


Fig. 2. Southern African universities and academic departments.

Ninety-two percent ($n=25$) of the respondents indicated that the accreditation of academic programmes in South Africa is necessary. Nine respondents indicated that their academic programmes were accredited, which were programmes at three universities, including programmes accredited by the British Computer Society. Twenty-five of the respondents indicated they would support the SACAB to accredit degree and/or diploma programmes at their institution and 26 indicated that they would support having staff members from their institution serve on the SACAB.

The respondents were asked why their programmes were not accredited (Fig. 3a). The main reasons included costs ($n=5$), time ($n=4$) and personnel requirements. Two respondents indicated costs and personnel shortages. P7 said “Its additional work to get through that academic staff do not have the time for” and P4 “Never had the time, effort and money for external accreditation”. P21 indicated “University departments do not have enough human resources to handle such accreditation processes. Overall for public universities funds are limited” and P24 said “No accreditation board exists”.

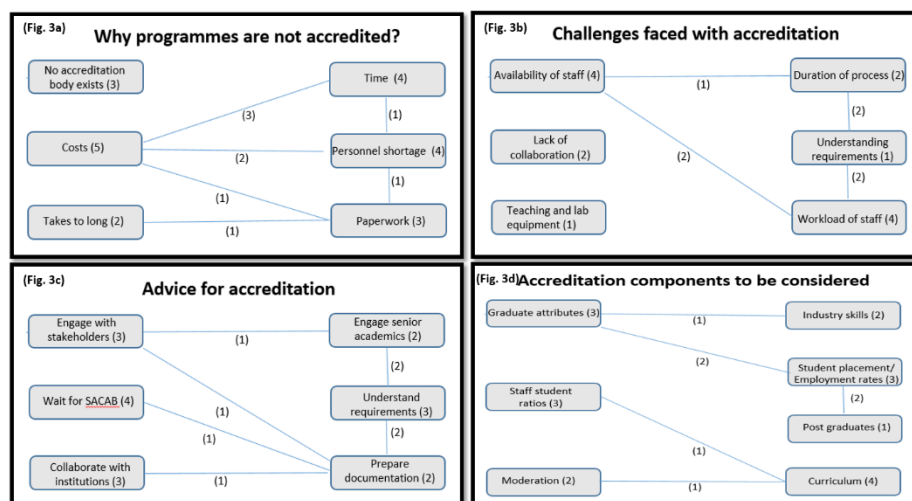


Fig. 3. Accreditation feedback

The respondents were asked what are the challenges they may face having programmes accredited (Fig. 3b). The availability of staff ($n=4$) and staff workload ($n=4$) were the main challenges, with two respondents indicating staff availability and present staff workloads. P4 indicated “Rapidly changing technology and a lack of collaboration between universities and accreditation bodies are resulting in a lack of understanding of the accreditation process and requirements”. P6 said “The duration of the accreditation process” and P5 “Current staff teaching workload is stretched to capacity” and “shortage of academic staff and office spaces within the school” are challenges.

The advice they would give to other institutions wanting to have their programmes accredited, included engaging with stakeholders ($n=3$), understand requirements ($n=3$) and collaborate with other institutions (Fig. 3c). P2 suggested “Engage with relevant stakeholders early on ensure that all business processes and historical documentations relative to courses etc. are up to date and regularly do a reflection on teaching practices and curriculum to remain relevant”.

The other components of accreditation (Fig. 3d) that needed to be considered were the ACM curriculum ($n=4$), staff / student ratios ($n=3$), graduate attributes ($n=3$) and graduate employment ($n=3$). P12 said “The content of the curriculum must be in line with international specifications” and P5 said “Employment rates, articulation rates to postgraduate study’ were important.

