

**School of Engineering and Computing**

**MSc <*insert programme title here*>**

**Interim Report**

**Agile Methodology and Its Impact on IT Project Success**

by

***<student’s full name>***

***<student’s banner id>***

**Date of submission: *<dd/mm/yyyy>***

**Supervisor: *<full name of supervisor>***

**Moderator: *<full name of moderator>***

**Table of Contents:**

[**1. Introduction** 3](#_Toc182241976)

[1.1 Problem Statement 3](#_Toc182241977)

[1.2 Research Objective 4](#_Toc182241978)

[1.3 Research Questions: 4](#_Toc182241979)

[1.4 Significance of the Research: 5](#_Toc182241980)

[**2. Summary Literature Review:** 6](#_Toc182241981)

[2.1. Agile Methodologies and Their Principles: 6](#_Toc182241982)

[2.2. Comparative Studies on Agile and Traditional Project Management: 7](#_Toc182241983)

[2.3. Success Factors for Agile in Large Organizations: 7](#_Toc182241984)

[2.4. Agile in Large-Scale Organizations 7](#_Toc182241985)

[2.5. Hybrid Approaches: Blending Agile and Traditional Methods: 8](#_Toc182241986)

[**3.0 Introduction** 10](#_Toc182241987)

[3.1 Research Design 10](#_Toc182241988)

[3.1.1 Systematic Literature Review 10](#_Toc182241989)

[3.1.1.1 Search Strategy 11](#_Toc182241990)

[3.1.1.2 Study Selection Criteria 13](#_Toc182241991)

[3.1.1.3 Data Analysis Approach 14](#_Toc182241992)

[3.1.2 Artefact Development 15](#_Toc182241993)

[3.1.2.1 Artefact Design and Development Process 16](#_Toc182241994)

[3.1.2.2 Task Management Design 16](#_Toc182241995)

[3.1.2.3 Sprint Planning and Execution 16](#_Toc182241996)

[3.1.2.4 Collaboration and Communication Setup 17](#_Toc182241997)

[3.1.2.4 Data Analysis Approach 17](#_Toc182241998)

[3.3 Plan for Completion 18](#_Toc182241999)

[**References:** 19](#_Toc182242000)

**LIST OF FIGURES:**

[Figure 1: The core phases of Systematic Literature Review 11](#_Toc182242274)

**LIST OF TABLES:**

[***Table 3.1 Study Selection Criteria*** 13](#_Toc182242617)

# **1. Introduction**

Agile methodology has become very popular in software development due to its flexibility, progressive process, and customer collaboration focus. Unlike the traditional Waterfall model, which follows rigid, step-by-step phases, Agile aligns quickly to changing project requirements and aims to deliver valuable solutions faster.  Agile methodology is an interconnected group process where people are designed to deliver a potentially usable and valuable solution to the customer as soon as possible.

The essence of Agile as put in the Agile Manifesto (Beck et al., 2001), has been embraced across several business sectors especially those whose projects involve Information Technology, location where requirements may be inherently volatile. The purpose of this project therefore is to evaluate the effectiveness of Agile methodology tools, specifically Asana and Jira, in improving project outcomes. This will involve comparing their features, assessing their influence on key success factors such as delivery time, budget adherence, and team efficiency.

This interim report will describe the work accomplished to date and will include a brief literature survey of the available literature, a description of the research method used in the project, and a work plan for the completion of the project. It will also identify possible limitations and countermeasures to the investigations to keep the study on the right track for the successful accomplishment of the project.

1.1 Problem Statement

Agile project management has gained significant interest in the IT sector due to its inherent flexibility and emphasis on collaboration among team members. This methodology allows for iterative development and continuous feedback, which are Important in dynamic environments where requirements frequently change. Tools such as Asana and Jira have emerged as pivotal enablers of Agile practices, facilitating essential functions like task tracking, sprint planning, and real-time collaboration among team members. These tools not only streamline project management processes but also enhance team communication and efficiency, thereby contributing to improved project outcomes (Özkan & Mishra, 2019; Suetin et al., 2016).

The failure to effectively implement Agile methodology can result in project delays, budget overruns, and diminished quality of deliverables, ultimately limiting organizations' potential for innovation and competitiveness in a rapidly changing technological environment. Studies suggest that organizations utilizing Agile methodologies typically achieve higher success rates in IT projects. For instance, Khoza and Marnewick (2020) indicate that Agile methodologies yield better process improvement outcomes compared to traditional Waterfall methodologies. Conversely, Kalenda, Hyna, & Rossi (2018) identified that the use of Agile within large organizations introduces unique challenges, complicating successful implementation.

Despite the growing adoption and benefit of Agile methodologies, organizations still face challenges in achieving project success, such as maintaining stakeholder engagement and adapting workflows to fit specific project needs. Understanding how different Agile project management tools impact project outcomes is important, most especially for organizations seeking to optimize their practices and enhance productivity.

This study will therefore evaluate the effectiveness of Agile project management tools, specifically Asana and Jira on IT project success. It will compare the features and functionalities of these tools, analyze their contributions to key success factors, and assess how they facilitate Agile methodologies in various organizational settings.

