



A Project REPORT On

BUSINESS MANAGEMENT INFORMATION SYSTEM

BACHLOR IN COMPUTER APPLICATION (BCA)

UNDER THE GUIDENCE

XXXXXXXXXXXXXX

SUBMITTED BY:

NAME: - XXXXXXXXXXXXXXXXX

ROLL.NO:- XXXXXXXXX

IGNOU THE PEOPLES UNIVERSITY



A PROJECT REPORT
ON

"BUSINESS MANAGEMENT INFORMATION SYSTEM"

SUBMITTED BY :

NAME : XXXXXXXXXXXXX

ROLL NO.: xxxxxxxxxxxx

BUSINESS MANAGEMENT INFORMATION SYSTEM

SUBMITTED BY

**ROLL NO-XXXXXXX
XXXXXXX**

A project report submitted to IGNOU
in partial fulfilment of the requirements for
the degree BCA

2024

CERTIFICATE OF APPROVAL

Business Management Information System

The foregoing project entitled, “**Business Management Information System**” is hereby approved as a creditable study of research topic and has been presented in a satisfactory manner to warrant its acceptance as prerequisite to the degree for it was submitted.

It is understood that by this approval, the undersigned do not necessarily endorse any conclusion drawn or opinion expressed therein, but approve the thesis for the purpose for which it is submitted.

(Internal Examiner)

(External Examiner)

ACKNOWLEDGEMENT

I would like to take this opportunity to express my gratitude to the following people below who have directly or indirectly helped me during the course of completing this project.

I would like to express my most sincere gratitude to my Guide Mr. xxxxxxxxxxxxxxxx for his invaluable advice and patience throughout the whole course of this project. xxxxxxxxxxxx has imparted most of his knowledge in programming and good advice to me. When I encounter any problems, he will not hesitate to meet up with me and discuss without fail. I strongly believe that without Mr. xxxxxxxxxxxxxx patient and final year project will not be as successful as it is.

Last but not least, I will also like to take this last opportunity to thank all my fellow IGNOU classmates and friends of their continuous support throughout my stay in this College as well as their invaluable opinion, comment and ideas they had made on this project.

INTRODUCTION:-

Businesses use information systems at all levels of operation to collect, process, and store data. Management aggregates and disseminates this data in the form of information needed to carry out the daily operations of business. Everyone who works in business, from someone who pays the bills to the person who makes employment decisions, uses information systems. A car dealership could use a computer database to keep track of which products sell best. A retail store might use a computer-based information system to sell products over the Internet. In fact, many (if not most) businesses concentrate on the alignment of MIS with business goals to achieve competitive advantage over other businesses.

MIS professionals create information systems for data management (i.e., storing, searching, and analyzing data). In addition, they manage various information systems to meet the needs of managers, staff and customers. By working collaboratively with various members of their work group, as well as with their customers and clients, MIS professionals are able to play a key role in areas such as information security, integration, and exchange. As an MIS major, you will learn to design, implement, and use business information systems in innovative ways to increase the effectiveness and efficiency of your company.

INDEX

<u>S. N</u>	<u>CONTENTS</u>
1.	INTRODUCTION
2.	ANALYSIS
2.1	SYSTEM ANALYSIS
2.2	SYSTEM SPECIFICATIONS
3.	DESIGN APPROACH
3.1	INTRODUCTION TO DESIGN
3.2	UML DIAGRAMS
3.3	DATA FLOW DIAGRAMS
3.4	E-R DIAGRAMS
4.	PROJECT MODULES
5.	IMPLEMENTATION
4.1	CONCEPTS AND TECHNIQUES
4.2	TESTING
4.2.1	TEST CASES
6.	OUTPUT SCREENS
7.	CONCLUSION
8.	FUTURE ENHANCEMENTS
9.	BIBILIOGRAPHY

Introduction

INTRODUCTION:

Business information systems are sets of inter-related procedures using IT infrastructure in a business enterprise to generate and disseminate desired information.

Such systems are designed to support decision making by the people associated with the enterprise in the process of attainment of its objectives. The business information system gets data and other resources of IT infrastructure as input from the environment and process them to satisfy the information needs of different entities associated with the business enterprise.

Management Information System

There are systems of control over the use of IT resources and the feedback system offers useful clues for increasing the benefits of information systems to business. The business information systems are sub-systems of business system and by themselves serve the function of feedback and control in business system.

Features of Business Information System:

Characteristics & features of business information system are:

1. The business information systems are subject to the dynamics of business environment and need to be flexible enough to absorb the inevitable changes in the information needs of business. They have to be efficient to satisfy the demanding and 'hard task masters,' the business

managers. Thus, there is need to balance the conflicting objectives in the process of designing business information systems.

2. Business information systems need to be proactive. They should anticipate changes in information needs of users and accordingly adapt themselves to suit their needs. This has become important because of the fact that the managers get involved in the routine activities to the extent that the decision making becomes a matter of imitating what competitors are doing or planning to do, rather than making an informed choice.

3. The purpose of business information system is to cater to the information needs for decision making in business.

4. The business information systems have to be designed keeping in view the availability of financial and human resources to the business enterprise.

5. The cost effectiveness is a matter of prime concern in the development and maintenance of business information systems. Economic justification for investment in IT infrastructure for business information systems is a pre condition for its existence and sustenance.

Key Components of Business Information System:

Information systems can be described by four of their key components which are:

1. Decisions
2. Transactions and processing
3. Information and its flow

Analysis

SYSTEM ANALYSIS:

Management information systems can help you make valid decisions by providing accurate and up-to-date information and performing analytic functions. You have to make sure the management information system you choose can work with the information formats available in your company and has the features you need. Suitable management information systems can structure the basic data available from your company operations and records into reports to present you with guidance for your decisions.

Information

When you base your decisions on data available from management information systems, they reflect information that comes from the operations of your company. Management information systems take data generated by the working level and organize it into useful formats. Management information systems typically contain sales figures, expenses, investments and workforce data. If you need to know how much profit your company has made each year for the past five years to make a decision, management information systems can provide accurate reports giving you that information.

Scenarios

The capability to run scenarios is a key decision-making tool. Some management information systems have this feature built in, while others can provide the information required for running scenarios on other applications, such as spreadsheets. Your decision is influenced by what happens if you decide a certain way. What-if scenarios show you how different variables change when you make a decision. You can enter reduced staff levels or increased promotion budgets and see what happens to revenue, expenses and profit for different levels of cuts or increases. Management information systems play a critical role in making realistic scenarios possible.

Projections

Any decisions you make result in changes in the projected company results and may require modifications to your business strategy and overall goals. Management information systems either have trend analysis built in or can provide information that lets you carry out such an analysis. Typical business strategies include projections for all fundamental operating results. A trend analysis allows you to show what these results would be in the current situation and how they will change once you have implemented the decisions you have taken. The new values form the basis of your strategic approach going forward.

Implementation

While you make your decisions with specific goals in mind and have the documentation from management information systems and trend analysis to support your expectations, you have to track company results to make sure they develop as planned. Management information systems give you the data you need to determine whether your decisions have had the desired effect, or whether you have to take corrective action to reach your goals. If specific results are not on track, you can use management information systems to evaluate the situation and decide to take additional measures if necessary.

System Specifications

Hardware Requirements:-

- Pentium-IV(Processor).
- 256 MB Ram
- 512 KB Cache Memory
- Hard disk 10 GB
- Microsoft Compatible 101 or more Key Board

Software Requirements: -

- **Operating System :** Windows
- **Web-Technology:** PHP
- **Front-End:** HTML,CSS,JAVASCRIPT
- **Back-End:** MySQL
- **Web Server:** Apache SERVER.

Design

INTRODUCTION:

Design is the first step in the development phase for any techniques and principles for the purpose of defining a device, a process or system in sufficient detail to permit its physical realization.

Once the software requirements have been analyzed and specified the software design involves three technical activities - design, coding, implementation and testing that are required to build and verify the software.

The design activities are of main importance in this phase, because in this activity, decisions ultimately affecting the success of the software implementation and its ease of maintenance are made. These decisions have the final bearing upon reliability and maintainability of the system. Design is the only way to accurately translate the customer's requirements into finished software or a system.

Design is the place where quality is fostered in development. Software design is a process through which requirements are translated into a representation of software. Software design is conducted in two steps. Preliminary design is concerned with the transformation of requirements into data.

UML Diagrams:

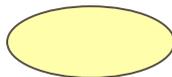
Actor:

A coherent set of roles that users of use cases play when interacting with the use cases.



Use case:

A description of sequence of actions, including variants, that a system performs that yields an observable result of value of an actor.



UML stands for Unified Modeling Language. UML is a language for specifying, visualizing and documenting the system. This is the step while developing any product after analysis. The goal from this is to produce a model of the entities involved in the project which later need to be built. The representation of the entities that are to be used in the product being developed need to be designed.

There are various kinds of methods in software design:

They are as follows:

- Use case Diagram
- Sequence Diagram
- Collaboration Diagram
- Activity Diagram
- State chat Diagram

USECASE DIAGRAMS:

Use case diagrams model behavior within a system and helps the developers understand of what the user require. The stick man represents what's called an actor. Use case diagram can be useful for getting an overall view of the system and clarifying who can do and more importantly what they can't do.

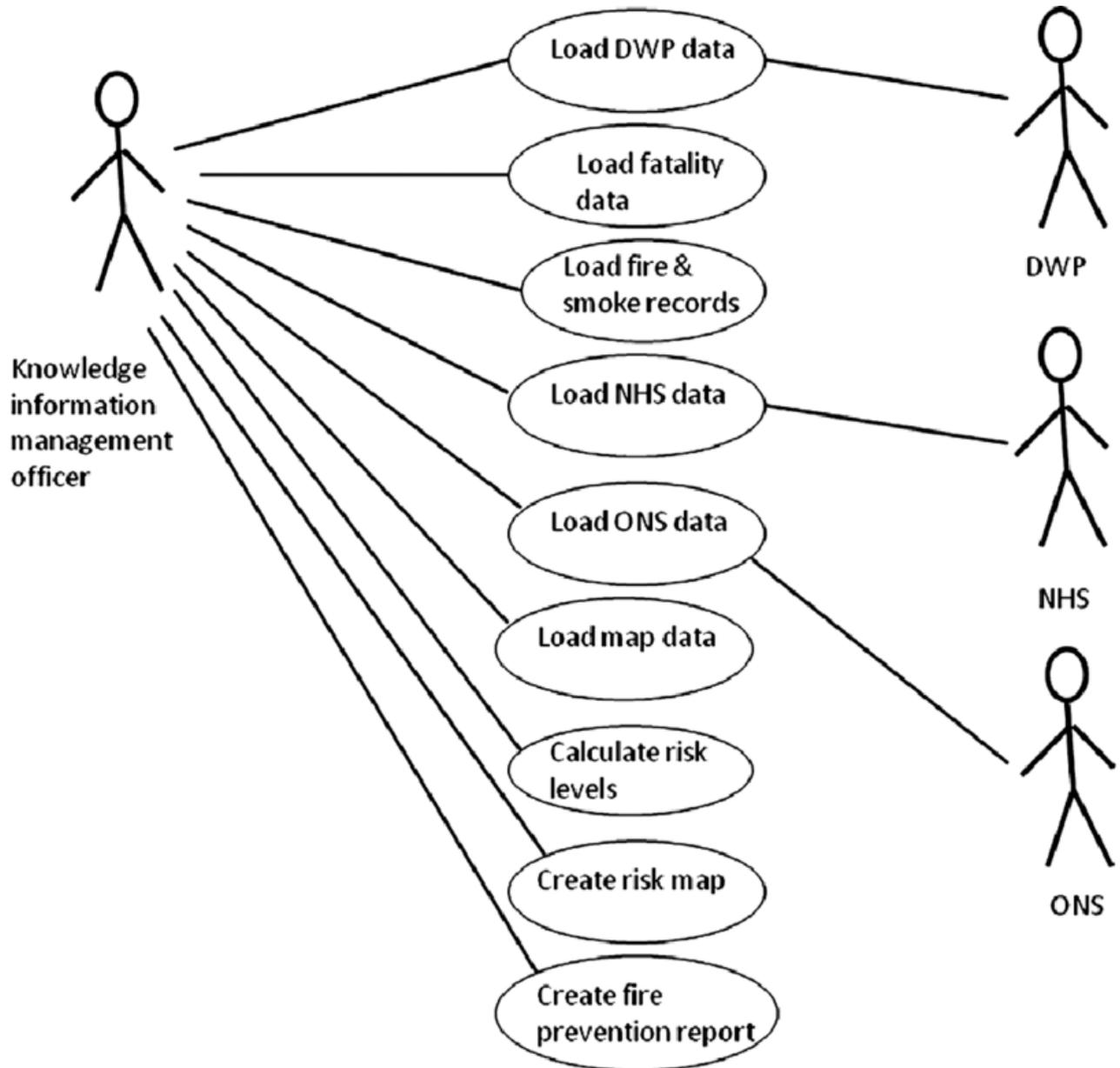
Use case diagram consists of use cases and actors and shows the interaction between the use case and actors.

- The purpose is to show the interactions between the use case and actor.
- To represent the system requirements from user's perspective.
- An actor could be the end-user of the system or an external system.

USECASE DIAGRAM:

A Use case is a description of set of sequence of actions. Graphically it is rendered as an ellipse with solid line including only its name. Use case diagram is a behavioral diagram that shows a set of use cases and actors and their relationship. It is an association between the use cases and actors. An actor represents a real-world object. Primary Actor – Sender, Secondary ActorReceiver.