Advantages of programme accreditation

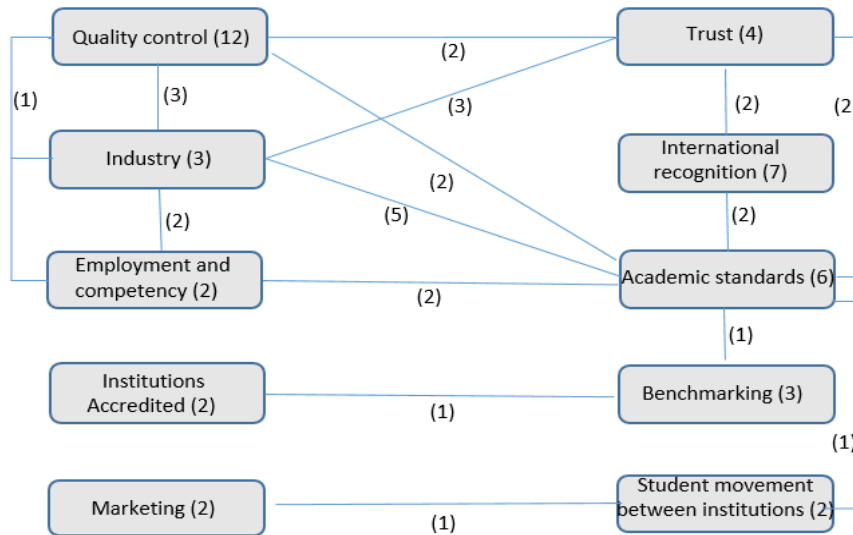


Fig. 4. Advantages of accreditation

The thematic analysis of the advantages of accreditation as indicated by the respondents are presented in Fig. 4. Quality control ($n=12$) was the most important advantage, followed by international recognition ($n=7$), maintaining academic standards ($n=6$) and trust ($n=5$) in the quality of the qualification. Five respondents indicated academic standards and industry. Participant P12 said “Maintaining standards, having a benchmark in which students and industry can have confidence”.

Participant P6 indicated “Self-evaluation and associated improvements; proofing quality to students and industry; attracting students; if associated with e.g. Seoul Accord it will simplify movement internationally for students”. P3 said “National standards allow students to move between institutions. Employees are guaranteed on academic standards” and P13 “Accreditation help to maintain standards and helps argue for resource at university”. P26 indicated “It should enhance the status of accredited programs by association with other accredited programmes, presumably impacting on the employability of graduates”.

The analysis of the responses (Table 2) regarding the importance of the components during the accreditation process and site visit indicated that all 18 listed components were seen as *important to very important*. Eleven respondents indicated that *Alumni records* were of little importance. *Programme's goals and standards*, *Curricular overview for the programme, with specific and measurable objectives and learning outcomes of programme* and *Outcomes of all modules in the programme* were seen as very important. Staff workloads were again highlighted as a concern.

Table 2. Importance of the components of the accreditation process (n=27)

Statement	Not important	Of little importance	Important	Very important
Mission and vision statement.	2	5	16	4
Program's goals and standards.	0	0	13	14
Curricular overview for the program, with specific and measurable objectives and learning outcomes of programme.	0	0	9	18
Outcomes of all modules in the programme.	0	0	9	18
Strategies and processes in place for quality control.	0	0	13	14
Teaching and learning process	0	1	16	10
Assessment strategies and procedures surrounding assessments.	0	0	12	15
Samples of student work.	0	5	16	6
Syllabi and textbooks.	0	2	17	8
Support available to students.	0	4	14	9
The human resources used to support the programme, for example, staff, staff development, and leadership.	0	0	17	10
Physical and material resources, for example, classrooms, offices, laboratories, libraries, and associated equipment.	1	1	15	10
Institutional support and leadership from top management.	1	3	12	11
External engagements, for example information about industry advisory boards.	0	5	16	6
Funds and equipment.	0	2	15	10
The HEI's internal regulations regarding academic progression and awarding of degrees.	0	6	11	10
Alumni records.	3	11	11	2
Faculty / School department workload.	1	2	9	15

Regarding the specific abilities graduates must be assessed on (Table 3), the respondents regarded the 12 statements all to be important. The *Design and Development of Solutions* was rated the most important statements because *Design, implement and evaluate solutions to complex computing problems and design and evaluate systems, components or processes that satisfy specific needs*.

Table 3. Specific abilities of graduates (n=27)

Statement	Not important	Of little importance	Important	Very important
Management techniques which may be used to achieve objectives within a computing context.	0	4	20	3
Recognise the legal, social, ethical, and professional issues involved in the exploitation of computer technology and be guided by the adoption of appropriate professional, ethical and legal practices.	0	4	17	6
Compute knowledge: Apply knowledge of mathematics, science and computing appropriate to the solution of defined problems and their requirements in the discipline of the program.	0	1	11	15
Problem analysis: Identify, formulate, research literature and solve complex computing problems and other relevant disciplines in the domain.	0	0	4	23
Design and development of solutions: Design, implement and evaluate solutions to complex computing problems and design and evaluate systems, components, or processes that satisfy specific needs.	0	0	3	24
Use of modern tools: Create, select, adapt and apply modern techniques, resources, and tools for the practice of computing and understand their limitations.	0	0	9	18
Individual and team work: Function effectively as an individual, as a member or leader of diverse teams.	0	1	15	11
Communication: Communicate effectively orally and in writing, in a variety of professional contexts.	0	0	17	10
Continuous learning: Recognize the need and can engage in autonomous learning for continuous professional development.	0	1	17	9
Coverage of fundamentals of algorithms and data structure.	0	2	13	12
Variety of programming languages and systems.	0	6	14	7
At least one high level language.	0	1	10	16

5 Conclusions, Limitations and Future Research

The literature study indicated that there are differences in opinions on issues surrounding accreditation, such as the objectives, membership, benefits, size, leadership, meetings and guidelines [34]. Accreditation of computing programmes is conducted by different organisations, ranging from government bodies to private associations, as listed in Table 1. In South Africa, the SACAB has been established and a set of nine accreditation documents compiled.