1.2 Research Objective

Specifically, this study aims to:

1. Evaluate and compare Asana and Jira’s task management, sprint planning, and collaboration features by assessing ease of use, task assignment, and tracking, using observational testing metrics such as task assignment time and navigation efficiency
2. Analyse the impact of Asana and Jira on IT project success by measuring predefined success metrics, including task completion rates and workflow efficiency
3. Develop and demonstrate Agile artifacts (such as sprint boards and task boards) using Asana and Jira, chosen for their role in visualising Agile workflows, and assess ease of setup and usability through predefined metrics like setup time and accuracy in reflecting Agile processes.

## 1.3 Research Questions:

1. Is there a significant difference between Asana and Jira in terms of ease of use, task assignment, and tracking efficiency when managing tasks and sprint planning in IT projects?
2. Is there a significant difference between the impact of Asana and Jira on IT project success, particularly in terms of task completion rates and workflow efficiency, as measured by predefined success metrics?

## 1.4 Significance of the Research:

In this study, several factors make the research important. First of all, it responds to ongoing discussion in the IT sector regarding the actual effects that Agile processes have especially in a large-scale organization. Agile practices are recognized to provide greater flexibility and responsiveness in project environments today, yet organizations still face challenges in achieving project success. This research will help to address that issue by presenting an extensive comparison of Agile project management tools, focusing on Asana and Jira on IT project success.

Second, the implications of this research will be of importance to IT project managers and organizations that want to adopt, improve, or select optimal project management frameworks. This study will assist organizations in making better decisions in the adoption and effective implementation of Agile methodologies due to the identification of conditions that lead to success or failure. In addition, the results could be used by organizations to optimize their project management practices and increase project efficiency as well as customer satisfaction.

Last but not least, this research contributes to the global portfolio of work undertaken in the project management methodologies field by providing empirical information on the efficacy of Agile approaches in complex, massive-scale organizational contexts. It contains the possibility of having an impact on future studies and will stimulate additional discussion on the topic of hybrid Agile models that integrate the elements of traditional approaches.

# **2. Summary Literature Review:**

The agile approach has its origin in the Agile Manifesto developed by Beck et al in the year 2001; this document set more value on the effective collaboration of individuals and communication rather than the procedures and tools, delivering working software more than creating documentation, customer collaboration more than carrying out contracts and responding to change more than following a plan. Such principles are in contrast with the Waterfall approach, within which the process is linear and divided into stages – from requirements gathering to design, development, testing, and deployment (Adenowo & Adenowo, 2013). Advocates of Agile claim that investments in the approach help to minimize the impact of the risks and provide for more flexibility when it comes to constantly changing customer requirements which is highly likely in software development projects.

## 2.1. Agile Methodologies and Their Principles:

Agile frameworks comprise several frameworks with Scrum, Kanban, Extreme programming (XP), and Lean software development among others. Each of these frameworks has its own set of practices, but they all adhere to the core principles of Agile: reiteration, interdisciplinary cooperation, and feedback loops and improvement. Sprint and put, Scrum's inventors, Schwaber and Sutherland (2020), define it as a process for creating, building, and continuously improving products or services through successive cycles of work. One of the advantages coming from the usage of Scrum is the ability to change something constantly – this is possible due to the cyclic structure of Scrum and is used to respond to emerging new requirements without directly affecting the overall scheme of the project.

The other mainstream Agile framework is Kanban which is aimed at visualizing the work, controlling the amount of work being performed as well as maintaining the work's flow (Anderson, 2010). Kanban's boards allow teams to track how they are progressing through their workflow at any time and recognize problem areas. Based on the outlines of Poppendieck and Poppendieck (2003), lean software development is a modification of lean production disguised as lean software improvement.

## 2.2. Comparative Studies on Agile and Traditional Project Management:

A few samples of comparative research work present how Agile has advantages over other conventional project management techniques. Serrador and Pinto (2015) surveyed over 1000 projects to conclude that Agile was more successful than the Waterfall method. Based on their studies, they argue that Agile addresses program and organizational priorities of customer collaboration and flexibility and thus yields greater success. Likewise, Kupiainen et al. (2015) also provided a systematic literature review and found out that Agile measures such as the velocity of the team and burndown chart facilitate better tracking and improved identification and handling of problems and so on, therefore, Agile metrics make the team more efficient and enhances the success rates among Agile projects.

However, not all works presented tend to endorse Agile without a second glance. Conboy (2009) also points out that the Agile method can be very useful for small to medium development projects but could not be applied to very large and very complex development projects where there is a great need for documentation and formal procedures. Similarly, Boehm and Turner (2004) explain that since Agile is fully flexible, costs and requirements are likely to grow as well as the requirements definition is not likely to be complete, which leads to project delay and cost increase. These critics' claims state that a more manufacturing-oriented methodology such as Waterfall with the right focus on initial planning and rigid workflow may be more appropriate when a specific contract length and deliverables are required due to fewer possibilities of scope slippage.

## 2.3. Success Factors for Agile in Large Organizations:

Academic sources also recommend that cultural change, support from leadership, and teams' willingness were the key factors, that defined Agile success in organizations. Campanelli and Parreiras (2015) note that Agile methods must be adjusted to the organization’s needs of a particular company. They postulate that the industrial-wide implementation of Agile has become a recipe for disaster in most large organizations with intricate projects. As such, it is more strategic for organizations to identify and apply Agile practices suited to the organizations and corporate objectives, system limitations, and personnel.