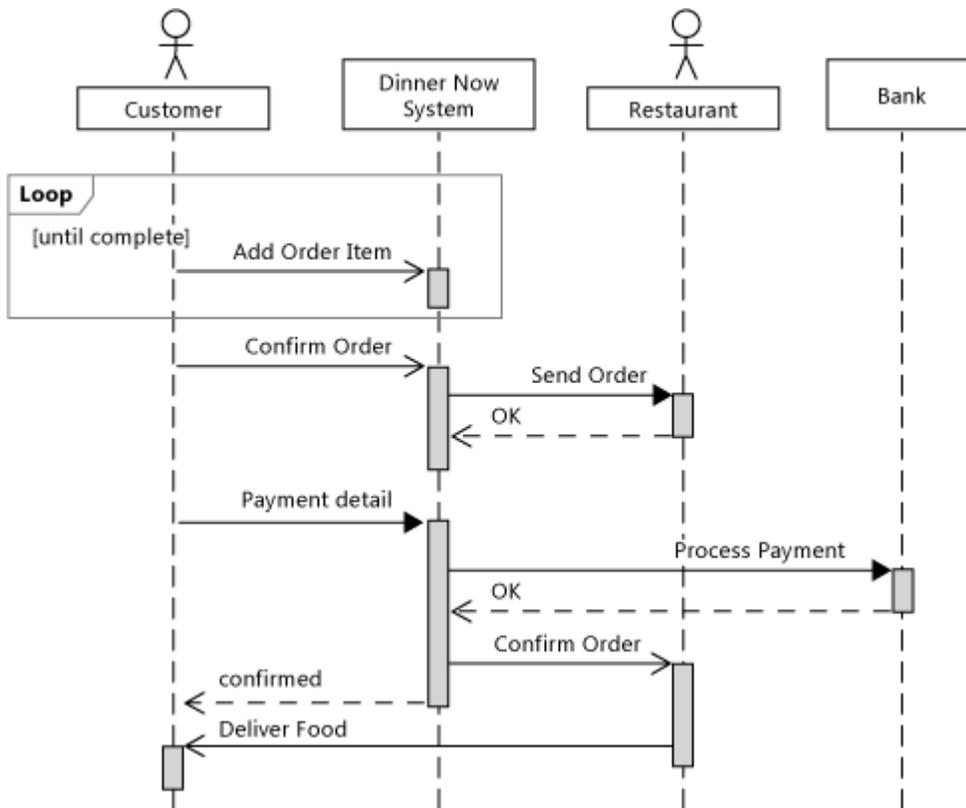
Business Management Information System



SEQUENCE DIAGRAM:

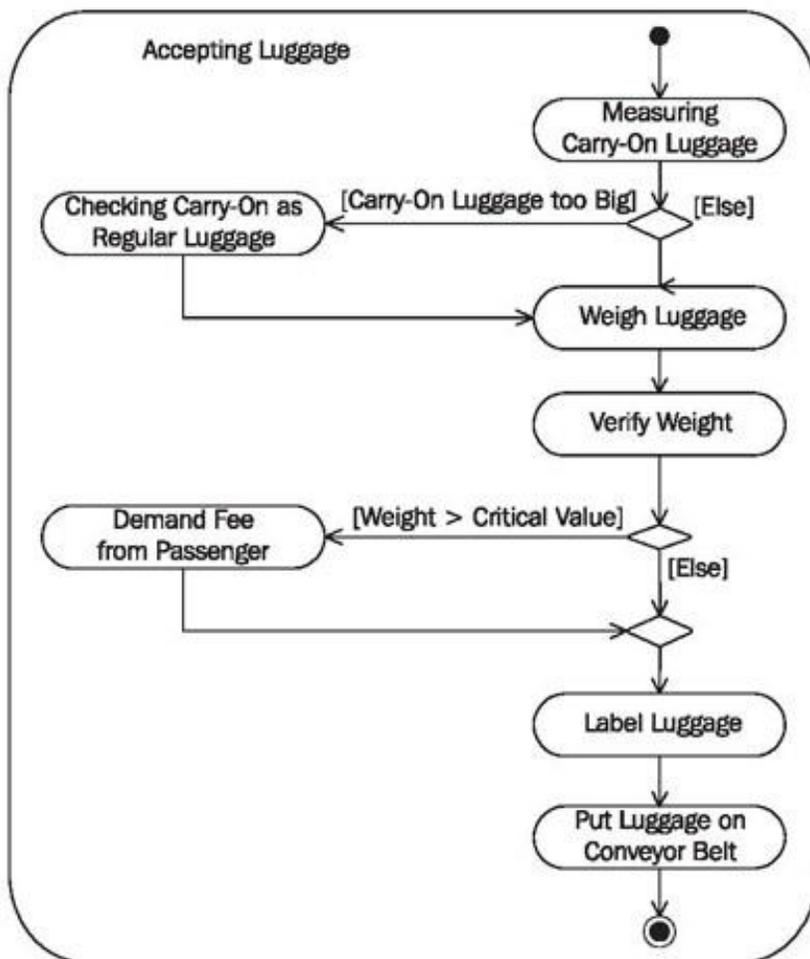
Sequence diagram and collaboration diagram are called INTERACTION DIAGRAMS. An interaction diagram shows an interaction, consisting of set of objects and their relationship including the messages that may be dispatched among them.

A sequence diagram is an introduction that empathizes the time ordering of messages. Graphically a sequence diagram is a table that shows objects arranged along the X-axis and messages ordered in increasing time along the Y-axis



COLLABORATION DIAGRAM:

A collaboration diagram is an introduction diagram that emphasizes the structural organization of the objects that send and receive messages. Graphically a collaboration diagram is a collection of vertices and arcs.



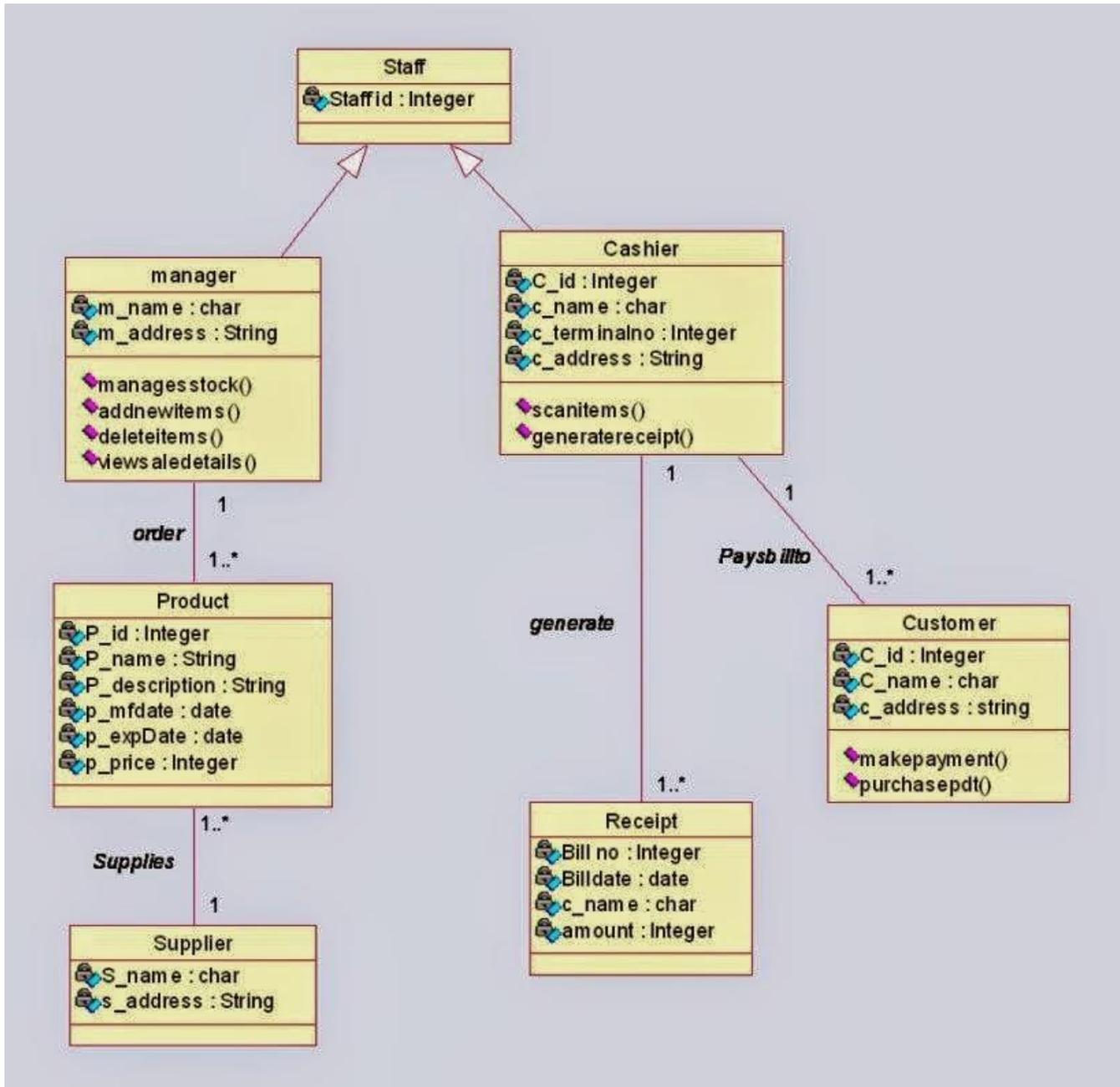
CLASS DIAGRAM:

Class is nothing but a structure that contains both variables and methods. The Class Diagram shows a set of classes, interfaces, and collaborations and their relationships. There is most common diagram in modeling the object oriented systems and are used to give the static view of a system. It shows the dependency between the classes that can be used in our system.

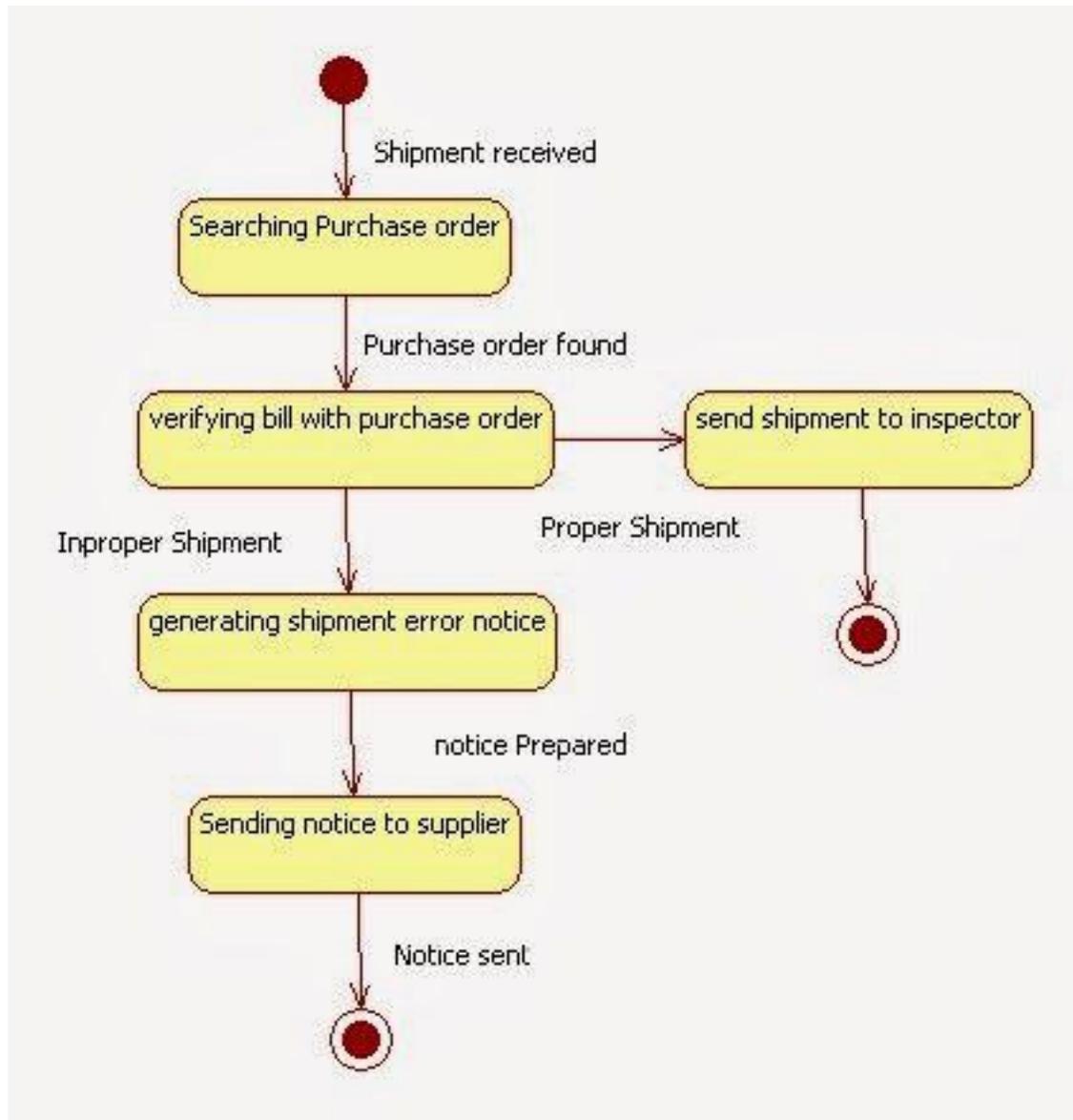
The interactions between the modules or classes of our projects are shown below. Each block contains Class Name, Variables and Methods.