The findings of the *Accreditation survey 2023* indicate overwhelming support for the SACAB by respondents representing 14 universities. The respondents indicated that their institutions will support staff members to serve on the SACAB. Only three universities in Southern Africa had computing related programmes accredited by organisations, such as the British Computer Society. The main reasons for not accrediting programmes were costs, time and personnel requirements. The availability of staff ($n=4$) and staff workload ($n=4$) were the main challenges faced by institutions.

The advice respondents provided to institutions wanting to accredit their programmes included engaging with stakeholders early, understanding the accreditation requirements and collaborate with other institutions that have completed the accreditation process. Learning best practices from institutions that have successfully completed the accreditation process is important. All 18 listed components listed in Table 2 were rated as important to very important during the accreditation process and site visit. The respondents rated the 12 statements regarding the specific abilities graduates must be assessed on in Table 3 all to be important.

The Seoul Accord [32] is an international agreement amongst organisations responsible for accrediting computing and related programmes. Accreditation bodies that sign the Seoul Accord are expected to follow certain criteria and guidelines in assessing and accrediting computing programmes. These criteria include factors such as curriculum, faculty qualifications, student outcomes, and facilities and resources. The IITPSA and specifically the SACAB will have to become a signatory of the Seoul Accord.

The limitations of this study were that not all universities offering computing related degree and diploma programmes in South Africa responded to the two calls for participation in the survey. Further research will be required for establishing the South African IT Diploma Advisory Board (SAITDAB) and related documentation.

References

1. CS2023: CS2023 ACM/IEEE-CS/AAAI Computer Science Curricula (2023), <https://csed.acm.org>, last accessed 2023/04/17.
2. IS2020: IS2020 ACM-AIS Curriculum (2023), <https://is2020.hosting2.acm.org>, last accessed 2023/04/17.
3. IT2017: Information Technology Curricular Guidelines (2023,) <https://it2017.acm.org>, last accessed 2023/04/17.
4. ABET: What is accreditation? (2023), <https://www.abet.org/accreditation>, last accessed 2023/04/17.
5. CAC: Computing Accreditation Commission (CAC), (2023), <https://www.abet.org/accreditation/get-accredited/accreditation-step-by-step/self-study-report/dual-templates-for-cac>, last accessed 2023/04/17.
6. SACAB: South African Accreditation Board (2023), <https://www.sacab.org.za>, last accessed 2023/04/17.
7. IITPSA: Institute of IT Professionals of South Africa (2023), www.iitpsa.org.za, last accessed 2023/04/17.
8. Accreditation.org: Accrediting bodies (2023), <https://accreditation.org/accreditation-processes/accrediting-bodies>, last accessed 2023/04/17.
9. SAICSIT: South African Institute of Computer Scientists & Information Technologists (2023), www.SAICSIT.org.za, last accessed 2023/04/17.
10. Calitz, A. P., Cowley, L. Petratos, S.: A Proposed Structure for Managing IT Diploma Qualifications in South Africa. In ICT Education: 49th Annual Conference of the Southern African Computer Lecturers' Association, SACLA 2020, Virtual Event, July 6–9 (2020).
11. Saeed, S., Almuhaideb, A.M., Bamarouf, Y.A., Alabaad, D.A., Gull, H., Saqib, M., Iqbal, S.Z., Salam, A.A.: Sustainable Program Assessment Practices: A Review of the ABET and NCAAA Computer Information Systems Accreditation Process. *International Journal of Environmental Research and Public Health*, 18, 12691 (2021).
12. Topi, H.: Reflections on the current state and future of information systems education. *Journal of Information Systems Education*, 30(1) (2019).
13. Almuhaideb, A. M., Saeed, S.: Fostering sustainable quality assurance practices in outcome-based education: lessons learned from ABET accreditation process of computing programs. *Sustainability*, 12(20), 8380 (2020).
14. Shafi, A., Saeed, S., Bamarouf, Y. A., Iqbal, S. Z., Min-Allah, N., Alqahtani, M. A.: Student outcomes assessment methodology for ABET accreditation: A case study of computer science and computer information systems programs. *IEEE Access*, 7, 13653-13667 (2019).
15. James-Okeke, P. A., Ladeji-Osias, J. O., Scott, C. J.: ABET Accreditation: Best Practices for A Systematic Coordinated Multi-Program Approach. In 2019 ASEE Annual Conference & Exposition (2019).
16. Rahman, A. U., Alfawarch, H. M., Altowaijri, S. M.: Developing and qualifying computing programs for ABET accreditation: a practical approach. *International Journal of Computer Science and Network Security*, 19, 29-45 (2019).
17. Andreani, M., Russo, D., Salini, S., Turri, M.: Shadows over accreditation in higher education: Some quantitative evidence. *Higher Education*, 79, 691-709 (2020).
18. ACS: Australian Computer Society (2023), <https://www.acs.org.au/>, last accessed 2023/04/17.
19. ASIIN: Global Leader in Quality Assurance in Higher Education (2023) <https://www.asiin.de/en/>, last accessed 2023/04/17.