## 2.4. Agile in Large-Scale Organizations

Large-scale organizations are increasingly demonstrating interest in the implementation of Agile methodologies. Although Agile was originally intended for, and aptly applied to, small, collocated teams in fast-changing requirements, its adoption and applications have increased in large enterprises. Scaling up does bring with it unique challenges and complexities. According to Dikert et al., large-scale Agile transformations require adaptation of their classic Agile frameworks for complexities introduced by large teams and organizational hierarchies and extensive involvement of stakeholders. Key challenges identified in the systematic literature review on Agile transformations in large organizations include resistance to change, lack of management support, and difficulties related to the maintenance of coordination across teams. They also indicated that organizations should fit Agile practices into their contexts, which could be in hybrid model forms and integrate traditional project management methods into Agile.

Moe et al. (2010) further identify leadership and team autonomy as paramount variables that ensure Agile works within large organizations. In instances where decision-making remains highly centralized, self-managing agile teams face challenges in flexibility and speed. This calls for Agile transformations to change the organizational culture by allowing the organization to be more decentralized in terms of decision-making and empowerment of teams.

## 2.5. Hybrid Approaches: Blending Agile and Traditional Methods:

Considering that the need for project management has always been evolutionary, hybrid project management approaches are one of the common current approaches adopted where organizations are faced with the diverse needs of projects. These hybrid models will allow organizations to exploit the flexibility of Agile in areas that call for rapid adaptation and customer feedback while retaining structured planning and documentation processes from traditional methods for other aspects of the project. One of the first to advance a balanced approach from Agile to traditional methods was Boehm and Turner. They say Agile is ideal for environments where requirements are volatile or change at a rapid pace, but traditional methodologies would be more suitable for projects that require upfront planning, regulatory compliance, or extensive documentation. Such a hybrid approach enables an organization to switch from one methodology to the other. This can be depending upon the needs of different phases of various projects.

In large organizations, where projects can span across different departments, technologies, and geographic locations, hybrid approaches may be required to meet the variety of complexity. Such a project may follow Agile practices for the software development and customer-facing components but adhere to a more traditional Waterfall approach in the infrastructure and compliance-related aspects. Other studies by Stettina and Hörz (2015) reveal that in cases where Agile cannot be applied, for instance, in very regulated environments, projects with fixed deadlines, or fixed-price contracts, hybrid models turn out to be more efficient. A study they conducted on Agile adoption in large-scale organizations showed that hybrid models were able to adapt to the continuously changing needs of customers without losing control and predictability, which such a highly structured environment requires.

This research will evaluate the effectiveness of Agile project management tools, specifically Asana and Jira on IT project success. It will compare the features and functionalities of these tools, analyze their contributions to key success factors, and assess how they facilitate Agile methodologies in various organizational settings.

# **3.0 Introduction**

This chapter presents the methodology used to assess Agile methodology’s impact on IT project success through a comparative analysis of Asana and Jira. The study uses a mixed-method approach: a systematic review and artefact development, enabling both theoretical and practical evaluations.

## 3.1 Research Design

This study employs a mixed research method combining a systematic review with artefact development to analyse the impact of Agile methodology on IT project success. According to Almeidai (2018), mixed methods research is an approach that combines both quantitative and qualitative methods into a single study in order to provide a broader and more complete vision of a problem. In this study, the quantitative method involved creating artefacts to compare the features and functionalities of the selected agile software. On the other hand, the qualitative method involved a Systematic Literature Review that enabled a rigorous synthesis of existing research on Agile project management tools, focusing on key functionalities like task management, sprint planning, and collaboration (Varsha, Chakraborty, & Kar, 2024; Xiao & Watson, 2019). These methods offer a comprehensive understanding by blending theoretical insights with practical demonstrations.

## 3.1.1 Systematic Literature Review

A systematic literature review (SLR) was conducted because it adheres to a rigorous and structured methodology and provides a comprehensive coverage and unbiased synthesis of relevant literature in a particular research area. Unlike general literature reviews, which may lack transparency and rigour in the literature search and selection process, systematic reviews adhere to a predefined protocol and transparent methodology, reducing the risk of selection bias and ensuring the reproducibility of the findings. In addition, the SLR approach ensures that all relevant studies are identified and included, regardless of publication status or geographical location, thereby maximizing the comprehensiveness and generalizability of the findings. The main goal of conducting the SLR was to evaluate the impact of agile project management tools, specifically Asana and Jira, on IT project success, focusing on task completion rate and delivery timelines adherence (Xiao & Watson, 2019).

The SLR approach begins with well-defined research questions which guide the literature search, selection, and data analysis. The primary research question guiding this study’s SLR is: “What is the influence of project management tools, specifically Asana and Jira, on IT project success, focusing on task completion rate and delivery timelines adherence?” In this study, the researcher adopted the SLR’s guidelines proposed by Kitchenham et al (2009). According to Kitchenham et al. (2009), a SLR consists of three core phases: (i) planning, (ii) conducting, and (iii) reporting the review. For the planning phase of SLR (Fig. 1), a research plan was developed to specify the research questions, keywords, search string and repositories, and inclusion/exclusion criteria for selecting potential research articles.