CLASS:

A description of set of objects that share the same attributes, operations, relationships, and semantics



State Chart Diagram



DATA FLOW DIAGRAMS:

The DFD takes an input-process-output view of a system i.e. data objects flow into the software, are transformed by processing elements, and resultant data objects flow out of the software.

Data objects represented by labeled arrows and transformation are represented by circles also called as bubbles. DFD is presented in a hierarchical fashion i.e. the first data flow model represents the system as a whole. Subsequent DFD refine the context diagram (level 0 DFD), providing increasing details with each subsequent level.

The DFD enables the software engineer to develop models of the information domain & functional domain at the same time. As the DFD is refined into greater levels of details, the analyst perform an implicit functional decomposition of the system. At the same time, the DFD refinement results in a corresponding refinement of the data as it moves through the process that embody the applications.

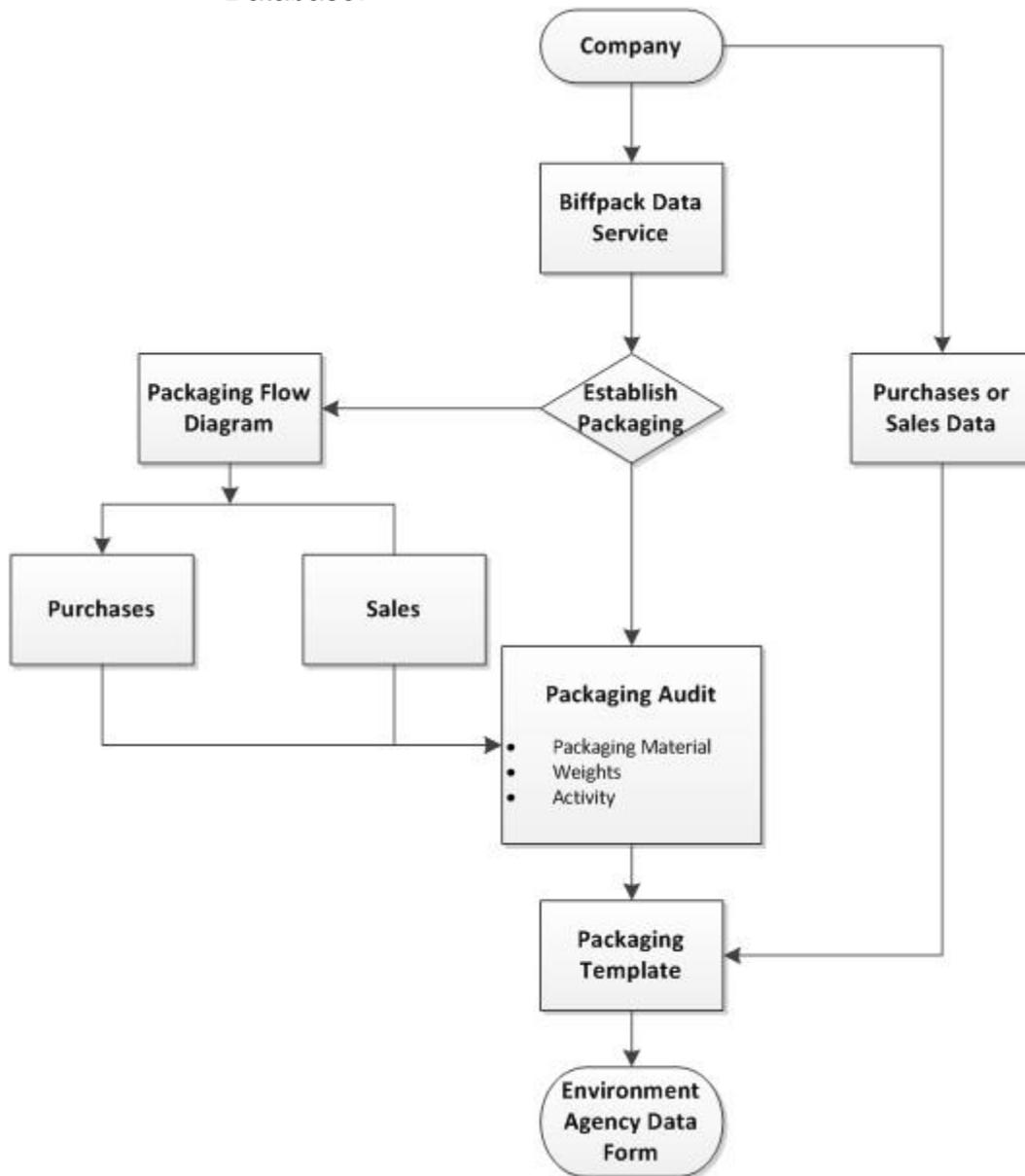
A context-level DFD for the system the primary external entities produce information for use by the system and consume information generated by the system. The labeled arrow represents data objects or object hierarchy.

RULES FOR DFD:

- Fix the scope of the system by means of context diagrams.
- Organize the DFD so that the main sequence of the actions
- Reads left to right and top to bottom.
- Identify all inputs and outputs.
- Identify and label each process internal to the system with Rounded circles.
- A process is required for all the data transformation and Transfers. Therefore, never connect a data store to a data Source or the destinations or another data store with just a Data flow arrow.
- Do not indicate hardware and ignore control information.
- Make sure the names of the processes accurately convey everything the process is done.
- There must not be unnamed process.
- Indicate external sources and destinations of the data, with Squares.
- Number each occurrence of repeated external entities.
- Identify all data flows for each process step, except simple Record retrievals.
- Label data flow on each arrow.
- Use details flow on each arrow.
- Use the details flow arrow to indicate data movements.

DATAFLOW DIAGRAMS:

Database:



E-R Diagrams:

The Entity-Relationship (ER) model was originally proposed by Peter in 1976 [Chen76] as a way to unify the network and relational database views. Simply stated the ER model is a conceptual data model that views the real world as entities and relationships. A basic component of the model is the Entity-Relationship diagram which is used to visually represents data objects. Since Chen wrote his paper the model has been extended and today it is commonly used for database design For the database designer, the utility of the ER model is:

- it maps well to the relational model. The constructs used in the ER model can easily be transformed into relational tables.
- it is simple and easy to understand with a minimum of training. Therefore, the model can be used by the database designer to communicate the design to the end user.
- In addition, the model can be used as a design plan by the database developer to implement a data model in a specific database management software.

Connectivity and Cardinality

The basic types of connectivity for relations are: one-to-one, one-to-many, and many-to-many. A *one-to-one* (1:1) relationship is when at most one instance of a entity A is associated with one instance of entity B. For example, "employees in the company are each assigned their own office. For each employee there exists a unique office and for each office there exists a unique employee.

A *one-to-many* (1:N) relationships is when for one instance of entity A, there are zero, one, or many instances of entity B, but for one instance of entity B, there is only one instance of entity A. An example of a 1:N relationships is

a department has many employees

each employee is assigned to one department

A *many-to-many* (M:N) relationship, sometimes called non-specific, is when for one instance of entity A, there are zero, one, or many instances of entity B and for one instance of entity B there are zero, one, or many instances of entity A. The connectivity of a relationship describes the mapping of associated

ER Notation

There is no standard for representing data objects in ER diagrams. Each modeling methodology uses its own notation. The original notation used by Chen is widely used in academics texts and journals but rarely seen in either CASE tools or publications by non-academics. Today, there are a number of notations used, among the more common are Bachman, crow's foot, and IDEFIX.

All notational styles represent entities as rectangular boxes and relationships as lines connecting boxes. Each style uses a special set of symbols to represent the cardinality of a connection. The notation used in this document is from Martin. The symbols used for the basic ER constructs are:

- **entities** are represented by labeled rectangles. The label is the name of the entity. Entity names should be singular nouns.
- **relationships** are represented by a solid line connecting two entities. The name of the relationship is written above the line. Relationship names should be verbs
- **attributes**, when included, are listed inside the entity rectangle. Attributes which are identifiers are underlined. Attribute names should be singular nouns.
- **cardinality** of many is represented by a line ending in a crow's foot. If the crow's foot is omitted, the cardinality is one.
- **existence** is represented by placing a circle or a perpendicular bar on the line. Mandatory existence is shown by the bar (looks like a 1) next to the entity for an instance is required. Optional existence is shown by placing a circle next to the entity that is optional

MODULES

OVERVIEW OF TECHNOLOGIES USED

PHP

PHP: Hypertext Preprocessor, is a widely used, general-purpose scripting language that was originally designed for web development, to produce dynamic web pages. It can be embedded into HTML and generally runs on a web server, which needs to be configured to process PHP code and create web page content from it. It can be deployed on most web servers and on almost every operating system and platform free of charge.

PHP was originally created by Rasmus Lerdorf in 1995 and has been in continuous development ever since. The main implementation of PHP is now produced by The PHP Group and serves as the de facto standard for PHP as there is no formal specification. PHP is free software released under the PHP License, which is incompatible with the GNU General Public License (GPL) because of restrictions on the use of the term PHP

PHP has evolved to include a command line interface capability and can also be used in standalone graphical applications.

USAGE

PHP is a general-purpose scripting language that is especially suited for web development. PHP generally runs on a web server. Any PHP code in a requested file is executed by the PHP runtime, usually to create dynamic web page content. It can also be used for command-line scripting and client-side GUI applications. PHP can be deployed on most web servers, many operating systems and platforms, and can be used with many relational database management systems. It is available free of charge, and the PHP Group provides the complete source code for users to build, customize and extend for their own use.

PHP primarily acts as a filter, taking input from a file or stream containing text and/or PHP instructions and outputs another stream of data; most commonly the output will be HTML. Since PHP 4, the PHP parser compiles input to produce byte code for processing by the Zend Engine, giving improved performance over its interpreter predecessor

Originally designed to create dynamic web pages, PHP now focuses mainly on server-side scripting, and it is similar to other server-side scripting languages that provide dynamic content from a web server to a client, such as Microsoft's Active Server Pages, Sun Microsystems' JavaServer Pages and mod_perl. PHP has also attracted the development of many frameworks that provide building blocks and a design structure to promote rapid application development (RAD). Some of these include CakePHP, Symfony, CodeIgniter, and Zend Framework, offering features similar to other web application frameworks.

About HTML

HTML, which stands for **Hyper Text Markup Language**, is the predominant markup language for web pages. It provides a means to create structured documents by denoting structural semantics for text such as headings, paragraphs, lists etc as well as for links, quotes, and other items. It allows images and objects to be embedded and can be used to create interactive forms. It is written in the form of HTML elements consisting of "tags" surrounded by angle brackets within the web page content. It can include or can load scripts in languages such as JavaScript which affect the behavior of HTML processors like Web browsers; and Cascading Style Sheets (CSS) to define the appearance and layout of text and other material. The W3C, maintainer of both HTML and CSS standards, encourages the use of CSS over explicit presentational markup.

Hyper Text Markup Language(HTML) is the encoding scheme used to create and format a web document. A user need not be an expert programmer to make use of HTML for creating hypertext documents that can be put on the internet.

Most graphical e-mail clients allow the use of a subset of HTML (often ill-defined) to provide formatting and semantic markup not available with plain text. This may include typographic information like coloured headings, emphasized and quoted

text, inline images and diagrams. Many such clients include both a GUI editor for composing HTML e-mail messages and a rendering engine for displaying them. Use of HTML in e-mail is controversial because of compatibility issues, because it can help disguise phishing attacks, because it can confuse spam filters and because the message size is larger than plain text.

NAMING CONVENTIONS

The most common filename extension for files containing HTML is .html. A common abbreviation of this is .htm, which originated because some early operating systems and file systems, such as DOS and FAT, limited file extensions to three letters.

HTML APPLICATION

An HTML Application is a Microsoft Windows application that uses HTML and Dynamic HTML in a browser to provide the application's graphical interface. A regular HTML file is confined to the security model of the web browser, communicating only to web servers and manipulating only webpage objects and site cookies. An HTA runs as a fully trusted application and therefore has more privileges, like creation/editing/removal of files and Windows Registry entries. Because they operate outside the browser's security model, HTAs cannot be executed via HTTP, but must be downloaded (just like an EXE file) and executed from local file system

ABOUT JAVASCRIPT

JavaScript is an object-oriented scripting language used to enable programmatic access to objects within both the client application and other applications. It is primarily used in the form of client-side JavaScript, implemented as an integrated component of the web browser, allowing the development of enhanced user interfaces and dynamic websites. JavaScript is a dialect of the ECMAScript standard and is characterized as a dynamic, weakly typed, prototype-based language with first-class functions. JavaScript was influenced by many languages and was designed to look like Java, but to be easier for non-programmers to work with.

PROTOTYPE-BASED

JavaScript uses prototypes instead of classes for inheritance. It is possible to simulate many class-based features with prototypes in JavaScript.

Functions double as object constructors along with their typical role. Prefixing a function call with `new` creates a new object and calls that function with its local `this` keyword bound to that object for that invocation. The constructor's prototype property determines the object used for the new object's internal prototype. JavaScript's built-in constructors, such as `Array`, also have prototypes that can be modified.

Unlike many object-oriented languages, there is no distinction between a function definition and a method definition. Rather, the distinction occurs during function calling; a function can be called as a method. When a function is called as a method of an object, the function's local `this` keyword is bound to that object for that invocation.