20. BCS: British Computer Society. Academic accreditation (2023), <https://www.bcs.org/deliver-and-teach-qualifications/academic-accreditation>, last accessed 2023/04/17.
21. Blair, J. R., Jones, L., Leidig, P., Murray, S., Raj, R. K., Romanowski, C. J.: Establishing ABET accreditation criteria for data science. In *Proceedings of the 52nd ACM Technical Symposium on Computer Science Education*, 535-540 (2021).
22. Calitz, A. P., Greyling, J. H., Cullen, M. D.: The accreditation of ICT degree programs in South Africa. In *Proceedings of the South African Institute of Computer Scientists and Information Technologists Conference on Knowledge, Innovation and Leadership in a Diverse, Multidisciplinary Environment*, 49-58 (2011).
23. CIP: Canada's Association of Information Technology Professionals. Accredited programmes (2023), <https://cips.ca/accreditation>, last accessed 2023/04/17.
24. ITPNZ: Institute of IT Professionals New Zealand (2023), <https://itp.nz>, last accessed 2023/04/17.
25. Mohiuddin, K., Islam, A., Mohd, S., Shariff, M.: Evaluation of an Academic Program: The Case of Computing Accreditation Commission Framework in Higher Education. *International Journal of Emerging Technologies in Learning (Online)*, 14(11), 70 (2019).
26. Oudshoorn, M., Raj, R.K., Thomas, S., Parrish, A.: The value of abet accreditation to computing programs. In *Proceedings of the 2018 ASEE Annual Conference & Exposition*, Salt Lake City, UT, USA, 25 June (2018).
27. ICACIT: Instituto de Calidad y Acreditación de Programas de Computación, Ingeniería y Tecnología. What is accreditation? (2023), www.icacit.org.pe, last accessed 2023/04/17.
28. CONAIC: Council for Accreditation in Informatics and Computing (2023), <https://www.conaic.net/ingles/acreditados.ht>, last accessed 2023/04/17.
29. Carelli, J.: Seeking ABET Accreditation: A Case Study in Outcome Assessment. In *Proceedings of the International Conference on Frontiers in Education: Computer Science and Computer Engineering (FECS)* (pp. 81-87). The Steering Committee of the World Congress in Computer Science, Computer Engineering and Applied Computing (WorldComp) (2019)
30. MBOT: MBOT accreditation (2023) <https://ttasmbot.org.my>, last accessed 2023/04/17.
31. PICAB: Philippine Information and Computing Accreditation Board. PICAB accreditation (2023), <https://philippinecomputersociety.org/picab2-2>, last accessed 2023/04/17.
32. Seoul Accord.: Graduate Attributes (2023), <https://www.seoulaccord.org/document.php?id=79>, last accessed 2023/04/17.
33. HEQSF: Higher Education Qualification Sub Framework (2023), <http://www.dhet.gov.za>, last accessed 2023/04/17.
34. Crick, T., Davenport, J. H., Hanna, P., Irons, A., Pearce, S., Prickett, T.: Repositioning BCS degree accreditation (2020), <https://www.bcs.org/articles-opinion-and-research/repositioning-bcs-degree-accreditation>, last accessed 2023/04/17.

