## Disclaimer

This scenario breakdown is a fictionalized, illustrative case study created for educational and strategic thinking purposes. While inspired by real-world patterns and organizational challenges, all details—company context, team structure, and suggested approaches—are generalized and do not represent any specific employer, client, or confidential situation.

The content is designed to demonstrate strategic problem-solving, not to prescribe one-size-fits-all solutions. Readers are encouraged to adapt ideas and frameworks to suit their unique organizational needs, capabilities, and compliance contexts.

# **Context: The Situation**

You're supporting a global enterprise where various departments have independently started experimenting with AI — from chatbot pilots in customer support to small recommendation engines in product. However, there's no unifying framework. Each team uses different tools, models, and vendors. Some rely on consultants, others have built internal scripts, and many don't share what they're learning.

The org is now planning a broader AI rollout. Leadership realizes there's potential but also risk: lack of reuse, inconsistent standards, and ethical oversights. They want to move from scattered initiatives to coordinated capability — without stifling team ownership.

#### **Root Problems**

- **Siloed Experimentation**: No visibility across teams, leading to redundant or conflicting efforts.
- Tool and Stack Fragmentation: Different clouds, libraries, and infra in use.
- No Capability Model: Hard to assess maturity or plan org-wide uplift.
- **Inconsistent Guardrails**: Data handling, explainability, and privacy policies vary.
- Knowledge Drain: Experiments live in notebooks, not shared or documented.

# Approach & Framework

# Apply the AI Capability Uplift Canvas: Awareness → Foundation → Application → Productization

# Phase 1: Awareness

- Run Al literacy campaigns (videos, case demos, brown bags)
- Appoint "Al Anchors" in each team part coach, part coordinator
- Use storytelling to surface grassroots experiments

# Phase 2: Foundation

- Create a lightweight Capability Tracker by team
- Standardize access: GPU sandboxes, prompt libraries, approved APIs
- Publish guidance docs (model usage, logging, risks)

# Phase 3: Application

- Encourage inter-team show-and-tell forums
- Launch 2–3 cross-functional hackathons
- Introduce tagging and review rituals for AI repos (e.g., "reviewed by ethics", "can reuse")

# Phase 4: Productization

- Define go/no-go playbooks: What qualifies a prototype to graduate
- Share success metrics: adoption, reuse, quality
- Build team heatmaps and capability dashboards

# 30-60-90 Day Execution Plan (The Core Blueprint)

# Days 0–30: AWARENESS + MAPPING

- Launch "AI Stories from the Ground" series on internal Slack/newsletter
- Conduct a team-level capability survey (skills, infra, pilots)
- Host 3 AI literacy sessions: "What is Prompt Engineering", "How a Vector DB Works", "AI Ethics in Practice"
- Appoint AI Anchors in each major department
- Deliverables: Al Anchor Roster, Capability Tracker v1, awareness participation stats

## Days 31–60: FOUNDATION + ALIGNMENT

- Provision access to shared sandboxes, approved APIs (OpenAI, HuggingFace, internal models)
- Launch internal office hours for prompt tuning, LLM usage
- Publish Model Usage Handbook v1 (approved models, data zones, retry policies)
- Run team demo sessions 1 per function
- Deliverables: Handbook v1, API Access Tracker, team demos scheduled

## Days 61–90: APPLICATION + SCALING

- Launch 2 org-wide AI hackathons or capability sprints
- Run reuse audits what projects/tools can be templatized?
- Introduce shared dashboards of AI use cases by BU
- Publish an internal "AI Stack Catalogue" with common tools, workflows, and owners
- Deliverables: Stack Catalogue 2 reuse wins, leaderboard of AI adoption per team

#### **Success Metrics**

- % of teams with at least 1 Al Anchor
- No. of AI use cases shared org-wide
- % adoption of shared sandbox/tools
- Reuse score (% of new projects using templates/components from others)
- No. of teams participating in capability sprints/hackathons

# **Risks & Trade-Offs**

Risk	Mitigation
Knowledge remains tribal	Encourage async documentation with shoutouts/rewards
Governance overload	Start with light guardrails and scale as needed
Tool bloat	Centralize procurement and integration options
Too top-down	Keep room for local experimentation while aligning on standards

# Try This (Interactive Simulation)

**Scenario Challenge:** You're tasked with building a capability uplift plan across a scattered enterprise.

- 1. Create a dashboard to track team-level capability maturity
- 2. Design a lightweight model usage guide for internal teams
- 3. Propose 2 reuse wins that could reduce effort by 30% in other teams

# **Resources to Use:**

 Al Capability Uplift Canvas (available in 'Visual Frameworks & Custom Models' Section)

Thank you Happy Learning!