USAGE

The primary use of JavaScript is to write functions that are embedded in or included from HTML pages and interact with the Document Object Model (DOM) of the page.

Because JavaScript code can run locally in a user's browser (rather than on a remote server) it can respond to user actions quickly, making an application feel more responsive. Furthermore, JavaScript code can detect user actions which HTML alone cannot, such as individual keystrokes. Applications such as Gmail take advantage of this: much of the user-interface logic is written in JavaScript, and JavaScript dispatches requests for information (such as the content of an e-mail message) to the server. The wider trend of Ajax programming similarly exploits this strength.

A JavaScript engine (also known as *JavaScript interpreter* or *JavaScript implementation*) is an interpreter that interprets JavaScript source code and executes the script accordingly. The first JavaScript engine was created by Brendan Eich at Netscape Communications Corporation, for the Netscape Navigator web browser. A web browser is by far the most common host environment for JavaScript. Web browsers typically use the public API to create "host objects" responsible for reflecting the DOM into JavaScript.

ABOUT MySQL

MySQL Introduction

There are a large number of database management systems currently available, some commercial and some free.

Some of them : Oracle, Microsoft Access, Mysql and PostgreSQL.

These database systems are powerful, feature-rich software, capable of organizing and searching millions of records at very high speeds.

Understanding Databases, Records, and Primary Keys

Every Database is composed of one or more tables.

These Tables, which structure data into rows and columns, Impose organization on the data.

The records in a table(below) are not arranged in any particular order.

To make it easy to identify a specific record,therefore, it becomes necessary

standing Relationships and Foreign Keys(RDBMS)

You already know that a single database can hold multiple tables.

In a Relational database management system(RDBMS), these tables can be linked to each other by one or more common fields, called foreign keys.

What is Database administrator(DBA) ?

Database administrator is the super user of database, he has unrestricted rights and privileges to access database, grant permission to other database users.

What is Database user(DBU) ?

Database user is the person who uses the database in a restricted privileges, provided by database administrator.

Download MySQL Database

If you have installed PHP's WAMP or XAMPP server, then mysql database already exists. if you don't have then download mysql database from here <http://www.mysql.com>

DATABASE TABLES:

```

Database: `eman`
--
CREATE DATABASE `eman` DEFAULT CHARACTER SET latin1 COLLATE
latin1_swedish_ci;
USE `eman`;

Table structure for table `leaks`
--

CREATE TABLE IF NOT EXISTS `leaks` (
  `PNo` int(30) NOT NULL,
  `Product` varchar(200) NOT NULL,
  `DDate` varchar(12) NOT NULL,
  `MDate` varchar(12) NOT NULL,
  `Quantity` int(30) NOT NULL,
  `Usern` varchar(100) NOT NULL,
  `LeakID` int(20) NOT NULL AUTO_INCREMENT,
  PRIMARY KEY (`LeakID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

--
-- Dumping data for table `leaks`
--

INSERT INTO `leaks` (`PNo`, `Product`, `DDate`, `MDate`, `Quantity`,
`Usern`, `LeakID`) VALUES
(2, 'Bottle Water 5 ltrs', '06/02/2010', '06/02/2010', 1, '', 1);

-----

--
-- Table structure for table `tbl_exp`
--

CREATE TABLE IF NOT EXISTS `tbl_exp` (
  `refno` varchar(30) NOT NULL,

```

```
`Description` varchar(500) NOT NULL,  
`DDate` varchar(12) NOT NULL,  
`Reason` varchar(500) NOT NULL,  
`Cost` int(30) NOT NULL,  
`Remark` varchar(500) NOT NULL,  
`ExpID` int(20) NOT NULL AUTO_INCREMENT,  
`usern` varchar(20) NOT NULL,  
PRIMARY KEY (`ExpID`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;  
  
--  
-- Dumping data for table `tbl_exp`  
--  
  
INSERT INTO `tbl_exp` (`refno`, `Description`, `DDate`, `Reason`, `Cost`,  
`Remark`, `ExpID`, `usern`) VALUES  
( '1', 'NEPA', '06/08/2010', 'For NEPA', 5000, 'Just a part payment for  
June', 1, '' );  
  
-----  
  
--  
-- Table structure for table `tbl_pack`  
--  
  
CREATE TABLE IF NOT EXISTS `tbl_pack` (  
  `PackId` int(200) NOT NULL AUTO_INCREMENT,  
  `Pack` varchar(200) NOT NULL,  
  `PiecePack` int(200) NOT NULL,  
  `PricePiece` int(200) NOT NULL,  
  `PricePack` int(200) NOT NULL,  
  PRIMARY KEY (`PackId`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;  
  
--  
-- Dumping data for table `tbl_pack`  
--  
  
INSERT INTO `tbl_pack` (`PackId`, `Pack`, `PiecePack`, `PricePiece`,  
`PricePack`) VALUES  
(1, 'Bottle Water 5 ltrs', 12, 100, 1200);  
  
-----  
  
--  
-- Table structure for table `tbl_rawmat`  
--  
  
CREATE TABLE IF NOT EXISTS `tbl_rawmat` (  

```

```

`RawMatId` int(200) NOT NULL AUTO_INCREMENT,
`RefNo` varchar(200) NOT NULL,
`MType` varchar(200) NOT NULL,
`Quantity` int(200) NOT NULL,
`Meas` varchar(20) NOT NULL,
`Amount` int(200) NOT NULL,
`DDate` varchar(12) NOT NULL,
PRIMARY KEY (`RawMatId`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=2 ;

--
-- Dumping data for table `tbl_rawmat`
--

INSERT INTO `tbl_rawmat` (`RawMatId`, `RefNo`, `MType`, `Quantity`, `Meas`,
`Amount`, `DDate`) VALUES
(1, 'hdhd', 'Polythene', 23, 'Pieces', 2400, '06/02/2010');

-----

--
-- Table structure for table `tbl_sales`
--

CREATE TABLE IF NOT EXISTS `tbl_sales` (
  `SalesID` int(30) NOT NULL AUTO_INCREMENT,
  `refno` varchar(200) NOT NULL,
  `Product` varchar(200) NOT NULL,
  `Price` int(30) NOT NULL,
  `DDate` varchar(12) NOT NULL,
  `Amount` int(30) NOT NULL,
  `Quantity` int(30) NOT NULL,
  `usern` varchar(100) NOT NULL,
  `Meas` varchar(12) NOT NULL,
  PRIMARY KEY (`SalesID`)
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;

--
-- Dumping data for table `tbl_sales`
--

INSERT INTO `tbl_sales` (`SalesID`, `refno`, `Product`, `Price`, `DDate`,
`Amount`, `Quantity`, `usern`, `Meas`) VALUES
(1, '1', 'Bottle Water 5 ltrs', 100, '06/09/2010', 100, 1, '', 'Pieces'),
(3, '2', 'Bottle Water 5 ltrs', 100, '02/20/2012', 1200, 12, '', 'Pieces');

-----

--

```

```
-- Table structure for table `tbl_stock`  
--
```

```
CREATE TABLE IF NOT EXISTS `tbl_stock` (  
  `StockID` int(200) NOT NULL AUTO_INCREMENT,  
  `Product` varchar(200) NOT NULL,  
  `DDate` varchar(12) NOT NULL,  
  `Quantity` int(200) NOT NULL,  
  `PNo` int(200) NOT NULL,  
  `Meas` varchar(20) NOT NULL,  
  PRIMARY KEY (`StockID`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=4 ;
```

```
--  
-- Dumping data for table `tbl_stock`  
--
```

```
INSERT INTO `tbl_stock` (`StockID`, `Product`, `DDate`, `Quantity`, `PNo`,  
`Meas`) VALUES  
(3, 'Bottle Water 5 ltrs', '06/09/2010', 24, 3, 'Pieces'),  
(1, 'Bottle Water 5 ltrs', '06/05/2010', 12, 1, 'Packs'),  
(2, 'Bottle Water 5 ltrs', '06/02/2010', 2, 2, 'Packs');
```

```
-----  
--  
-- Table structure for table `tbl_user`  
--
```

```
CREATE TABLE IF NOT EXISTS `tbl_user` (  
  `userID` int(30) NOT NULL AUTO_INCREMENT,  
  `uname` varchar(100) NOT NULL,  
  `name` varchar(100) NOT NULL,  
  `pword` varchar(200) NOT NULL,  
  `Telephone` varchar(20) NOT NULL,  
  `roles` varchar(20) NOT NULL,  
  PRIMARY KEY (`userID`)  
) ENGINE=MyISAM DEFAULT CHARSET=latin1 AUTO_INCREMENT=3 ;
```

```
--  
-- Dumping data for table `tbl_user`  
--
```

```
INSERT INTO `tbl_user` (`userID`, `uname`, `name`, `pword`, `Telephone`,  
`roles`) VALUES  
(1, 'segun', 'Akinyemi Olusegun', 'samson', '08066405654', 'Admin'),  
(2, 'dada', 'Dada Edusetan', 'admin', '09087644556', 'User');
```