A framework for creating virtual escape rooms to teach computational thinking

Theané Janse van Rensburg and Machdel Matthee^[0000-0002-6973-1798]

Department of Informatics, University of Pretoria,
Pretoria, South Africa

theane.vanrensburg@gmail.com, machdel.matthee@up.ac.za

Abstract. Computational Thinking is considered one of the essential skills for success in the future workplace. However, integrating computational thinking into the curriculum remains an educational challenge. Escape room games could potentially aid in the development of computational thinking skills because they immerse learners in a gamified, problem-solving scenario. This paper describes the development of a framework, using Design Based Research, to guide the implementation of a virtual escape room to teach computational thinking in higher education. The components include understanding the participants and computational thinking learning objectives, choosing the digital platform, deciding on a theme and puzzles and finally evaluating the escape room. The framework was implemented and evaluated in a first-year Information Systems programming setting. Although the findings do not show a significant increase in understanding of computational thinking, participants indicated that the experience with the escape room increased their motivation to learn more about computational thinking. It is therefore suggested that virtual escape rooms be used in addition to other learning interventions for the ultimate computational thinking learning experience.

Keywords: Computational thinking, game-based learning, virtual escape rooms.

Value of explicit instruction in teaching computer programming to post-graduate students: The Kirkpatrick Training Evaluation Model

Pakiso J. Khomokhoana ¹[0000-0003-3642-1248] and Ruth Wario ²[0000-0003-3733-3485]

Department of Computer Science and Informatics, University of the Free State,
Bloemfontein, South Africa
khomokhoanap@ufs.ac.za

Abstract. The effectiveness of ways of teaching in various disciplines including Computer Science (CS) programming continues to be a debatable issue. In this study, we followed the Kirkpatrick training evaluation model to better understand how Explicit Instruction (EI) interventions in teaching computer programming to post-graduate CS students can be evaluated. This understanding can be fundamental in informing how improvements can be made in the future teaching of computer programming to create a better sense of quality and robustness. The aim of this study was threefold. Firstly, to provide an overview of the Kirkpatrick model. Secondly, to relate the four levels of this model to the teaching of computer programming. Thirdly, to evaluate the use of EI interventions as a teaching approach through the model. This study followed an integrated methodological approach where data was collected through asking questions (individual semi-structured interviews). Thematic analysis of the collected data revealed that our participants found a lot of value from the EI interventions, and hence appreciate them. Overall, the findings of this study are consistent with those of many researchers that EI is one of the effective teaching and learning strategies, not only in other disciplines, but in computer programming as well.

Keywords: Computer programming, explicit instruction, explicit instruction interventions, teaching strategies, evaluation, computer science education, The Kirkpatrick Training Evaluation Model.

Reducing Contention in an Undergraduate Capstone Project Allocation System

Stephen Phillip Levitt^[0000-0001-6054-6134], and Kenneth John Nixon^[0000-0001-5391-8147]

University of the Witwatersrand, Johannesburg, South Africa
{stephen.levitt, ken.nixon}@wits.ac.za

Abstract. An online project bidding and allocation system used by the School of Electrical and Information Engineering, at the University of the Witwatersrand, is described and critically analysed. An important question that arose when designing the bidding system for projects was how to deal with the contention for highly popular projects. An approach, common in the literature, is to optimise the project allocation; however, the system implemented aims to incentivise students to consider less popular projects. Projects are allocated to groups using a quasi-random algorithm, as opposed to an optimal algorithm. Over the eleven year analysis period the percentage of first choices awarded has ranged from 57.5% to 80.8%, while the average number of unallocated groups per year is 8.2%. In addition to these outcomes project contention is shown to reduce during bidding window period in nine out of the eleven years of the study. This is due to novel aspects of this system including the transparency of the allocation algorithm and the fact that groups are able to see all competing bids for the projects they are bidding on and change their own bids in response. The allocation algorithm is, by necessity, suboptimal in order to achieve the goals of transparency and fairness in that the stated winning probability, or a greater probability, for a bid always holds true when a project is allocated.

Keywords: capstone project, student project selection, allocation algorithm

Factors Determining the Success of Online Learning Videos for Programming

Janet Liebenberg^[0000-0001-7021-757X] and Suné van der Linde^[0000-0002-3431-4054]

School of Computer Science and Information Systems, North-West University,
Potchefstroom, South Africa
`janet.liebenberg@nwu.ac.za`

Abstract. With the COVID-19 pandemic causing universities to close, online learning became a popular solution for educators and students. This study explored the factors that determine the success of video-based online learning for a programming course. The programming course was designed based on principles from a Problem-Solving Learning Environment (PSLE) that develops computational thinking, with video lectures forming part of the scaffolding and information processing components. To conduct the research, a mixed methods approach was used, using a survey for quantitative data collection and open-ended questions to collect qualitative data from 509 survey respondents taking a C# programming course. The researchers used the Unified Theory of Acceptance and Use of Technology (UTAUT) as a lens in order to make sense of and obtain a deeper understanding of the factors, including performance expectancy, effort expectancy, attitude towards using technology, facilitating conditions, and behavioural intention. The use of online concept videos was found to be beneficial for learning programming concepts, with participants reporting improvements in academic achievement, increased efficiency, low effort, enjoyment, and compatibility with learning styles. Key factors for success include ease of use, short duration, relevance, thoroughness, engagement, and availability of necessary resources. The study provides insights for lecturers of programming courses to create effective online learning videos.