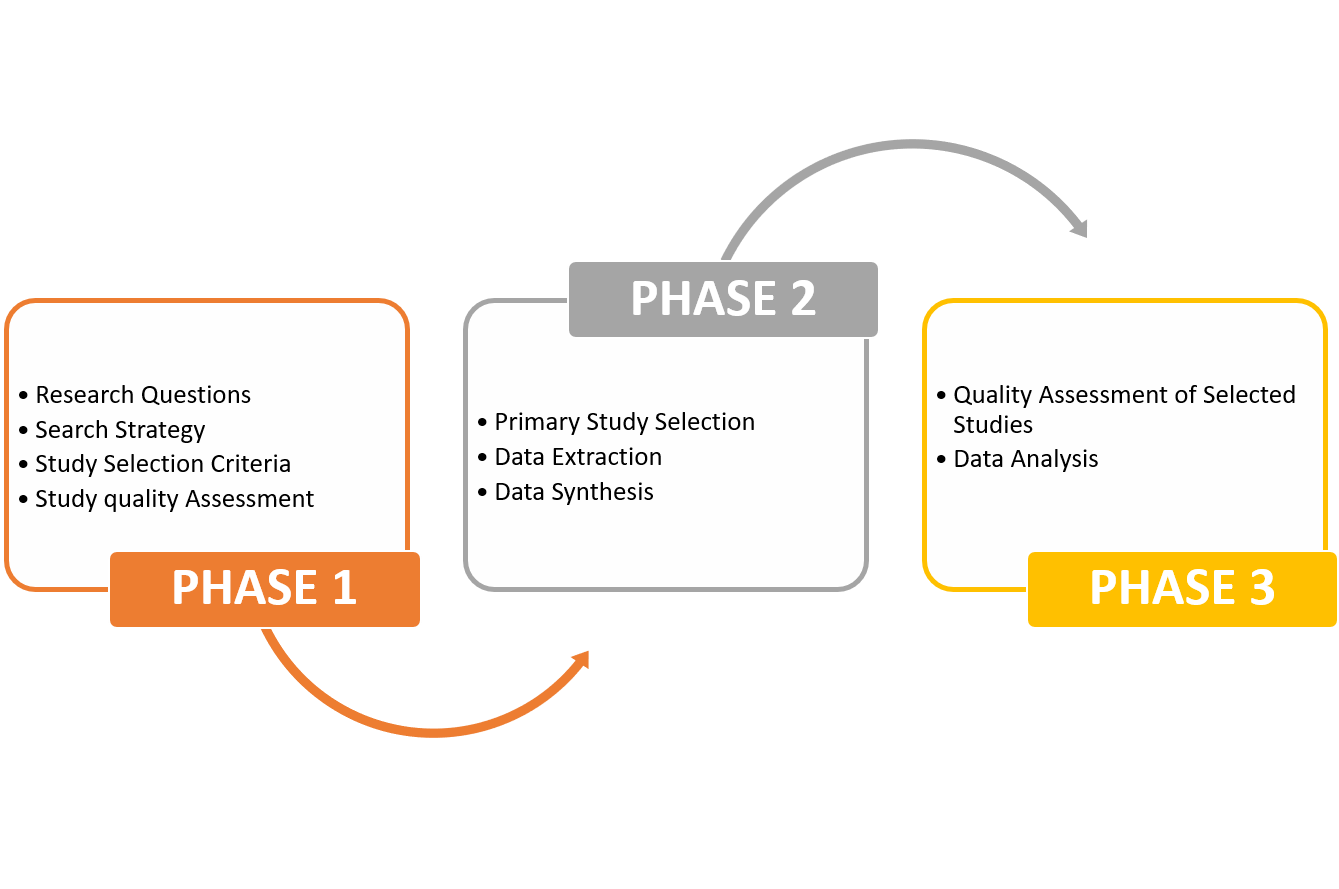


Figure 1The core phases of Systematic Literature Review

### 

### 3.1.1.1 Search Strategy

#### In this step, a search query and repositories are identified and selected (respectively) to find the relevant studies

1. *Search Query Structure*

The search query employed to gather the data for the studies was specifically designed for the chosen databases and followed a systematic method to identify relevant literature, as outlined below:

1. **Broad Search Query**: The initial query includes central terms related to the research questions, focusing on keywords like "Agile methodology," "IT project success," "task management," "sprint planning," and specific tools such as "Asana" and "Jira."

2. **Breaking Down Key Terms**: Each key term is explored to include related synonyms and variations. For example:

* Agile methodology: "Agile project management," "Agile framework," "Agile practices."
* IT project success: "project outcome," "project efficiency," "successful project delivery."
* Task management: "task handling," "workflow management," "task tracking."
* Sprint planning: "iteration planning," "sprint scheduling," "sprint backlog."
* Collaboration functionalities: "team collaboration," "team communication," "collaborative tools."
* Asana and Jira: "Asana software," "Jira tool," "project management software."

3. **Combining Synonyms and Variations**: Synonyms and variations of key terms are combined using Boolean operators (AND, OR) to refine the search. For example:

(Agile methodology OR Agile project management OR Agile framework) **AND** (IT project success OR project outcome OR successful project delivery) **AND** (task management OR task tracking OR workflow management) **AND** (Asana OR Jira OR project management software).