1. Index.html

```
2. <?php
3. //include("conit.php");
4. //$qry="select * from tbl_application";
5. //$res=mysql_query($qry);
6. //$rd=mysql_fetch_assoc($res);
7. //$nm=mysql_num_rows($res);
8. ?>
9.
10.<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Strict//EN"
    "http://www.w3.org/TR/xhtml1/DTD/xhtml1-strict.dtd">
11.<html xmlns="http://www.w3.org/1999/xhtml" lang="en">
12.
13.<!-- Version: Multiflex-3.12 / Header-1 (Default) -->
14.<!-- Date: January 20, 2008 -->
15.<!-- Design: www.1234.info -->
16.<!-- License: Fully open source without restrictions. -->
17.<!-- Please keep footer credits with the words -->
18.<!-- "Design by 1234.info" Thank you! -->
19.
20.<head>
21. <meta http-equiv="content-type" content="text/html; charset=utf-8"
    />
22. <meta http-equiv="cache-control" content="no-cache" />
23. <meta http-equiv="expires" content="3600" />
24. <meta name="revisit-after" content="2 days" />
25. <meta name="robots" content="index,follow" />
26. <meta name="publisher" content="Your publisher infos here ..." />
27. <meta name="copyright" content="Your copyright infos here ..." />
28. <meta name="author" content="Design: 1234.info / Modified: Your
    Name" />
29. <meta name="distribution" content="global" />
30. <meta name="description" content="Your page description here ..." />
31. <meta name="keywords" content="Your keywords, keywords, keywords,
    here ..." />
32. <link rel="stylesheet" type="text/css"
    media="screen,projection,print" href="./css/header1_setup.css" />
33.
34. <script language="javascript" type="text/javascript"
    src="css/niceforms.js"></script>
35.<script language="javascript" type="text/javascript"
    src="css/xmlwrap.js"></script>
36.
37.
38. <link rel="icon" type="image/x-icon" href="./img/favicon.ico" />
39. <title>e-Man</title>
40. <script type="text/javascript" src="jquery.min.js"></script>
```

```
41.
42.<script type="text/javascript" src="ddaccordion.js">
43.
44./*****
45.* Accordion Content script- (c) Dynamic Drive DHTML code library
   (www.dynamicdrive.com)
46.* Visit http://www.dynamicDrive.com for hundreds of DHTML scripts
47.* This notice must stay intact for legal use
48.*****/
49.
50.</script>
51.
52.
53.<script type="text/javascript">
54.
55.
56.
57.</script>
58. <style type="text/css">
59.<!--
60.
61.
62.#apDiv1 {
63.position:absolute;
64.width:99.3%;
65.height:75px;
66.z-index:1;
67.left: 6px;
68.top: 29px;
69.}
70.
71..arrowlistmenu{
72.width: 200px; /*width of accordion menu*/
73.
74.}
75.
76..arrowlistmenu .menuheader{ /*CSS class for menu headers in general
   (expanding or not!)*/*
77.color:#FFFFFF;
78.background: black url(titlebar.png) repeat-x center left;
79.margin-bottom: 3px; /*bottom spacing between header and rest of
   content*/
80.
81.cursor: pointer;
82.text-align:left;
83.font-family: Tahoma;
84.font-size: 14px;
85.text-decoration: none;
86.
```

```
87.
88.}
89..glossymenu{
90.margin: auto;
91.padding: 0;
92.width: 180px;
93.border-bottom-width: 0;
94.border-top-width: 1px;
95.border-right-width: 1px;
96.border-left-width: 1px;
97.border-top-style: solid;
98.border-right-style: solid;
99.border-bottom-style: solid;
100.border-left-style: solid;
101.border-top-color: #BEBEBE;
102.border-right-color: #BEBEBE;
103.border-bottom-color: #BEBEBE;
104.border-left-color: #BEBEBE;
105.
106.
107.}
108.
109..glossymenu a.menuitem{
110.color:#333333;
111.display: block;
112.position: relative; /*To help in the anchoring of the ".statusicon"
    icon image*/
113.width: auto;
114.padding: 4px 0;
115.padding-left: 10px;
116.text-decoration: none;
117.background-image:url(glossyback.gif);
118.background-repeat: no-repeat;
119.
120.font-family: arial;
121.font-size: 12px;
122.font-weight: bold;
123.}
124.
125..mxx{
126.color:#1C1C1C;
127.display: block;
128.position: relative; /*To help in the anchoring of the ".statusicon"
    icon image*/
129.width: auto;
130.padding: 4px 0;
131.padding-left: 10px;
132.text-decoration: none;
133.background-image: url(sis.png);
```

```
134.background-repeat: no-repeat;
135.background-color: #EDEDEB;
136.font-family: tahoma;
137.font-size: 11px;
138.border: thin solid #DEDEDE;
139.}
140.
141.
142..mxx:hover{
143.color: #666666;
144.display: block;
145.position: relative; /*To help in the anchoring of the ".statusicon"
    icon image*/
146.width: auto;
147.padding: 4px 0;
148.padding-left: 10px;
149.text-decoration: none;
150.background-image: url(sts.png);
151.background-repeat: no-repeat;
152.
153.font-family: Tahoma;
154.font-size: 10px;
155.font-weight: bold;
156.border: thin solid #666666;
157.
158.
159.}
160.
161.
162..glossymenu a.menuitem:visited, .glossymenu .menuitem:active{
163.color: #333333;
164.}
165.
166..glossymenu a.menuitem .statusicon{ /*CSS for icon image that gets
    dynamically added to headers*/
167.position: absolute;
168.top: 5px;
169.right: 5px;
170.border: none;
171.}
172.
173..glossymenu a.menuitem:hover{
174.background-image: url(glossyback2.png);
175.}
176.
177..glossymenu div.submenu{ /*DIV that contains each sub menu*/
178.background: white;
179.}
180.
```

```
181..glossymenu div.submenu ul{ /*UL of each sub menu*/
182.
183.list-style-type: none;
184.margin: 0;
185.padding: 0;
186.}
187.
188..glossymenu div.submenu ul li{
189.background-color: #EBEBE9;
190.border-bottom-style: none;
191.}
192.
193..glossymenu div.submenu ul li a{
194.display: block;
195.color: #333333;
196.text-decoration: none;
197.padding: 2px 0;
198.padding-left: 10px;
199.font-family: arial;
200.font-size: 10px;
201.font-style: normal;
202.font-weight: normal;
203.font-variant: normal;
204.}
205.
206..glossymenu div.submenu ul li a:hover{
207.background: #DFDCCB;
208.color: white;
209.}
210.
211..appcss{
212.font-family: Tahoma;
213.font-size: 12px;
214.color: #1C4482;
215.text-decoration: none;
216.}
217..mnu{
218.
219.padding: -20px;
220.margin-bottom: 0px;
221.width: 210px;
222.border-bottom-width: 0;
223.border-top-width: 1px;
224.border-right-width: 1px;
225.border-left-width: 1px;
226.border-top-style: solid;
227.border-right-style: solid;
228.border-bottom-style: solid;
229.border-left-style: solid;
```

```
230.border-top-color: #BEBEBE;
231.border-right-color: #BEBEBE;
232.border-bottom-color: #BEBEBE;
233.border-left-color: #BEBEBE;
234.
235.}
236.
237..mnu a{
238.color: #666666;
239.display: block;
240.position: relative; /*To help in the anchoring of the ".statusicon"
    icon image*/
241.width: auto;
242.padding: 4px 0;
243.margin-bottom: 2px;
244.margin-top: 0px;
245.padding-left: 10px;
246.text-decoration: none;
247.background-image: url(glossyback.png);
248.background-repeat: no-repeat;
249.background-color: #EDEDEB;
250.font-family: tahoma;
251.font-size: 9px;
252.}
253.
254..mnu a:hover{
255.color:#000066;
256.background-color: white;
257.text-decoration: none;
258.padding-bottom:2px;
259.padding-top:0px;
260.}
261..arrowlistmenu .menuheader a{
262.font-family:Tahoma;
263.text-decoration: none;
264.color: #FFFFFF;
265.
266.
267.}
268..arrowlistmenu .openheader{ /*CSS class to apply to expandable header
    when it's expanded*/
269.background-image: url(titlebar-active.png);
270.}
271.
272..arrowlistmenu ul{ /*CSS for UL of each sub menu*/
273.list-style-type: none;
274.margin: 10;
275.padding: 0;
```

```
276.margin-bottom: 8px; /*bottom spacing between each UL and rest of
    content*/
277.}
278.
279..arrowlistmenu ul li{
280.padding-bottom: 2px; /*bottom spacing between menu items*/
281.}
282.
283..arrowlistmenu ul li a{
284.color:#003399;
285.background: url(arrowbullet.png) no-repeat center left;
286.background-color:#C6C6C6;
287.display: block;
288.padding: 2px 0;
289.padding-left: 19px;
290.text-decoration: none;
291.font-weight: bold;
292.border-bottom: 1px solid #dadada;
293.font-size: 90%;
294.
295.}
296.
297..arrowlistmenu ul li a:visited{
298.color:#003366;
299.}
300.
301..arrowlistmenu ul li a:hover{ /*hover state CSS*/
302.color:#000066;
303.background-color: #C6C6C6;
304.}
305..tblss{
306.background-color:#DCDDDE;
307.border-right-width: medium;
308.border-left-width: medium;
309.border-right-style: solid;
310.border-left-style: solid;
311.border-right-color: #003366;
312.border-left-color: #003366;
313.}
314.
315..tblss td{
316.
317.
318.padding-left: 17px; /*link text is indented 19px*/
319.
320.}
321.#apDiv2 {
322.position:absolute;
323.width:100%;
```

```
324.height:40px;
325.z-index:0;
326.background-color:#9DB0BF;
327.}
328.#apDiv3 {
329.position:absolute;
330.width:200px;
331.height:115px;
332.z-index:2;
333.left: 5px;
334.top: 11px;
335.}
336.#apDiv4 {
337.position:absolute;
338.width:148px;
339.height:128px;
340.z-index:2;
341.left: 33px;
342.top: 21px;
343.}
344..style5 {
345.color: #FFFFFFF;
346.font-size: 24px;
347.font-family:Tahoma;
348.}
349.#apDiv5 {
350.position:absolute;
351.width:127px;
352.height:164px;
353.z-index:3;
354.left: 1216px;
355.top: -18px;
356.}
357..style7 {
358.font-size: 18px;
359.color: #FFFFFFF;
360.}
361..style8 {
362.font-size: 14px;
363.color: #FFFFFFF;
364.font-family: Calibri;
365.}
366.#apDiv6 {
367.position:absolute;
368.width:200px;
369.height:115px;
370.z-index:2;
371.left: 864px;
372.top: -24px;
```

```
373.}
374.#apDiv7 {
375.position:absolute;
376.width:115px;
377.height:139px;
378.z-index:2;
379.left: 935px;
380.top: 29px;
381.margin-top: 0px;
382.margin-left: 230px;
383.}
384.#apDiv8 {
385.position:absolute;
386.width:97px;
387.height:81px;
388.z-index:2;
389.top: 27px;
390.left: 16px;
391.}
392.#apDiv9 {
393.position:absolute;
394.width:80px;
395.height:90px;
396.z-index:3;
397.left: 91%;
398.top: 5px;
399.border-top-style: double;
400.border-right-style: double;
401.border-bottom-style: double;
402.border-left-style: double;
403.border-top-color: #000000;
404.border-right-color: #000000;
405.border-bottom-color: #000000;
406.border-left-color: #000000;
407.
408.}
409.
410..imgs{
411.border: thin double #000000;
412.
413.}
414..style9 {font-family: Tahoma}
415..style12 {
416.color: #13263E
417.}
418..trtbl{
419./*background:url(trbg.png);*/
420.
421.background-color:#DCDDDE;
```

```
422.
423.
424.}td img {display: block;}
425.td img {display: block;}
426.
427..tblbl{
428.background-image:url(st.png);
429.background-repeat: no-repeat;
430.}
431..style20 {color: #ACACAC}
432.-->
433. </style>
434.<br />
435.
436.<script language="javascript" src="jst.js"></script>
437.</head>
438.
439.<!-- Global IE fix to avoid layout crash when single word size wider
      than column width -->
440.<!--[if IE]><style type="text/css"> body {word-wrap: break-
      word;}</style><![endif]-->
441.
442.<body>
443.<div id="apDiv8"></div>
444.
445.<!-- Main Page Container -->
446.<div class="page-container">
447.
448. <!-- How to implement this header in your Multiflex-3 Layout:
      -->
449. <!-- 1. Copy the marked rows below
      -->
450. <!-- 2. Paste and replace marked rows in your "layoutNN.html" file
      -->
451. <!-- 3. Open CSS-file "header1_setup.css", and follow its
      instructions -->
452.
453. <!-- START COPY here -->
454.
455. <!-- A. HEADER -->
456. <div class="header">
457.
458. <!-- A.1 HEADER TOP -->
459. <div class="header-top">
460.
461. <!-- Sitelogo and sitename -->
462.
463.
464. <div id="apDiv1">
```

```

465.         <table width="100%" height="40" border="0" cellpadding="0"
            cellspacing="0" bgcolor="#DCE3E8">
466.             <tr>
467.                 <td width="78%"><div class="sitename"
                    id="tbl_schoolname">
468.                     <h1>Company Name</h1>
469.                 </div>&A           </td>
470.                 <td width="22%">&nbsp;</td>
471.             </tr>
472.         </table>
473.         <div id="apDiv2">
474.             <table width="100%" height="40" border="0"
                cellpadding="0" cellspacing="0">
475.                 <tr>
476.                     <td width="102%">&nbsp;</td>
477.                     <td width="703%"><span class="style5"
                        id="tbl_slogan">Culture of Continuous Improvement</span></td>
478.                     <td width="57%">&nbsp;</td>
479.                     <td width="111%"><p align="right"
                        class="style12">&nbsp;</p></td>
480.                 </tr>
481.                 <tr>
482.                     <td width="104%">&nbsp;</td>
483.                 </tr>
484.             </table>
485.         </div>
486.     </div>
487. </div>
488. <!-- A.2 HEADER MIDDLE -->
489. <div class="header-middle">
490.
491.     <!-- Site message -->
492.     <table width="100%" height="50" border="0" cellpadding="0"
        cellspacing="0">
493.         <tr>
494.             <td width="540%"></td>
495.             <td width="48%"><div align="right" class="style7"><span
                class="style9">Small/Medium Manufacturing Business Management
                Information System</span>&nbsp;</div>
496.                 <div align="right" class="style8">&quot;Truly &amp;
                Complete Data Management Solution&quot;&nbsp;</div></td>
497.             <td width="1%"></td>
498.             <td width="214%"><div align="right"></div></td>
499.         </tr>
500.     </table>
501. </div>
502.
503. <!-- A.3 HEADER BOTTOM -->

```



```

548.      <ul><a href="javascript:userT();" class="mxx">User</a></ul>
549.      <ul><a href="javascript:prod_pack();" class="mxx">Product
Package</a>
550.      </ul>
551.      </div>
552.      <a href="" class="menuitem">Production</a>
553.      <div id="ss1" style="display:block;">
554.      <ul><a href="javascript:raw_mat();" class="mxx">Manage Raw
Materials</a></ul>
555.      <ul><a href="javascript:stock();" class="mxx">Manage
Stock</a></ul>
556.      </div>
557.      <a href="" class="menuitem">Account</a>
558.      <div id="ss1" style="display:block;">
559.      <ul><a href="javascript:expenses();"
class="mxx">Expenses</a></ul>
560.      <ul><a href="javascript:sales();" class="mxx">Sales</a></ul>
561.      </div>
562.      <a href="" class="menuitem">Report</a>
563.      <div id="ss1" style="display:block;">
564.      <ul><a href="javascript:vstock();" class="mxx">Stock</a></ul>
565.      <ul><a href="javascript:sales_report();"
class="mxx">Sales</a></ul>
566.      <ul><a href="javascript:exp_report();"
class="mxx">Expenses</a></ul>
567.      <ul><a href="javascript:plos();" class="mxx">Profit/Loss
Analysis</a></ul>
568.      </div>
569.
570.
571.
572.      </td>
573.      <td width="3">&nbsp;</td>
574.      <td width="874" align="center">
575.      <table width="100%" border="0" cellspacing="0"
cellpadding="0">
576.      <tr>
577.      <td height="189" id="tbl_content"><label></label></td>
578.
579.      </tr>
580.</table>      </td>
581.      </tr>
582. </table>
583.<hr color="#FFFFFF"/>
584.      <table width="100%" height="117" border="0" cellpadding="0"
cellspacing="0" bgcolor="#3C617F">
585.      <tr>
586.      <td id="tbl_app"><br />