Keywords: Online Learning Videos, Online Teaching, Online Learning, Programming Teaching.

Exploring flipped learning in an introductory programming module: a literature review

Nita Mennega^{1[0000-0003-0893-7810]} and Tendani Mawela^{2[0000-0003-0574-3340]}

^{1,2}Department of Informatics, University of Pretoria, South Africa
nita.mennega@up.ac.za, tendani.mawela@up.ac.za

Abstract. Introductory programming courses form the foundation of all programming qualifications, which are in turn essential to a well-educated information technology workforce. Sadly these courses cannot boast high retention or success rates. The inverted (or "flipped") classroom approach has been touted as a way to engage large classes of mixed-ability students. To inform the design of a first-year university introductory programming course, the authors conducted a systematic literature review on existing flipped classroom implementations. Five electronic databases were searched with the keywords "flipped learning" and "introductory programming" for peer-reviewed academic literature published in the decade between 2013 and 2022 to answer the question "How may flipped learning be used to enhance student learning in an introductory programming module?". It was found that while the flipped learning approach poses challenges to both lecturers and students, it creates an increase in engagement during class and ensures that better use is made of limited face-to-face time between instructors and students. Meaningful future research would focus on seeking objective student feedback to understand the nature of the student experience.

Keywords: Flipped Classroom, Flipped Learning, H5P, Instructional Videos, Introductory Programming.

Transitioning an Introductory Programming course into a Blended Learning format

Aslam Safla¹, Hussein Suleman² and James Gain³

^{1,2,3} Department of Computer Science, University of Cape Town, Cape Town, Republic of South Africa

aslam@cs.uct.ac.za, hussein@cs.uct.ac.za, jgain@cs.uct.ac.za

Abstract. Teaching Introductory Programming is one of the foundations of Computer Science education and is generally the first course novice students take. It is important for the teachers to motivate students who are learning to program with the help of a variety of teaching methods. Over the years teachers have discussed how to best teach introductory programming, what learning styles and tools to use, how to motivate students and what programming languages should be taught. The steady growth in students' numbers has also contributed to the challenges of teaching this course. Blended learning is one approach to address these challenges. Many studies have concluded that blended learning can be more effective than traditional teaching and can improve the students' learning experience. This paper describes the transitioning of an introductory programming course, as part of a large-enrolment first-year subject, into a blended learning format. Periodic face-to-face lessons and traditional forms of assessment were combined with an integrated learning environment for engaging with video, code and quizzes. The result was a lower staff workload and a learning experience that students were satisfied with. Notably, feedback from students shows that they adopted different tools based on their individual learning preferences.

Keywords: Introductory programming, blended learning, e-learning.

Demystifying the Impact of ChatGPT on Teaching and Learning

Tapiwa Gundu¹ [0000-0002-1333-3416] and Colin Chibaya² [0000-0001-6995-605X]

¹Nelson Mandela University, Port Elizabeth, South Africa

²Sol Plaatje University, Kimberley, South Africa

¹tapiwag@mandela.ac.za, ²colin.chibaya@spu.ac.za

Abstract. The use of chatGPT as a teaching and learning tool is, generally, seen as a binary adventure to either destroy the traditional approaches to education or to innovatively revolutionize teaching and learning. Using large language models, chatGPT can generate detailed responses to prompts and follow-up questions. In this article, the hope is to explore the benefits and risks associated with using chatGPT in teaching and learning, the opportunities it offers to students and instructors, as well as the challenges it brings about. The key paradoxes thereof are discussed before precise recommendations are separately listed for students and instructors. However, generally, chatGPT is a tool that can be used along with specific strategies for educational benefits.

Keywords: ChatGPT, Generative Artificial Intelligence.