4. **Utilizing Truncation and Wildcards**: Truncation and wildcards are used to capture different forms and derivatives of terms, ensuring exhaustive coverage. For instance:

* "Agil\*" to capture "Agile," "Agility," etc.
* "collabor\*" to include "collaboration," "collaborative."

5. **Using Filters or Limits:** Filters or limits are applied to narrow the search based on specific criteria such as publication date range or study design. For example:

* Limiting the publication date range to studies published between 2014 and 2024 to focus on recent research.

1. *Search Repository*

To identify the relevant articles, a range of source repositories were chosen in this research. The study utilised popular digital source repositories, such as the following, to gather potential publications:

* Science Direct
* IEEE Xplore
* Springer Link
* JSTOR
* Wiley Online Library

### 3.1.1.2 Study Selection Criteria

The inclusion and exclusion criteria for this study were designed to ensure that the selected literature is directly relevant to understanding the impact of Agile methodology on IT project success, particularly focusing on Asana and Jira functionalities. It includes two types of criteria, specifically inclusion and exclusion criteria, as illustrated in Table 1. Limiting the studies to those published within the last ten years prioritises current perspectives on Agile practices and project management tools, given the rapid advancements in both areas. Also, including empirical studies, case studies, systematic reviews, and meta-analyses, ensures that the findings are supported by robust data and established methodologies, thereby enhancing the reliability of the review. Focusing on studies that address specific Agile functionalities—task management, sprint planning, and collaboration—aligns with the research objectives and ensures a detailed exploration of how Asana and Jira can impact project success in IT. Lastly, peer-reviewed articles and studies published in English further ensure quality and accessibility, as well as relevance to a global audience. These criteria collectively facilitate a systematic, targeted literature review, displaying both relevance and rigour in the study selection process.

**Table 3.1 Study Selection Criteria**

|  |  |  |
| --- | --- | --- |
| Criterion | Inclusion Criteria | Exclusion Criteria |
| Language | Studies published in English-language | The studies that are not written in English are excluded. |
| Study Focus | Studies focused on Agile methodology, project management tools, and IT project success | Studies unrelated to Agile methodology, project management tools, or IT projects |
| Publication Type | Peer-reviewed journal articles, conference proceedings, or reputable industry white papers | Editorials, opinion pieces, blog posts, or non-peer-reviewed sources are excluded |
| Publication Date | Studies published within the last 10 years to ensure relevance to current Agile and IT practices | Studies published more than 10 years ago are excluded |
| Project Management Tools | Studies analysing Asana, Jira, or both as agile project management tools | Studies focusing on other project management tools without a comparative analysis involving either Asana or Jira, or both. |
| Functionalities | Studies evaluating task management, sprint planning, and collaboration functionalities in Agile | Studies not covering any of the key functionalities (task management, sprint planning, collaboration) |

### 3.1.1.3 Data Analysis Approach

The systematic literature review used in this study follows a thematic analysis for identifying, analysing, and reporting the patterns (themes) within the qualitative data. According to Braun and Clarke (2006), thematic analysis involves several steps, including data familiarisation, coding, theme identification, theme review, and theme definition.

1. **Data Familiarization:** The researcher begins by thoroughly reviewing the selected literature through multiple readings to gain a deep understanding of the main concepts, findings, and arguments related to Agile project management in Asana and Jira. This familiarization process establishes a solid foundation for a structured analysis of these tools’ impact on IT project success.
2. **Coding:** In the coding phase, key text segments relevant to the research objectives are systematically tagged and categorized. This involves identifying significant excerpts related to specific Agile functionalities, such as task management, sprint planning, and collaboration, and how they influence IT project success metrics like task completion rate and delivery timelines adherence.
3. **Theme Identification**: After coding, the researcher examines the coded segments to identify recurring ideas or patterns, known as themes, across the literature. Themes related to Agile project management might include “efficient task tracking,” “enhanced team collaboration,” or “improved sprint planning,” which reflect key attributes of effective Agile workflows in project management tools.
4. **Theme Review:** Once initial themes are identified, the researcher rigorously reviews the coded data to confirm that each theme accurately represents core ideas. This involves revisiting the coded passages to ensure they align well with Agile principles and accurately capture each tool’s functionalities.
5. **Theme Definition**: Finally, each theme is clearly defined and explained, synthesizing findings from the coded data to articulate its relevance to the research objectives. This step provides a cohesive interpretation of the primary themes, establishing their significance in understanding how Asana and Jira support Agile project management practices and contribute to IT project success.

## 3.1.2 Artefact Development

This section outlines the process of developing and evaluating computing artefacts to demonstrate agile workflows and practices in Asana and Jira. These artefacts will serve as demonstration models for how agile methodologies can be structured and executed in each tool. By focusing on core agile components like task management, sprint planning, and collaboration, the artefacts illustrate the tools' operational capacities in realistic project scenarios, highlighting the relative strengths and limitations of each tool. This practical approach also provides insight into the tools’ applicability and efficiency, contributing directly to evaluating their influence on IT project success metrics, such as delivery timelines.