```

```
587.         <table width="100%" height="34" border="0" cellpadding="0"
           cellspacing="0">
588.             <tr>
589.                 <td width="48">&nbsp;</td>
590.                 <td width="84%">&nbsp;</td>
591.
592.                 <td width="51">&nbsp;</td>
593.             </tr>
594.         </table>           </td>
595.     </tr>
596. </table>
597.
598.     <hr color="#FFFFFF">
599.     <table width="100%" border="0" cellspacing="0" cellpadding="0"
           bgcolor="#3C617F">
600.         <tr>
601.             <td>&nbsp;</td>
602.         </tr>
603.     </table>
604. </div>
605. <script language="javascript">
606.
607. function loadmodule(id){
608. qr="select * from tbl_applicationmodule where
           ApplicationID='"+id+"'";
609. document.getElementById("tbl_mod").innerHTML="";
610. execut(qr,"ModuleName","ApplicationModuleID");
611. }
612.
613.
614. function swpa(){
615. swp("ssl");
616. }
617.
618. function prod_pack(){
619. if(isOn()){
620. loadpage("product_package.php","tbl_content");
621. }
622. else{
623. alert("You need to login to use this feature");
624. }
625. }
626.
627. function raw_mat(){
628. if(isOn()){
629. loadpage("raw_mat.php","tbl_content");
630. }
631. else{
632. alert("You need to login to use this feature");
```

```
633.}
634.}
635.
636.function stock(){
637.if(isOn()){
638.loadpage("stock.php","tbl_content");
639.}
640.else{
641.alert("You need to login to use this feature");
642.}
643.}
644.
645.function expenses(){
646.if(isOn()){
647.loadpage("expenses.php","tbl_content");
648.}else{
649.alert("You need to login to use this feature");
650.}
651.}
652.function sales(){
653.if(isOn()){
654.loadpage("sales.php","tbl_content");
655.}
656.else
657.{
658.alert("You need to login to use this feature");
659.}
660.}
661.
662.
663.function exp_report(){
664.if(isOn()){
665.if(isAdmin()){
666.loadpage("expenses_report.php","tbl_content");
667.}else{
668.alert("You do not have administrative right for this section");
669.}
670.}
671.else
672.{
673.alert("You need to login to use this feature");
674.}
675.}
676.
677.function vstock(){
678.if(isOn()){
679.if(isAdmin()){
680.loadpage("stock_report.php","tbl_content");
681.}else{
```

```
682.alert("You do not have administrative right for this section");
683.}
684.
685.}
686.else
687.{
688.alert("You need to login to use this feature");
689.}
690.}
691.
692.
693.function sales_report(){
694.if(isOn()){
695.if(isAdmin()){
696.loadpage("sales_report.php","tbl_content");
697.}else{
698.alert("You do not have administrative right for this section");
699.}
700.
701.}
702.else
703.{
704.alert("You need to login to use this feature");
705.}
706.}
707.
708.function plos(){
709.if(isOn()){
710.if(isAdmin()){
711.loadpage("plos.php","tbl_content");
712.}else{
713.alert("You do not have administrative right for this section");
714.}
715.}
716.else
717.{
718.alert("You need to login to use this feature");
719.}
720.}
721.
722.
723.function userT(){
724.if(isOn()){
725.if(isAdmin()){
726.loadpage("user.php","tbl_content");
727.}else{
728.alert("You do not have administrative right for this section");
729.}
730.}
```

```
731.else
732.{
733.alert("You need to login to use this feature");
734.}
735.}
736.
737.kok=getCookie("txtuser");
738.if(kok==""){
739.loadpage("login.php","tbl_content");
740.}
741.else{
742.loadpage("def.php","tbl_content");
743.}
744.
745.function toge(){
746.kok=getCookie("txtuser");
747.
748.if(kok==""){
749.loadpage("login.php","tbl_content");
750.}
751.else{
752.loadpage("def.php","tbl_content");
753.}
754.}
755.
756.function isOn(){
757.if(getCookie("txtuser")=="){
758.return false;
759.}
760.else{
761.return true;
762.}
763.
764.}
765.
766.function isAdmin(){
767.if(getCookie("roles")== "Admin"){
768.
769.return true;
770.}
771.else{
772.return false;
773.}
774.
775.}
776.</script>
777. </body>
778.</html>
```

2.Login.php

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<title>Login Page</title>
<style type="text/css">
<!--
#apDiv1 {
position:absolute;
width:528px;
height:262px;
z-index:1;
left: 212px;
top: 76px;
background-color: #D7D7D7;
border-top-width: medium;
border-right-width: medium;
border-bottom-width: thick;
border-left-width: medium;
border-top-style: solid;
border-right-style: solid;
border-bottom-style: solid;
border-left-style: solid;
border-top-color: #003366;
border-right-color: #333333;
border-bottom-color: #2F2F2F;
border-left-color: #003366;
}
.buton{
font-family: Tahoma;
font-size: 14px;
color: #FFFFFF;
background-color: #3399FF;
border-right-style: solid;
border-bottom-style: solid;
border-right-color: #2F2F2F;
border-bottom-color: #474747;
border-top-width: thin;
border-right-width: thin;
border-bottom-width: thin;
border-left-width: thin;
border-top-style: none;
}
.style1 {
font-family: Tahoma;
```



```

        </label></td>
    </tr>
    <tr>
        <td><span class="style1">Password</span></td>
        <td><input name="txtpass" type="password" id="txtpass"
/></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td><label></label></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td><label> <span class="style4">
            <input type="checkbox" name="checkbox" id="checkbox" />
            </span></label>
            <span class="style4"> Keep me sign in</span></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td><label>
            <input name="button" type="button" class="buton"
id="button" value="    Sign In    " onclick="logino();" />
            </label></td>
    </tr>
    <tr>
        <td>&nbsp;</td>
        <td>&nbsp;</td>
    </tr>
</table>
    </form>
</td>
</tr>
</table></td>
</tr>
</table>
</div>

<script language ="javascript">

var xmlHttp;
function getXMLHttpRequest() {
var xmlHttp=null;
try{// Firefox, Opera 8.0+, Safari

```

```
xmlHttp = new XMLHttpRequest();
}catch(e){
try{ // Internet Explorer
    xmlHttp=new ActiveXObject("Msxml2.XMLHTTP");
    }catch(e){
    xmlHttp=new ActiveXObject("Microsoft.XMLHTTP");
    }
}
return xmlHttp;
}

function logino(){
txtuser=document.getElementById("txtuser").value;
txtpass=document.getElementById("txtpass").value;
if(txtuser==""){
alert("Username is required");
return 1;
}

if(txtpass==""){
alert("Password is required");
return 1;
}

xmlHttp=getXMLHttpRequest();
if(xmlHttp==null){
alert("Your browser dose not support Ajax \n Try viewing this page with
Internet Explorer \n or browsers with Ajax functionality");
return;
}
url="log.php?" + "txtuser=" + txtuser + "&txtpass=" + txtpass;

    xmlHttp.onreadystatechange=function(){
if (xmlHttp.readyState==4){
var xx=xmlHttp.responseText;//get id txtBrand to change inner HTML

var xi=xx.split("|");
if(xi[0]=="0"){
alert("Error encountered");
alert(xi[1]);
return 1;
}
else{
alert(xi[1]);
window.location="def.php";
}
}
```

```

}else {
//document.getElementById(divi).innerHTML="<img width='100' height='80'
src='7hwy7at6.png.gif' />Loading Please wait...";
}
//else{
//document.getElementById(divi).innerHTML="<img width='100' height='80'
src='7hwy7at6.png.gif' />Loading Please wait...";
//}
}
xmlHttp.open("GET",url,true);
xmlHttp.send(null);

}

</script>
</body>
</html>

```

3List.php

```

<?php
include("conit.php");
if (!function_exists("GetSQLValueString")) {
function GetSQLValueString($theValue, $theType, $theDefinedValue = "",
$theNotDefinedValue = "")
{
    $theValue = get_magic_quotes_gpc() ? stripslashes($theValue) : $theValue;

    $theValue = function_exists("mysql_real_escape_string") ?
mysql_real_escape_string($theValue) : mysql_escape_string($theValue);

    switch ($theType) {
        case "text":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
        case "long":
        case "int":
            $theValue = ($theValue != "") ? intval($theValue) : "NULL";
            break;
        case "double":
            $theValue = ($theValue != "") ? "'" . doubleval($theValue) . "'" :
"NULL";
            break;
        case "date":
            $theValue = ($theValue != "") ? "'" . $theValue . "'" : "NULL";
            break;
    }
}
}

```

```

    case "defined":
        $theValue = ($theValue != "") ? $theDefinedValue :
$theNotDefinedValue;
        break;
    }
    return $theValue;
}
}

$currentPage = $_SERVER["PHP_SELF"];

$maxRows_Recordset1 = 5;
$pageNum_Recordset1 = 0;
if (isset($_GET['pageNum_Recordset1'])) {
    $pageNum_Recordset1 = $_GET['pageNum_Recordset1'];
}
$startRow_Recordset1 = $pageNum_Recordset1 * $maxRows_Recordset1;

mysql_select_db("eman",$con);
$query_Recordset1 = "SELECT * FROM leaks order by DDate DESC";
$query_limit_Recordset1 = sprintf("%s LIMIT %d, %d", $query_Recordset1,
$startRow_Recordset1, $maxRows_Recordset1);
$Recordset1 = mysql_query($query_limit_Recordset1, $con) or
die(mysql_error());
$row_Recordset1 = mysql_fetch_assoc($Recordset1);

if (isset($_GET['totalRows_Recordset1'])) {
    $totalRows_Recordset1 = $_GET['totalRows_Recordset1'];
} else {
    $all_Recordset1 = mysql_query($query_Recordset1);
    $totalRows_Recordset1 = mysql_num_rows($all_Recordset1);
}
$totalPages_Recordset1 = ceil($totalRows_Recordset1/$maxRows_Recordset1)-1;

$queryString_Recordset1 = "";
if (!empty($_SERVER['QUERY_STRING'])) {
    $params = explode("&", $_SERVER['QUERY_STRING']);
    $newParams = array();
    foreach ($params as $param) {
        if (strpos($param, "pageNum_Recordset1") == false &&
            strpos($param, "totalRows_Recordset1") == false) {
            array_push($newParams, $param);
        }
    }
}
if (count($newParams) != 0) {
    $queryString_Recordset1 = "&" . htmlentities(implode("&", $newParams));
}
}

```

```
$queryString_Recordset1 = sprintf("&totalRows_Recordset1=%d%s",
$totalRows_Recordset1, $queryString_Recordset1);
```

```
$qry1="select * from tbl_pack";
//$qry2="select * from tbl_applicationtask";
$res1=mysql_query($qry1);
```

```
$rd1=mysql_fetch_assoc($res1);
```

```
function getit($qr,$dt){
$rt=mysql_query($qr);
$rd=mysql_fetch_assoc($rt);
return $rd[$dt];
}
```

```
?>
```

```
<!DOCTYPE html PUBLIC "-//W3C//DTD XHTML 1.0 Transitional//EN"
"http://www.w3.org/TR/xhtml1/DTD/xhtml1-transitional.dtd">
<html xmlns="http://www.w3.org/1999/xhtml">
<head>
<title>Untitled Document</title>
<meta http-equiv="Content-Type" content="text/html; charset=utf-8" />
<script language="javascript" type="text/javascript"
src="niceforms.js"></script>
<script language="javascript" type="text/javascript"
src="xmlwrap.js"></script>
<link rel="stylesheet" type="text/css" media="all" href="niceforms-
default.css" />

<link rel="stylesheet" type="text/css" href="codebase/dhtmlxgrid.css">
<link rel="stylesheet" type="text/css"
href="codebase/skins/dhtmlxgrid_dhx_skyblue.css">
<link rel="stylesheet" type="text/css"
href="dhtmlxCalendar/codebase/dhtmlxcalendar.css">
<script src="codebase/dhtmlxcommon.js"></script>
<script src="codebase/dhtmlxgrid.js"></script>
<script src="codebase/dhtmlxgridcell.js"></script>
<link rel="stylesheet" href="Css/Grid.css">
<script src="dhtmlxCalendar/codebase/dhtmlxcalendar.js"></script>
<script src="codebase/excells/dhtmlxgrid_excell_dhxcalendar.js"></script>
<script type="text/javascript" src="calen/mootools.js"></script>
<script type="text/javascript" src="calen/calendar.js"></script>
```

```

<script language="JavaScript" src="tigra_calendar/calendar_us.js"></script>
<link rel="stylesheet" href="tigra_calendar/calendar.css">
<script type="text/javascript" src="func2.js"></script>
<link rel="stylesheet" type="text/css" href="calen/calendar.css"
media="screen" />
<link rel="stylesheet" type="text/css" href="calen/dashboar.css"
media="screen" />
<link rel="stylesheet" type="text/css" href="i-heart-.css" media="screen" />
<link rel="STYLESHEET" type="text/css"
href="dhtmlxCalendar/codebase/skins/dhtmlxcalendar_yahoolike.css">

```

```

<style type="text/css">
<!--
.style1 {
color: #000000;
font-weight: bold;
font-size: 14px;
font-family: Georgia, "Times New Roman", Times, serif;
}
.style2 {font-size: 10px}
-->
</style>

```

</head>

```

<body>
<div id="container">

```

```

<form method="post" enctype="multipart/form-data" name="testForm"
class="niceform" id="testForm" action="save_cou.php">
<fieldset>
<legend> Manage Leakages</legend>

```

```

<dl>
<dt>
<label for="txtprod">Production No.:</label>
</dt>
<dd>
<input type="text" name="txtprod" id="txtprod" size="33"
maxlength="128" alt="Supply product number, click on get details to load
production" />
<label>
<input type="button" name="button" id="button" value="Get
details" onclick="getproda();" />
</label>
</dd>
</dl>
<dl>

```

```

<dt>
  <label for="txtprod">Product:</label>
</dt>
  <dd>
    <div id="txtpack">

      </div>
    </dd>
</dl>

<dl>
<dt>
  <label for="txtprod">Date Produced:</label>
</dt>
  <dd>
    <div id="date1">

      </div>
    </dd>
</dl>
  <dl>
  <dt>
    <label for="txtprod">Date:</label>
  </dt>
  <dd>
    <input type="text" name="date2" id="date2" size="33"
maxlength="128" /><script language="JavaScript">
new tcal ({
// form name
'formname': 'testForm',
// input name
'controlname': 'date2'
});
</script>
    </dd>
  </dl>
<dl>
  <dt>
    <label for="password">Leakage Quantity:</label>
  </dt>
  <dd>
    <label></label>
    <input type="text" name="txtq" id="txtq" size="33"
maxlength="128"/>
    (Pieces)</dd>
</dl>

```