Using ChatGPT to Encourage Critical AI Literacy Skills and for Assessment in Higher Education

Cheng-Wen Huang¹[0000-0001-9776-1786], Max Coleman¹[0009-0001-7693-1675], Daniela Gachago¹[0000-0003-0677-9273] and Jean-Paul Van Belle²[0000-0002-9140-0143]

^{1,2}University of Cape Town, Rondebosch, South Africa
²jean-paul.vanbelle@uct.ac.za

Abstract. Generative AI is about to radically transform the way intellectual and creative work is being done. Since the release of ChatGPT in late 2022, the potential impact of generative AI tools on higher education has been intensively debated. ChatGPT can generate well-formulated human-like text passages and conversations that is often, but not always, of a surprisingly high quality. This paper reports on an early experiment to explore ways in which ChatGPT can be used in the higher education context. The experiment involved a written assignment which required postgraduate Information Systems students to formulate a critique of the outputs of ChatGPT to a specific question in Information Systems project management. The paper investigates the level of criticality that the students demonstrated in working with ChatGPT and assessing the quality of its outputs. It further explores the claim that ChatGPT can be used to generate rubrics and assess students' assignments by asking ChatGPT to produce a rubric for critical thinking and assess the students' assignments against the rubric produced. The findings indicate that students perceive the ChatGPT produced responses as generally accurate, although they tend to lack depth, with some key information omitted, produced biased responses and have limitations with academic writing conventions. The rubric that ChatGPT produced for assessing critical thinking is lacking in certain areas and the reliability of using it as an assessment tool is questionable given the inconsistency in the results. Overall, the paper concludes that while ChatGPT and other text generative AI can be useful learning and teaching companions for both students and lectures, human expertise and judgement is needed in working with ChatGPT.

Keywords: ChatGPT; Teaching Tools; AI in Education; Generative AI; ICT in Education.

A Consolidated Catalogue of Question Types for Programming Courses

Anitta Thomas¹[0000-0002-3637-259X]

¹ School of Computing, The Science Campus, University of South Africa, Florida Park,
South Africa
thomaa@unisa.ac.za

Abstract. Formulating questions, whether for assessments or to create exercises, is an essential component of teaching programming. The development of well-constructed questions requires considerable effort. In this research, a consolidated catalogue of question types was developed to assist programming instructors with the formulation of questions. Existing classifications of question types in both programming and computing were consulted in developing the catalogue. The catalogue was constructed using a systematised literature review by referring to the literature and unifying different categories of question types described in the literature. The catalogue consists of fourteen base types of questions, with the possibility of variations within most base types. The catalogue is primarily intended to assist instructors in designing and formulating more diverse types of questions in programming courses. Additionally, the author envisages that this catalogue can serve as a starting point for discussions on how to map current educational taxonomies to question types, the different skills related to these questions, appropriate formats (for instance, multiple choice vs open format questions) and the appropriate qualification levels at which these questions can be presented.

Keywords: Question types, programming courses, programming assessments, classification of questions.

An Alumni Satisfaction Model for Computing Departments

Andre P. Calitz¹[0000-0002-2555-9041], Margaret Cullen²[0000-0003-4225-4191],
Arthur Glaum³[0009-0005-5037-5905], Jean Greyling⁴[0000-0002-6773-9200]

^{1,4}Department of Computing Sciences, Nelson Mandela University, Port Elizabeth, S.A.

^{2,3}Business School, Nelson Mandela University, Port Elizabeth, S.A.

¹Andre.Calitz@Mandela.ac.za, ²Margaret.Cullen@Mandela.ac.za,

³Arthur.Glaum@gmail.com, ⁴Jean.Greyling@Mandela.ac.za

Abstract. The perception of Alumni about the extent of learning and the usefulness of the knowledge gained is a key measure for education institutions to assess their success. Alumni are important stakeholders, specifically for Higher Education Institutions, as they work in industry and can provide valuable feedback on education and service offerings, including course content. The importance of Alumni satisfaction in relation to the success of universities makes it imperative that the factors driving satisfaction be determined. The purpose of this paper is to measure Alumni satisfaction in a department offering Computer Science and Information Systems qualifications and identify areas of improvement.

A hypothesised model of the factors influencing Alumni satisfaction was proposed and used to guide the quantitative research study. The research was exploratory and eight factors affecting Alumni satisfaction were investigated. A questionnaire was distributed to department's Alumni who work in the IT industry. The results indicate that the Alumni agreed that the academic staff maintained high academic standards, the degree programmes and modern technologies prepared them for the world of work. One factor that should be improved upon is departmental communications with the Alumni network. Technology and education are rapidly changing and departments must regularly determine Alumni satisfaction.

Keywords: Alumni satisfaction; Service guarantee; Stakeholder theory; Perceived value.