### 3.1.2.1 Artefact Design and Development Process

This section explains the step-by-step process of designing and building the Agile workflows within Asana and Jira. Each artefact is developed with attention to standard Agile principles, ensuring it represents authentic Agile practices. Asana and Jira were chosen for their popularity and specific design to support Agile project management, particularly within the IT industry. Both tools offer extensive features for managing tasks, planning sprints, and fostering team collaboration, making them ideal for this study. The **Initial Setup** involves creating two project environments (one in Asana and one in Jira) with similar objectives, allowing for a fair comparison. Each tool’s environment will be configured to reflect Agile roles (e.g., product owner, Scrum master, team members) and project stages (backlog, sprint planning, etc.).

### 3.1.2.2 Task Management Design

* **Objective**: To demonstrate how Asana and Jira enable efficient task management, from backlog creation to task assignment and tracking.
* **Artifact Design Steps**:
  + **Task Backlog Creation**: Both tools are populated with a project backlog, listing tasks with detailed descriptions, priority levels, and dependencies.
  + **Task Assignment and Scheduling**: Tasks are assigned to specific team members, completion times are estimated, and deadlines are set.
* **Tracking and Progress Visualization**: Each tool’s visual aids (e.g., Kanban boards in Jira, and task lists in Asana) are used to monitor task progression

### 3.1.2.3 Sprint Planning and Execution

* **Objective**: To assess each tool’s capacity to support sprint planning and track sprint progress, demonstrating how these functionalities align with Agile principles.
* **Artifact Design Steps**:
  + **Sprint Setup**: Configure sprint cycles (e.g., two-week sprints) in both tools, including a sprint backlog and goals.
  + **Sprint Planning Meeting Simulation**: Outline a mock sprint planning session where tasks are moved from the backlog into the sprint cycle.
  + **Execution and Daily Stand-Ups**: Demonstrate how both tools allow tracking of daily progress and updating task status. This includes setting up “stand-up” features (e.g., comments, and status updates) to simulate daily team check-ins.

### 3.1.2.4 Collaboration and Communication Setup

* **Objective**: To evaluate how each tool fosters collaboration, focusing on features like comments, @mentions, file sharing, and notifications.
* **Artifact Design Steps**:
  + **Collaborative Task Updates**: Demonstrate how team members can update task status, add comments, and communicate directly within each tool.
  + **Real-Time Notifications**: Enable and document notification settings that alert team members of updates or changes in task status.
  + **File Sharing and Documentation**: Illustrate how files are uploaded and shared in both tools to support collaborative work.

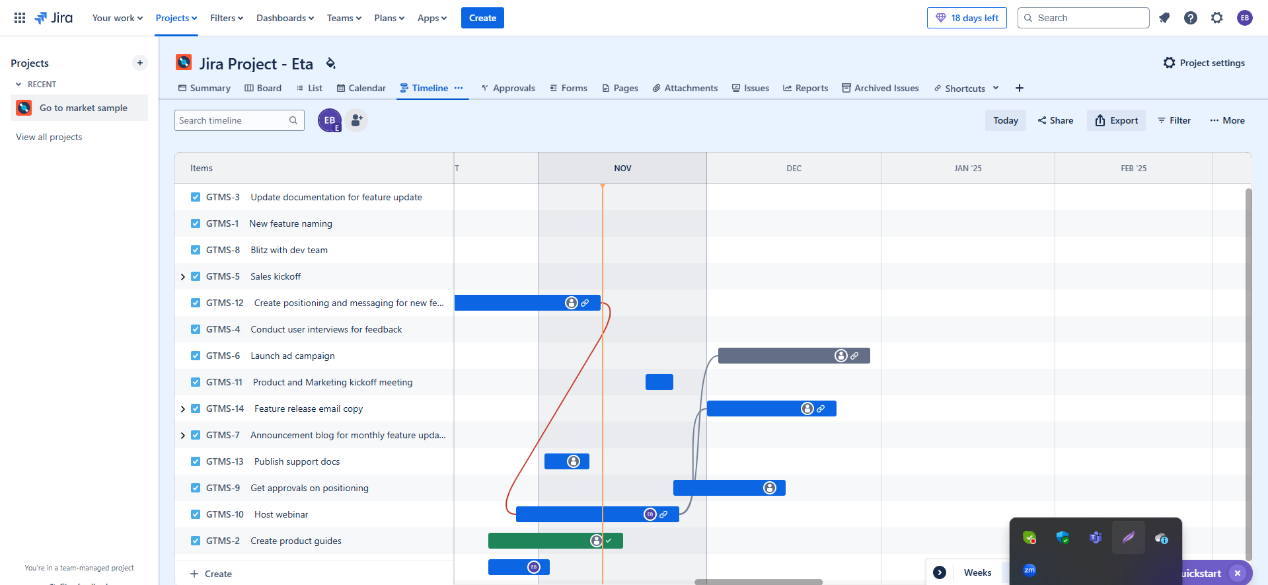


Fig 3.1: Timeline Tracking in Jira (in progress)