```

</fieldset>

<fieldset class="action">
<table width="589" border="0" cellspacing="0" cellpadding="0">
  <tr>
    <td width="66">&nbsp;</td>
    <td width="183"><label></label></td>
    <td width="168"><input name="Button" type="button" value="      Save
" onclick="addleaks();" /></td>
    <td width="97"><label></label></td>
    <td width="75"><label></label></td>
  </tr>
</table>
</fieldset>

</form>

<fieldset>
<table width="100%" border="1">
  <tr>
    <td width="4%" class="Gridheader">S/N</td>
    <td width="17%" class="Gridheader"><div
align="center">Product</div></td>
    <td width="28%" class="Gridheader"><div align="center">Date
Returned</div></td>
    <td width="18%" class="Gridheader"><div align="center">Leakages
(Qty)</div></td>
    <td width="17%" class="Gridheader"><div align="center">Production
No.</div></td>

    <td width="8%" class="Gridheader"><div align="center">Delete</div></td>
  </tr>
  <?php
  $i=0;
  do {
  $i=$i+1;?>
  <tr>
    <td class="Gridheader"><div align="center"><?php echo $i; ?></div></td>
    <td class="GridDataB"><div align="left">&nbsp;<
    <?php
  $id=$row_Recordset1['LeakID'];
  echo $row_Recordset1['Product']; ?>
    </div></td>
    <td class="GridDataB"><div align="left">&nbsp;<?php echo
$row_Recordset1['DDate']; ?></div></td>
    <td class="GridDataB"><div align="left">&nbsp;<?php echo
$row_Recordset1['Quantity']; ?></div></td>

```

```

<td class="GridDataB"><div align="left">&nbsp;<?php echo
$row_Recordset1['PNo']; ?></div></td>

<td class="Gridheader"><div align="center"><a href="<?php echo
"delete_leak?id=$id" ?>"></a></div></td>
</tr>
<?php } while ($row_Recordset1 = mysql_fetch_assoc($Recordset1)); ?>
</table>
<table width="100%" border="0">
<tr>
<td width="139" height="44" class="Gridheader">&nbsp;</td>
<td width="132" class="Gridheader"><?php if ($pageNum_Recordset1 > 0) {
// Show if not first page ?>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage, 0,
$queryString_Recordset1); ?>"></a>
<?php } // Show if not first page ?>
</td>

<td width="121" class="Gridheader"><?php if ($pageNum_Recordset1 > 0) {
// Show if not first page ?>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
max(0, $pageNum_Recordset1 - 1), $queryString_Recordset1); ?>"></a>
<?php } // Show if not first page ?> </td>
<td width="104" class="Gridheader"><?php if ($pageNum_Recordset1 <
$totalPages_Recordset1) { // Show if not last page ?>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
min($totalPages_Recordset1, $pageNum_Recordset1 + 1),
$queryString_Recordset1); ?>"></a>
<?php } // Show if not last page ?> </td>
<td width="182" class="Gridheader"><?php if ($pageNum_Recordset1 <
$totalPages_Recordset1) { // Show if not last page ?>
<a href="<?php printf("%s?pageNum_Recordset1=%d%s", $currentPage,
$totalPages_Recordset1, $queryString_Recordset1); ?>"></a>
<?php } // Show if not last page ?> </td>