Motivations and Experiences of Recognition of Prior Learning Candidates in Information Systems Programmes

Thelma. M.T Chitsa^[0009-0004-3980-9701] and Gwamaka Mwalemba^[0000-0001-8067-0343]

Department of Information Systems, Faculty of Commerce, University of Cape Town,
Rondebosch, Cape Town 7700, South Africa
chttad005@myuct.ac.za, gt.mwalemba@uct.ac.za

Abstract. Technology is rapidly changing the landscape of the workforce and society. Business owners and other employers are actively seeking a labour force with the required digital skills and qualifications. Recognition of prior learning (RPL) has been identified as one of the means of upskilling the workforce and expanding access to formal qualifications for working adults. This research explores the motivations and experiences of professionals in technology-related fields seeking access to tertiary-level qualifications through the RPL route. Findings point to RPL fulfilling its intended role of being a necessity and, in some cases, the only means for experienced professionals without formal education to advance their skills and obtain formal qualifications necessary for their career advancement. There is also a need for higher learning institutions to extend their accessibility through similar programs and initiatives.

Keywords: Recognition of Prior Learning, ICT Industry, Skills Development

First Year Computing Students' Career Choice Influencers

Margaret Cullen¹[0000-0003-4225-4191], Andre P. Calitz²[0000-0002-2555-9041],
Malibongwe Twani³[0000-0002-9901-5848] and Jean Greyling⁴[0000-0002-6773-9200]

¹Nelson Mandela University Business School, Port Elizabeth, South Africa
^{2,3,4} Nelson Mandela University, Department of Computing Sciences, Port Elizabeth, South Africa

¹Margaret.Cullen@Mandela.ac.za, ²Andre.Calitz@Mandela.ac.za,
³Mtwani@Mandela.ac.za, ⁴Jean.Greyling@Mandela.ac.za

Abstract. Research indicates that first year students who have chosen a career in Computer Science (CS), Information Systems (IS), Information Technology (IT) and other related computing subjects were generally influenced by parents, teachers and role models. Current research indicates that exposure to new technologies, gaming and programming Apps at school level can influence a scholar's IT career choice. Theories relating to career choice have focused on the characteristics of individuals and their environment. Presently, CS/IS/IT departments do not know who or what influenced first year students to pursue a career in IT. Understanding first year students' academic IT career choices influencers would assist academic departments to improve methods and strategies to recruit first year CS, IS and IT students.

The purpose of this study was to determine who or what influenced current registered first year students to choose an IT career. A questionnaire was completed by first year CS, IS and IT students at a Comprehensive University. The findings contradict the literature, which states that parents and teachers, influence a first-year student's IT career choice. Thirty-five percent of the respondents did not know their father, 19% did not know their mother and 38% did not know their father's occupation. Social media and IT role models were important influencers for first year CS/IS/IT students speaking an African language at home. This study introduces an innovative gaming project introduced at schools in order to introduce school children, parents and teachers to coding.

Keywords: IT career choice, influencers, career advice.

Author Index

A

Adigun, Matthew 92

C

Calitz, Andre 141, 168, 170

Canani, Ammar 74

Chibaya, Collin 165

Chitsa, Thelma 169

Coetzee, Marijke 1

Coleman, Max 166

Cullen, Margaret 168, 170

D

Davids, Zane 108

De Agrela, Bronte 125

E

Esiefarienrhe, Bukohwo Michael 17

F

Finsterbusch, Bianca 125

G

Gachago, Daniela 166

Gain, James 164

Geeling, Sharon 108

Glaum, Arthur 168

Govender, Irene 59

Greyling, Jean 168, 170

Gundu, Tapiwa 165

H

Huang, Cheng-Wen 166

I

Isaks, Enwill 108

J

Janse van Rensburg, Theane 159

K

Kautondokwa, Popyeni 74

Khomokhoana, Pakiso 160

L

Le Roux, Daniel B 125

Levitt, Stephen 161

Liebenberg, Janet 162

M

Machanick, Philip 30

Matthee, Machdel 159

Mawela, Tendani	163
Mennega, Nita	163
Moemi, Thusoyaone Joseph	17
Muller, Jacqui	1
Mwalemba, Gwamaka	169
N	
Ndlovu, Thamsanqa	46
Nixon, Ken	161
O	
Oosterwyk, Grant	74
P	
Petratos, Anastasia	141
R	
Raghavjee, Rushil	59
S	
Safla, Aslam	164
Saunders, Jurgen	125
Seymour, Lisa	108
Subramaniam, Prabhakar	59
Suleman, Hussein	164
T	
Taylor, Estelle	141
Thomas, Anitta	167
Tsibolane, Pitso	74
Twani, Malibongwe	170
V	
Van Belle, Jean-Paul	166
Van der Linde, Suné	162
W	
Wario, Ruth	160
Z	
Zwane, Skhumbuzo	92