Topic Analysis and Synthesis Report

Software Project Management (SOEN 6481)

”Topic 13: How does project management differ between hardware and software projects”

by

Sharanya Akkenapally (40238186)

Under the Supervision of
Professor: Pankaj Kamthan

UNIVERSITÉ
Concordia
UNIVERSITY

Department of Computer Science and Software Engineering

Contents

1 Introduction **3**

1.1 Motivation 3

1.2 Problem Statement 3

1.3 Objectives 3

2 References **4**

Abstract

This study explores essential distinctions between project management techniques for tangible (hardware) and intangible (software) products. Even while fundamental project management concepts are relevant everywhere, there are notable differences because of the distinct qualities of these deliverables. Hardware projects often follow a waterfall-like life cycle that is organized and includes strict change management procedures and clearly defined stages. On the other hand, agile approaches are frequently used in software projects, which has flexibility to change according to the needs. The software development industry is characterized by frequent changes and the requirements according to the client requests, therefore flexibility is essential. Furthermore, there are significant differences in these two fields' testing schedules and methodologies. Software projects need ongoing testing since they have several interim deliverables, whereas hardware projects often condense testing near project finish.

In addition, there is a noticeable difference in the level of experience that project managers need in these areas. Because technology is changing at a slower pace in hardware projects, technical depth is an absolute necessity. But because software development is increasingly cross-functional, project managers in this field rely significantly on the specific expertise and skills of their teams and need to be highly competent leaders. Project managers must adjust their approaches to the particular requirements of each domain in order to successfully execute projects containing both tangible and intangible outcomes, therefore it is critical that they recognize these differences in project management methodologies.

1 Introduction

1.1 Motivation

Why Investigate the Problem, Domain, and Industry?

The need to better understand the key differences in project management techniques between software and hardware projects is what motivated this study. The aim of this research is driven by the practical applications it can improve project success, enhance decision-making, facilitate efficient teamwork, and encourage continuous growth for project managers.

Understanding the distinctions between project management for physical and intangible outputs is crucial for enhancing project management strategies and producing better project outcomes. We seek to improve collaboration between project managers, teams, and stakeholders by identifying the particular difficulties. This will lead to more efficient project management and goal alignment. Making strategic choices about project investments can have a big impact. Our goal is to provide corporate executives and decision-makers with knowledge about the subtle differences between software and hardware project management techniques. They may use this information to make well-informed decisions about the allocation of resources and strategic planning. Project management is dynamic, meaning it must be adjusted often to be up to date with changes in the market. Our research encourages a culture of continuous learning and professional development to guarantee that project management practitioners remain effective and relevant in their roles.

1.2 Problem Statement

This study looks into the key differences between software and hardware projects' project management methodologies. It seeks to address the particular difficulties and demands that project managers have when working with outputs that are both tangible (hardware) and intangible (software).

1.3 Objectives

These are the objectives for our study:

- **Understanding Distinctions**: The research aims to uncover the distinctions between hardware and software project management. Understanding these differences is essential for project managers to adapt their practices effectively, make informed decisions, and achieve better project outcomes.
- **Enhancing Project Success**: By identifying and emphasizing best practices, the research seeks to enhance project success. This includes reducing risks, improving overall project quality, and ultimately leading to more successful project outcomes.
- **Facilitating Effective Collaboration**: Effective collaboration is paramount for project success. The research promotes collaboration among project stakeholders, ensuring that teams work cohesively, align with project goals, and execute projects efficiently.
- **Informing Strategic Decision-Making**: Informed strategic decisions about resource allocation are crucial for project success and resource optimization. The research empowers decision-makers to allocate resources more efficiently, make strategic investments, and enhance project outcomes.
- **Supporting Academic and Educational Growth**: This research contributes to academic and educational growth by providing valuable resources and insights for students, researchers, and educators. It enriches academic experiences, fosters research, and improves the quality of project management education.
- **Fostering Continuous Learning**: The project management landscape is dynamic, requiring continuous learning and adaptability. The research encourages project management practitioners to remain effective, relevant, and capable of responding to industry changes and challenges.

2 References

[1] Kamthan, Pankaj. - INTRODUCTION TO FRAMEWORKS FOR SOFTWARE MEASUREMENT. Retrieved from Concordia ENCS: http://users.encs.concordia.ca/~kamthan/courses/soen-6611/software_measurement_frameworks_introduction.pdf

[2] Lowe, S. A. (n.d.). 9 metrics that can make a difference to today's software development teams. Retrieved from TechBeacon: https://techbeacon.com/app-dev-testing/9-metrics-can-make-difference-todays-software-dev

[3] Bottoni, Alex. (2012, December 6). What are the metrics for software documentation? Retrieved from StackExchange: https://pm.stackexchange.com/questions/8089/what-are-the-metrics-for-software-documentation

[4] Microsoft Learn. (2022, October 10). Code metrics values. Retrieved from: https://learn.microsoft.com/en-us/visualstudio/code-quality/code-metrics-values?view=vs-2022

[5] Stackify. (2017, September 16). What Are Software Metrics and How Can You Track Them? Retrieved from Stackify: https://stackify.com/track-software-metrics/

[6] G¨ohler, S. M., Eifler, T., & Howard, T. J. (2016). Robustness Metrics: Consolidating the multiple approaches to quantify Robustness. Retrieved from DTU: https://backend.orbit.dtu.dk/ws/files/125649168/Robustness_Metrics_Consolidating_the_multiple_approaches_to_%20quantify_Robustness_WebReady.pdf

[7] Try QA. (n.d.). What is Portability testing in software? Retrieved from: https://tryqa.com/what-is-portability-testing-in-

[8] Template Lab (n.d.). 40 Use Case Templates & Examples (Word,PDF). Retrieved from Template Lab: https://templatelab.com/use-case-templates/

[9] Paradigm, V. (n.d.). What is a Use Case Diagram? Retrieved from Visual Paradigm: https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-use-case-diagram/