</tr></table></fieldset>

<script language="javascript">
</script>
</div></body>
</html>

```

FEASIBILITY STUDY:

Feasibility study is conducted once the problem is clearly understood. Feasibility study is a high level capsule version of the entire system analysis and design process. The objective is to determine quickly at a minimum expense how to solve a problem. The purpose of feasibility is not to solve the problem but to determine if the problem is worth solving.

The system has been tested for feasibility in the following points.

1. Technical Feasibility
2. Economical Feasibility
3. Operational Feasibility.

1. Technical Feasibility

The project entitles "Courier Service System" is technically feasibility because of the below mentioned feature. The project was developed in Java which Graphical User Interface.

It provides the high level of reliability, availability and compatibility. All these make Java an appropriate language for this project. Thus the existing software Java is a powerful language.

2. Economical Feasibility

The computerized system will help in automate the selection leading the profits and details of the organization. With this software, the machine and manpower utilization are expected to go up by 80-90% approximately. The costs incurred of not creating the system are set to be great, because precious time can be wanted by manually.

3. Operational Feasibility

In this project, the management will know the details of each project where he may be presented and the data will be maintained as decentralized and if any inquires for that particular contract can be known as per their requirements and necessities.

Implementation:

Implementation is the stage where the theoretical design is turned into a working system. The most crucial stage in achieving a new successful system and in giving confidence on the new system for the users that it will work efficiently and effectively.

The system can be implemented only after thorough testing is done and if it is found to work according to the specification.

It involves careful planning, investigation of the current system and its constraints on implementation, design of methods to achieve the change over and an evaluation of change over methods a part from planning. Two major tasks of preparing the implementation are education and training of the users and testing of the system.

The more complex the system being implemented, the more involved will be the systems analysis and design effort required just for implementation.

The implementation phase comprises of several activities. The required hardware and software acquisition is carried out. The system may require some software to be developed. For this, programs are written and tested. The user then changes over to his new fully tested system and the old system is discontinued.

TESTING:

The testing phase is an important part of software development. It is the computerized system will help in automate process of finding errors and missing operations and also a complete verification to determine whether the objectives are met and the user requirements are satisfied.

Software testing is carried out in three steps:

1. The first includes unit testing, where in each module is tested to provide its correctness, validity and also determine any missing operations and to verify whether the objectives have been met. Errors are noted down and corrected immediately. Unit testing is the important and major part of the project. So errors are rectified easily in particular module and program clarity is increased. In this project entire system is divided into several modules and is developed individually. So unit testing is conducted to individual modules.

2. The second step includes Integration testing. It need not be the case, the software whose modules when run individually and showing perfect results, will also show perfect results when run as a whole. The individual modules are clipped under this major module and tested again and verified the results. This is due to poor interfacing, which may results in data being lost across an interface. A module can have inadvertent, adverse effect on any other or on the global data structures, causing serious problems.

3. The final step involves validation and testing which determines which the software functions as the user expected. Here also some modifications were. In the completion of the project it is satisfied fully by the end user.

Maintenance and environment:

AS the number of computer based systems, grievance libraries of computer software began to expand. In house developed projects produced tones of thousand soft program source statements. Software products purchased from the outside added hundreds of thousands of new statements. A dark cloud appeared on the horizon. All of these programs, all of those source statements-had to be corrected when false were detected, modified as user requirements changed, or adapted to new hardware that was purchased. These activities were collectively called software Maintenance.

The maintenance phase focuses on change that is associated with error correction, adaptations required as the software's environment evolves, and changes due to enhancements brought about by changing customer requirements. Four types of changes are encountered during the maintenance phase.

Correction
Adaptation
Enhancement
Prevention

Correction:

Even with the best quality assurance activities is lightly that the customer will uncover defects in the software. Corrective maintenance changes the software to correct defects.

Maintenance is a set of software Engineering activities that occur after software has been delivered to the customer and put into operation. Software configuration management is a set of tracking and control activities that began when a software project begins and terminates only when the software is taken out of the operation.

We may define maintenance by describing four activities that are undertaken after a program is released for use:

Corrective Maintenance
Adaptive Maintenance
Perfective Maintenance or Enhancement
Preventive Maintenance or reengineering

Only about 20 percent of all maintenance work are spent "fixing mistakes". The remaining 80

percent are spent adapting existing systems to changes in their external environment, making enhancements requested by users, and reengineering an application for use.

ADAPTATION:

Over time, the original environment (E>G., CPU, operating system, business rules, external product characteristics) for which the software was developed is likely to change. Adaptive maintenance results in modification to the software to accommodate change to its external environment.

ENHANCEMENT:

As software is used, the customer/user will recognize additional functions that will provide benefit. Perceptive maintenance extends the software beyond its original function requirements.

PREVENTION :

Computer software deteriorates due to change, and because of this, preventive maintenance, often called software re engineering, must be conducted to enable the software to serve the needs of its end users. In essence, preventive maintenance makes changes to computer programs so that they can be more easily corrected, adapted, and enhanced. Software configuration management (SCM) is an umbrella activity that is applied throughout the software process. SCM activities are developed to

SOFTWARE METHODOLOGY

The software methodology followed in this project includes the object-oriented methodology and the application system development methodologies. The description of these methodologies is given below.

Application System Development – A Life cycle Approach

Although there are a growing number of applications (such as decision support systems) that should be developed using an experimental process strategy such as prototyping, a significant amount of new development work continue to involve major operational applications of broad scope. The application systems are large highly structured. User task comprehension and developer task proficiency is usually high. These factors suggest a linear or iterative assurance strategy. The most common method for this stage class of problems is a system development life cycle modal in which each stage of development is well defined and has straightforward requirements for deliverables, feedback and sign off. The system development life cycle is described in detail since it continues to be an appropriate methodology for a significant part of new development work.

The basic idea of the system development life cycle is that there is a well-defined process by which an application is conceived and developed and implemented. The life cycle gives structure to a creative process. In order to manage and control the development effort, it is necessary to know what should have been done, what has been done, and what has yet to be accomplished. The phrases in the system development life cycle provide a basis for management and control because they define segments of the

flow of work, which can be identified for managerial purposes and specifies the documents or other deliverables to be produced in each phase.

The phases in the life cycle for information system development are described differently by different writers, but the differences are primarily in the amount of necessity and manner of categorization. There is a general agreement on the flow of development steps and the necessity for control procedures at each stage.

The information system development cycle for an application consists of three major stages.

- 1) Definition.
- 2) Development.
- 3) Installation and operation.

The first stage of the process, which defines the information requirements for a feasible cost effective system. The requirements are then translated into a physical system of forms, procedures, programs etc., by the system design, computer programming and procedure development. The resulting system is test and put into operation. No system is perfect so there is always a need for maintenance changes. To complete the cycle, there should be a post audit of the system to evaluate how well it performs and how well it meets the cost and performance specifications. The stages of definition, development and installation and operation can therefore be divided into smaller steps or phrases as follows.

Definition

Proposed definition : preparation of request for proposed applications.

Feasibility assessment : evaluation of feasibility and cost benefit of proposed system.

Information requirement analysis : determination of information needed.

Design

Conceptual design : User-oriented design of application development.

Physical system design : Detailed design of flows and processes in applications processing system and preparation of program specification.

Development

Program development : coding and testing of computer programs.

Procedure development : design of procedures and preparation of user instructions.

Installation and operation

Conversion : final system test and conversion.

Operation and maintenance : Month to month operation and maintenance

Post audit : Evaluation of development process,application system and results of use at the completion of the each phase, formal approval sign-off is required from the users as well as from the manager of the project development.

Testing

Testing is a process of executing a program with the indent of finding an error. Testing is a crucial element of software quality assurance and presents ultimate review of specification, design and coding.

System Testing is an important phase. Testing represents an interesting anomaly for the software. Thus a series of testing are performed for the proposed system before the system is ready for user acceptance testing.

A good test case is one that has a high probability of finding an as undiscovered error. A successful test is one that uncovers an as undiscovered error.

Testing Objectives:

1. Testing is a process of executing a program with the intent of finding an error
2. A good test case is one that has a probability of finding an as yet undiscovered error
3. A successful test is one that uncovers an undiscovered error

Testing Principles

1. All tests should be traceable to end user requirements
2. Tests should be planned long before testing begins
3. Testing should begin on a small scale and progress towards testing in large
4. Exhaustive testing is not possible
5. To be most effective testing should be conducted by an independent third party

The primary objective for test case design is to derive a set of tests that has the highest likelihood for uncovering defects in software. To accomplish this objective two different categories of test case design techniques are used. They are

White box testing.

Black box testing.

White-box testing:

White box testing focus on the program control structure. Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Block-box testing:

Black box testing is designed to validate functional requirements without regard to the internal workings of a program. Black box testing mainly focuses on the information domain of the software, deriving test cases by partitioning input and output in a manner that provides through test coverage. Incorrect and missing functions, interface errors, errors in data structures, error in functional logic are the errors falling in this category.

Testing strategies:

A strategy for software testing must accommodate low-level tests that are necessary to verify that all small source code segment has been correctly implemented as well as high-level tests that validate major system functions against customer requirements.

Testing fundamentals:

Testing is a process of executing program with the intent of finding error. A good test case is one that has high probability of finding an undiscovered error. If testing is conducted successfully it uncovers the errors in the software. Testing cannot show the absence of defects, it can only show that software defects present.

Testing Information flow:

Information flow for testing flows the pattern. Two class of input provided to test the process. The software configuration includes a software requirements specification, a design specification and source code.

Test configuration includes test plan and test cases and test tools. Tests are conducted and all the results are evaluated. That is test results are compared with expected results. When erroneous data are uncovered, an error is implied and debugging commences.

Unit testing:

Unit testing is essential for the verification of the code produced during the coding phase and hence the goal is to test the internal logic of the modules. Using the detailed design description as a guide, important paths are tested to uncover errors within the boundary of the modules. These tests were carried out during the programming stage itself. All units of ViennaSQL were successfully tested.

Integration testing :

Integration testing focuses on unit tested modules and build the program structure that is dictated by the design phase.

System testing:

System testing tests the integration of each module in the system. It also tests to find discrepancies between the system and its original objective, current specification and system documentation. The primary concern is the compatibility of individual modules. Entire system is working properly or not will be tested here, and specified path ODBC connection will correct or not, and giving output or not are tested here these verifications and validations are done by giving input values to the system and by comparing with expected output. Top-down testing implementing here.

Acceptance Testing:

This testing is done to verify the readiness of the system for the implementation. Acceptance testing begins when the system is complete. Its purpose is to provide the end user with the confidence that the system is ready for use. It involves planning and execution of functional tests, performance tests and stress tests in order to demonstrate that the implemented system satisfies its requirements.

Tools to special importance during acceptance testing include:

Test coverage Analyzer – records the control paths followed for each test case.

Timing Analyzer – also called a profiler, reports the time spent in various regions of the code are areas to concentrate on to improve system performance.

Coding standards – static analyzers and standard checkers are used to inspect code for deviations from standards and guidelines.

Test Cases:

Test cases are derived to ensure that all statements in the program have been executed at least once during testing and that all logical conditions have been executed.

Using White-Box testing methods, the software engineer can drive test cases that

- Guarantee that logical decisions on their true and false sides.
- Exercise all logical decisions on their true and false sides.
- Execute all loops at their boundaries and with in their operational bounds.
- Exercise internal data structure to assure their validity.

The test case specification for system testing has to be submitted for review before system testing commences.

Output screens

Business Management Information System

The screenshot shows a web browser window with the URL `localhost/eman/`. The page header includes the text "Company Name" and "Culture of Continuous Improvement". Below this, a navigation bar contains "Small/Medium Manufacturing Business Management Information System" and "Ajax Powered". A secondary navigation bar includes "Log Out" and "Home" links. On the left side, there is a vertical menu with categories: "Setup" (User, Product Package), "Production" (Manage Raw Materials, Manage Stock), "Account" (Expenses, Sales), and "Report" (Stock, Sales, Expenses, Profit/Loss Analysis). The main content area is dominated by a "User Authentication" dialog box with the following elements:

- Title: "User Authentication"
- Instruction: "Supply Username and password and click on login"
- Fields: "Username" and "Password" (both with input boxes)
- Checkbox: "Keep me sign in" (unchecked)
- Button: "Sign In" (blue)

The Windows taskbar at the bottom shows several open files (7.jpg, 6.jpg, 4.jpg, 3.jpg, 2.png, 2.png) and the system tray displays the time as 2:38 PM on 5/13/2018.

Home page

Business Management Information System

The screenshot shows a web browser window displaying a Business Management Information System. The browser's address bar shows the URL `localhost/eman/`. The application header includes the text "Company Name" and "Culture of Continuous Improvement". Below this, it identifies itself as a "Small/Medium Manufacturing Business Management Information System" and "Truly & Complete Data Management Solution", with a badge indicating it is "Ajax Powered".

The main content area is divided into several sections:

- Notifications:** A section showing a notification for "Today 05/13/2018" with "Current User: anshuman".
- Stock Alert:** A section titled "Yesterday 05/12/2018 Transaction summary" containing a table with one row: "Bottle Water 5 ltrs" with a "Remaining Amount:0". A "View Stock" button is located to the right of this row.
- Shortcuts:** A grid of shortcuts for various functions: "Leakages", "Product Packages", "Manage Raw Materials", "Sales", "Expenses", and "Manage Stock".

A left-hand navigation menu lists various system functions under categories like "Setup", "Production", "Account", and "Report". The Windows taskbar at the bottom shows the system clock as 2:38 PM on 5/13/2018.

User Registration

Company Name
Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Ajax Powered

Log Out Home

Setup
User
Product Package
Production
Manage Raw Materials
Manage Stock
Account
Expenses
Sales
Report
Stock
Sales
Expenses
Profit/Loss Analysis

Deprecated: mysql_connect(): The mysql extension is deprecated and will be removed in the future: use mysqli or PDO instead in C:\wamp2\www\leman\conit.php on line 3

#	Time	Memory	Function	Location
1	0.0580	280856	{main}()	...\user.php:0
2	0.0582	282720	include('C:\wamp2\www\leman\conit.php')	...\user.php:2
3	0.0582	283024	mysql_connect()	...\conit.php:3

Manage User

Username:

Password:

Confirm Password:

Name:

Telephone:

Role: User

7.jpg 6.jpg 4.jpg 3.jpg 2.png 2.png Show all

Search the web and Windows 2:39 PM 5/13/2018

Entering Registration values

Company Name

Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Ajax Powered

Log Out Home

Setup
User
Product Package

Production
Manage Raw Materials
Manage Stock

Account
Expenses
Sales

Report
Stock
Sales
Expenses
Profit/Loss Analysis

Deprecated: mysql_connect(): The mysql extension is deprecated and will be removed in the future: use mysqli or PDO instead in C:\wamp2\www\eman\conit.php on line 3

Call Stack

#	Time	Memory	Function	Location
1	0.0151	276752	{main}()	...\product_package.php:0
2	0.0155	278616	include('C:\wamp2\www\eman\conit.php')	...\product_package.php:2
3	0.0155	278920	mysql_connect()	...\conit.php:3

Product Package

Package Name:

Pieces in Package:

Price/Piece:

Price/Package:

ADD

S/N	Package Name	Piece(s) Per pack	Price Per piece	Price Per pack	Delete	Edit
1	Bottle Water 5 ltrs	12	100	1200		

Company Name

Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Ajax Powered

Log Out Home

Setup
User
Product Package

Production
Manage Raw Materials
Manage Stock

Account
Expenses
Sales

Report
Stock
Sales
Expenses
Profit/Loss Analysis

Call Stack

#	Time	Memory	Function	Location
1	0.0185	276080	{main}()	...raw_mat.php:0
2	0.0189	277944	include('C:\wamp2\www\eman\conit.php')	...raw_mat.php:2
3	0.0189	278248	mysql_connect()	...conit.php:3

Manage Raw Materials

Reference No.:

Type: Polythene ▾

Quantity:

Litres ▾

Amount:

Date Purchased:

7.jpg 6.jpg 4.jpg 3.jpg 2.png 2.png Show all X

Search the web and Windows

2:40 PM
5/13/2018

Business Management Information System

Company Name

Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Ajax Powered

Log Out Home

Setup
User
Product Package

Production
Manage Raw Materials
Manage Stock

Account
Expenses
Sales

Report
Stock
Sales
Expenses
Profit/Loss Analysis

Notifications

Today 05/13/2018 Current User: anshuman

Stock Alert

Yesterday 05/12/2018 Transaction summary

Bottle Water 5 ltrs	Remaining Amount:0	View Stock
---------------------	--------------------	----------------------------

Shortcuts

Leakages	Product Packages	Manage Raw Materials
Sales	Expenses	Manage Stock

7.jpg 6.jpg 4.jpg 3.jpg 2.png 2.png Show all X

Search the web and Windows

2:41 PM
5/13/2018

Company Name
Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Log Out Home

Setup
User
Product Package
Production
Manage Raw Materials
Manage Stock
Account
Expenses
Sales
Report
Stock
Sales
Expenses
Profit/Loss Analysis

Deprecated: mysql_connect(): The mysql extension is deprecated and will be removed in the future: use mysqli or PDO instead in C:\wamp2\www\leman\conit.php on line 3

#	Time	Memory	Function	Location
1	0.0223	244400	{main}()	...\plos.php:0
2	0.0229	246304	include('C:\wamp2\www\leman\conit.php')	...\plos.php:2
3	0.0229	246608	mysql_connect()	...\conit.php:3

Profit and Loss Analysis

From date:

To date:

RUN ANALYSIS

7.jpg 6.jpg 4.jpg 3.jpg 2.png 2.png Show all

Search the web and Windows 2:43 PM 5/13/2018

Company Name
Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Log Out Home

Setup
User
Product Package
Production
Manage Raw Materials
Manage Stock
Account
Expenses
Sales
Report
Stock
Sales
Expenses
Profit/Loss Analysis

Deprecated: mysql_connect(): The mysql extension is deprecated and will be removed in the future: use mysqli or PDO instead in C:\wamp2\www\eman\conit.php on line 3

#	Time	Memory	Function	Location
1	0.1120	248248	{main}()	...sales_report.php:0
2	0.1129	250120	include('C:\wamp2\www\eman\conit.php')	...sales_report.php:2
3	0.1129	250424	mysql_connect()	...conit.php:3

Sales Report based on Date only

From date:

To date:

VIEW REPORT BASED ON DATE RANGE

Sales report based on Product and Date

Product: Bottle Water 5 ltrs

From:

7.jpg 6.jpg 4.jpg 3.jpg 2.png 2.png Show all X

Search the web and Windows

2:42 PM 5/13/2018

Company Name
Culture of Continuous Improvement

Small/Medium Manufacturing Business Management Information System
"Truly & Complete Data Management Solution"

Ajax Powered

Log Out Home

Setup
User
Product Package
Production
Manage Raw Materials
Manage Stock
Account
Expenses
Sales
Report
Stock
Sales
Expenses
Profit/Loss Analysis

Deprecated: mysql_connect(): The mysql extension is deprecated and will be removed in the future: use mysqli or PDO instead in C:\wamp2\www\leman\conit.php on line 3

#	Time	Memory	Function	Location
1	0.0417	277752	{main}()	...\expenses.php:0
2	0.0429	279616	include('C:\wamp2\www\leman\conit.php')	...\expenses.php:2
3	0.0430	279920	mysql_connect()	...\conit.php:3

Manage Expenses

Ref. No:

Description:

Date:

Reason:

Cost:

Remark:

Conclusion

CONCLUSION

CONCLUSION:

The package was designed in such a way that future modifications can be done easily. The following conclusions can be deduced from the development of the project.

- Automation of the entire system improves the efficiency
- It provides a friendly graphical user interface which proves to be better when compared to the existing system.
- It gives appropriate access to the authorized users depending on their permissions.
- It effectively overcomes the delay in communications.
- Updating of information becomes so easier.
- System security, data security and reliability are the striking features.
- The System has adequate scope for modification in future if it is necessary.

Future Enhancements

Project Management as Technology

As with most other business processes, the future of project management is driven by the following disruptive technologies:

Cloud

Cloud based solutions will feature more robust tools and higher computing power that will empower line managers to make fast decisions on the ground and take more responsibility to deliver results faster and with less chances of mistakes. Meantime, project leaders can focus on optimizing the project and spend less time on controlling teams.

SaaS project management software allows project and team leaders to share information and collaborate anywhere and anytime there's an Internet connection. Real time data is shared, annotated, approved, edited, read, translated, documented on the spot, etc. These features are provided by many vendors now; expect the best to come soon.

Mobile

More people will own smartphones and tablets, which means more companies adopting Bring your own device (BYOD) to access company systems including project management software. Using employee's personal devices looks like cutting on hardware costs, but businesses may have to pay more on software add-ons for complex security and deployment to ensure the integrity of the project plan across different platforms and devices owned by different people and interconnected to different social networks.

It's a double-sided sword; allowing employees to access sensitive project data to make them mobile and more productive, while exposing confidential information on two fronts: the vendor's server and employee's personal device (phone theft is a real issue here). The future of project management software will focus on sophisticated encryption and tighter user access tiers.

Social

We already have vendors integrating social network plugins in their project management software, more so will follow. Social channels like LinkedIn, Twitter, Google+, and Facebook have groupware tools that can easily match the most powerful enterprise collaboration solution. Not only will project management software integrate with these platforms, but some will be built on them targeting small businesses with teams in multiple locations. In some cases, we'll see a reversal in roles: some software will be merely sold or given free as plug-ins to these powerful social channels.

Read more: [Market Trends for Project Management Software](#)

Big data

What to do with the huge, seemingly disparate chunk of information customers are sharing online? From GPS locations to personal data, political leanings to dietary preferences, customers inundate their social networks with data that when taken as a whole can produce business insights and market trends. In the meantime that marketing is scouring big data for leads and campaigns, project leaders and stakeholders will use big data to ensure their goals

are aligned to the right trends. As a result, either more project management software will develop a business insight module or a middleware add-on to integrate to a CRM system with a business insights tool.

Project Management as Human Resource

Beyond the technology lies the project leader, the person responsible for its success or failure. Not only does technology help to develop better project leaders, but it's changing their roles and a few traits. Let's compare the future of project management roles:

Democratic

Project plan is more than ever shared across team members and collaboration is on high gear, allowing the democratization of data and access to business information. Where before the project leader has the sole influence in decisions, today's leaders have to lend an ear to credible ideas and suggestions coming from the ground: the team leaders' inputs. As a result, we'll see more ideas and shared goals between stakeholders, senior management, project leaders, team leaders, and members.

FUTURE ENHANCEMENTS:

This application avoids the manual work and the problems concern with it. It is an easy way to obtain the information regarding the different scheduled examinations information that are Currently issued.

Well I and my team members have worked hard in order to present an improved website better than the existing one's regarding the information about the various activities. Still ,we found out that the project can be done in a better way. Primarily, when we request information about a particular schedules it just shows the exam date and platform. So, after getting the information we can get access to the onlineexam.

The enhancement that we can add the searching option. We can directly search to the particular student details from this site.

Bibliograph

y

BIBLIOGRAPHY

The following books were referred during the analysis and execution phase of the project

✓ **Books Referred:**

- BEGINNING PHP 5 ---DAVE MERCER
- BLACK BOOK HTML ---WILEY DREAMTECH
- PHP AND MYSQL WEB DEVELOPMENT --- LUKEWELLING,LAURA
- MICROSOFT SQL SERVER-2000 ---RANKIN, PAUL & JENSEN
- SQL SERVER-2000 ---DUSAN PETKOVIC
- PHP IN A NUTSHELL --- PAUL HUDSON

✓ **Websites Referred:**

<http://www.phptpoit.com>