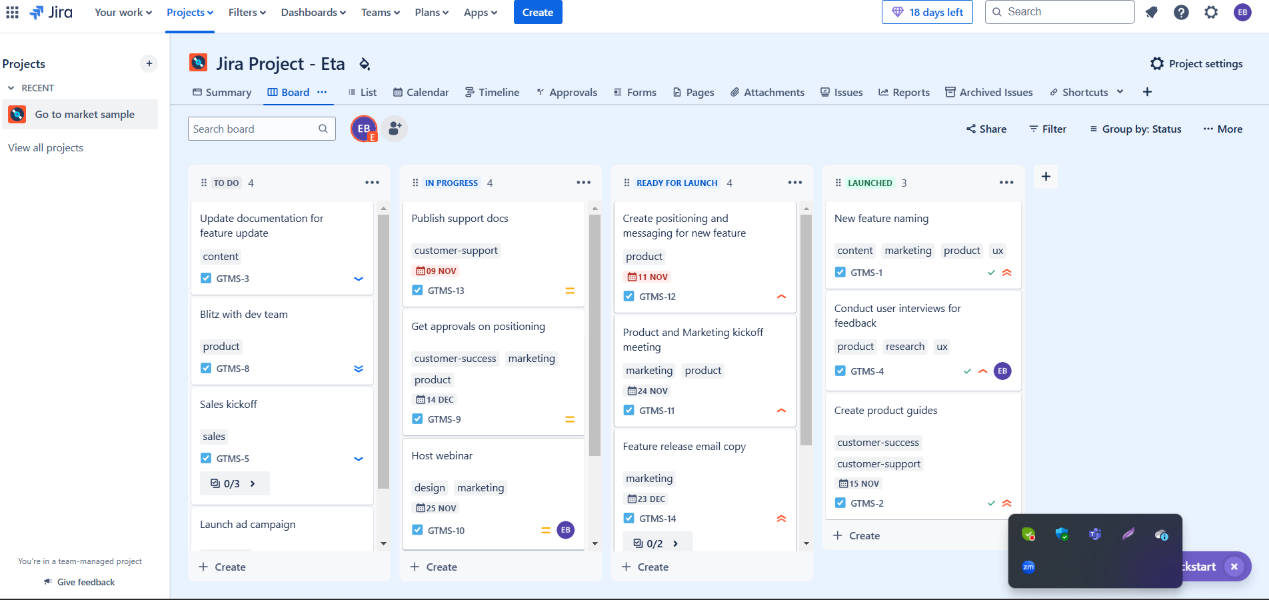


Fig 3.2: Task Tracking Workboard in Jira (in progress)

### 3.1.2.4 Data Analysis Approach

For analyzing the artefact development in this research, a descriptive analysis approach will be suitable, focusing on feature demonstration and functionality comparison without user testing. This analysis will document each Agile tool’s functionalities in a structured, objective manner, highlighting their task management, sprint planning, and collaboration capabilities. The data will be drawn from observations, configurations, and screenshots of workflows within Asana and Jira, following step-by-step documentation of Agile tasks and processes in each tool. A comparative analysis will be conducted by categorizing the main features demonstrated (e.g., task prioritization, sprint planning board, real-time updates) and assessing their alignment with Agile principles. Each feature’s effectiveness will be evaluated based on its operational visibility, functionality, and ease of use, with an emphasis on how well it supports Agile project success metrics, such as task tracking and timeline adherence. Observations will be recorded qualitatively, noting strengths, limitations, and unique aspects of each tool that impact Agile project management practices. Lastly, the findings will be summarized to provide a structured overview of Asana and Jira’s capabilities, using side-by-side comparisons for clarity. This approach will result in a clear analysis of the evident features in both tools, emphasizing their respective abilities to support Agile workflows and project success criteria without requiring external user feedback.

## 3.3 Plan for Completion

**Documentation**: I will begin by gathering and organizing all required project documents, including the detailed project plan, research specifications, and a structured schedule for each project phase. This step ensures I have a clear roadmap and that all preliminary paperwork is prepared and accessible.

**Dashboard Design in Jira and Asana**: I will design and configure dashboards in Jira and Asana to showcase critical Agile metrics, including task assignments, sprint planning, and team collaboration activities. This step involves customizing the tools to visually represent project KPIs such as task completion rates, backlog management, and sprint progress, ensuring they align with Agile methodologies.

**Data Analysis and Presentation**: With all data collected, I will conduct a detailed statistical analysis to interpret observational metrics and determine the effectiveness of Jira and Asana for Agile project management. This analysis will cover metrics like average task completion time, user satisfaction, and dashboard usability. I will then compile the results into tables and charts for a clear, visual presentation of the findings.

**Literature Review and Writing**: I will expand and refine the literature review to establish a contextual foundation for my research, analyzing prior studies on Agile tools and methodologies. Afterward, I will write the main body of the report, detailing the research methods, data analysis, and findings, discussing how the results illustrate the impact of Jira and Asana on Agile project success.

**Review, Feedback, and Revision**: I will submit the first draft of the report to my supervisor and peers for review. Based on their feedback, I will make necessary revisions to enhance the clarity, depth, and quality of the document. This may involve refining the analysis, clarifying points, and addressing any identified gaps.

**Formatting and Proofreading**: I will meticulously review the document to ensure it adheres to all required formatting standards. I will proofread for grammar, style consistency, and formatting errors to produce a polished, professional report that meets submission standards.

**Submission**: Once final checks are complete, I will submit the finalized report by the due date, ensuring that all submission requirements are met.

**PowerPoint Presentation Preparation**: I will develop a comprehensive PowerPoint presentation to summarize the research objectives, methodologies, key findings, and conclusions. The presentation will include visual aids, such as graphs and charts, to effectively communicate the study results during the thesis defense or project presentation. Top of Form

Bottom of Form

# 

# **References:**

Adenowo, A. A., & Adenowo, B. A. (2013). Software engineering methodologies: a review of the waterfall model and object-oriented approach. International Journal of Scientific & Engineering Research, 4(7), 427-434.

Almeida, F. (2018). Strategies to perform a mixed methods study. *European Journal of Education Studies, 5*(1).

Anderson, D. J. (2010). *Kanban: Successful Evolutionary Change for Your Technology Business*. Blue Hole Press.

Beck, K., Beedle, M., van Bennekum, A., Cockburn, A., Cunningham, W., Fowler, M., ... & Thomas, D. (2001). *Manifesto for Agile Software Development*. Available at: <https://agilemanifesto.org/>

Braun, V., & Clarke, V. (2006). Using thematic analysis in psychology. *Qualitative Research in Psychology*, *3*(2), 77–101. <https://doi.org/10.1191/1478088706qp063oa>

Bloomfield, J., & Fisher, M. J. (2019). Quantitative research design. Journal of the Australasian Rehabilitation Nurses Association, 22(2), 27-30.

Boehm, B., & Turner, R. (2004). *Balancing Agility and Discipline: A Guide for the Perplexed*. Addison-Wesley Professional.

Brad, M. C., Birloi, F. C., Bratulescu, A., & Blaga, I. B. (2016). A comparative study of agile project management software tools. Academy of Economic Studies. Economy Informatics, 16(1), 27-38.

Campanelli, A. S., & Parreiras, F. S. (2015). Agile methods tailoring – A systematic literature review. *Journal of Systems and Software*, 110, 85-100.

Conboy, K. (2009). Agility from first principles: Reconstructing the concept of agility in information systems development. *Information Systems Research*, 20(3), 329-354.

Costa, V. (2017). Correlation and regression. Fundamentals of statistical hydrology, 391-440.

Dikert, K., Paasivaara, M., & Lassenius, C. (2016). Challenges and success factors for large-scale agile transformations: A systematic literature review. Journal of Systems and Software, 119, 87-108. https://doi.org/10.1016/j.jss.2016.06.013

Islam, A. K. M. Z., & Ferworn, A. (2020). A Comparison between agile and traditional software development methodologies. Global Journal of Computer Science and Technology, 20(2), 7-42.

Ivezić, Ž., Connolly, A. J., VanderPlas, J. T., & Gray, A. (2020). Statistics, data mining, and machine learning in astronomy: a practical Python guide for the analysis of survey data (Vol. 8). Princeton University Press.

Kupiainen, E., Mäntylä, M. V., & Itkonen, J. (2015). Using metrics in Agile and Lean software development – A systematic literature review of industrial studies. *Information and Software Technology*, 62, 143-163.

Kitchenham, B., Pearl Brereton, O., Budgen, D., Turner, M., Bailey, J., Linkman, S. (2009). Systematic literature reviews in software engineering - A systematic literature review, *Inf. Softw. Technol. 51 (1)*, 7–15. <http://dx.doi.org/10.1016/j.infsof.2008.09.009>.

Moe, N. B., Dingsøyr, T., & Dybå, T. (2010). A teamwork model for understanding an agile team: A case study of a Scrum project. Information and Software Technology, 52(5), 480-491. https://doi.org/10.1016/j.infsof.2009.11.004

Mostafa, B. (2015). Assessing the applicability of Traditional and Agile methodology to enhance success of software development projects.

Poppendieck, M., & Poppendieck, T. (2003). *Lean Software Development: An Agile Toolkit*. Addison-Wesley.

Qumer, A., & Henderson-Sellers, B. (2008). An evaluation of the degree of agility in six agile methods and its applicability for method engineering. Information and software technology, 50(4), 280-295.

Rao, K. N., Naidu, G. K., & Chakka, P. (2011). A study of the Agile software development methods, applicability and implications in industry. International Journal of Software Engineering and its applications, 5(2), 35-46.

Schwaber, K., & Sutherland, J. (2020). *The Scrum Guide™: The Definitive Guide to Scrum: The Rules of the Game*. Available at: <https://scrumguides.org/>

Serrador, P., & Pinto, J. K. (2015). Does Agile work? A quantitative analysis of agile project success. *International Journal of Project Management*, 33(5), 1040-1051.

Stettina, C. J., & Hörz, J. (2015). Agile portfolio management: An empirical perspective on the practice in use. International Journal of Project Management, 33(1), 140-152. https://doi.org/10.1016/j.ijproman.2014.03.008

Varsha P S, Chakraborty, A., & Kar, A. K. (2024). How to Undertake an Impactful Literature Review: Understanding Review Approaches and Guidelines for High-impact Systematic Literature Reviews. *South Asian Journal of Business and Management Cases, 13(*1), 18-35. <https://doi.org/10.1177/22779779241227654>

Xiao, Y., & Watson, M. (2019). Guidance on Conducting a Systematic Literature Review. *Journal of Planning Education and Research, 39*(1), 93-112. <https://doi.org/10.1177/0739456X17723971>