PM Pulse

Exploring a Feasible Platform for Project Management

Project ID: RP-24-002

Project Proposal Report

De Silva D.S.P.K.D

B.Sc. (Hons) Degree in Information Technology Specialized in Information Systems Engineering

Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

February 2024

PM Pulse

Exploring a feasible framework for Agile

Project Management For Web Applications

Project ID: RP-24-002

Project Proposal Report

De Silva D.S.P.K.D
Supervised by – Dr. Anuradha Jayakodi

B.Sc. (Hons) in Information Technology Specializing in

Information Systems Engineering

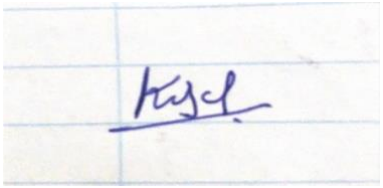
Department of Information Technology

Sri Lanka Institute of Information Technology
Sri Lanka

February 2024

DECLARATION

I declare that this is my own work, and this proposal does not incorporate without acknowledgement any material previously submitted for a degree or diploma in any other university or Institute of higher learning and to the best of my knowledge and belief it does not contain any material previously published or written by another person except where the acknowledgement is made in the text.

Name	**Student ID**	**Signature**
De Silva D.S.P.K.D	IT21034954	

The above candidates are carrying out research for the undergraduate Dissertation under my supervision.

Name of supervisor: Dr. Anuradha Jayakody
Name of co-supervisor: Mrs. Buddhima Attanayaka

Anuradha Jayakody (signed)

....2024/02/29.....

Signature of the supervisor: Date

(Dr. Anuradha Jayakody)

Buddhima Attanayaka (signed)

..... .2024/02/29.....

Signature of the co-supervisor: Date

(Mrs. Buddhima Attanayaka)

ACKNOWLEDGEMENTS

The senior lecturer at the Sri Lanka Institute of Information Technology, Dr. Anuradha Jayakody, and our Co - Supervisor Mrs. Buddhima Attanayaka, has my genuine gratitude, and that is the first thing I want to say. Without their help and committed participation in each step of the process, this paper would not have been completed. I sincerely appreciate all your help and patience over the past four years.

And also, all academic and non-academic staff at the Sri Lanka Institute of Information Technology for their insightful comments and ongoing support. I also would like to convey my gratitude to all the participants in the process of data collection and for giving their valuable time to complete this study. Finally, I would like to thank my family for encouraging me all the time and for their moral support to complete this. Finally, I wish to thank all those who have contributed in one way or another to this study.

ABSTRACT

Effective risk management is a critical factor in ensuring successful software projects. However, low to mid-level software companies often face challenges due to limited resources and insufficient historical data. Our proposed solution aims to address this gap by providing a systematic approach to risk identification and mitigation.

Our solution bridges the gap between risk assessment and actionable strategies, benefiting software companies striving for efficient project management.
Focusing on low to mid-level software companies in Sri Lanka is our strategic choice, and our research problem addresses a critical need.

In summary, our solution aims to empower low to mid-level software companies in Sri Lanka by providing a comprehensive risk management platform. By addressing these challenges head-on, we contribute to sustainable project outcomes and industry growth.

Keywords: Risk management, software projects, risk identification, mitigation strategies

Table of Contents

LIST OF FIGURES

LIST OF ABBREVIATIONS

Abbreviation	Description
SDLC	Software Development Life Cycle

1. Introduction

Risk management is a critical aspect of software project success, particularly in the dynamic landscape of agile development. As organizations increasingly adopt agile methodologies, the need for effective risk identification, assessment, and mitigation becomes paramount. In this proposal, we delve into the intricacies of risk management specifically tailored for agile software projects. Let's explore the key components:

1.1 About Risk Management for Agile Software Projects

Agile methodologies, such as Scrum and Kanban, emphasize iterative development, flexibility, and collaboration. Unlike traditional waterfall approaches, agile projects require continuous risk assessment throughout the project lifecycle. Risks can emerge at any stage, affecting sprint planning, daily stand-ups, and overall project outcomes. Our focus lies in understanding how risk management aligns with agile principles and how it contributes to project success.

1.2 Features of Risk Management

Effective risk management involves more than just identifying risks; it requires a holistic approach. We explore the following features:
Risk Identification: Agile teams must proactively identify risks early. We delve into techniques such as brainstorming sessions, checklists, and historical data analysis.
Contextual Mitigation Strategies: Tailored strategies are essential. We discuss contingency planning, risk transfer (insurance), risk reduction, and risk acceptance, all within the agile context.

1.3 Area of Research

Our research focuses on bridging the gap between existing risk management practices and the specific needs of agile software projects. By contextualizing risk management strategies and integrating them seamlessly into agile workflows, we aim to enhance project outcomes.

1.4 About the Research Component

Our proposed system, the 'Smart Surveillance System for Risk Identification and Mitigation in Agile Software Projects', will empower low to mid-level software companies. By addressing the unique challenges faced by agile teams, we anticipate improved resource allocation, reduced risks, and enhanced decision-making.

2. BACKGROUND STUDY AND LITERATURE SURVEY

1. Context and Significance

The software industry operates in a dynamic landscape, where projects vary widely in complexity, scope, and available resources. Risk management is a critical factor in ensuring project success, as unforeseen issues can lead to project delays, budget overruns, and compromised quality. In Sri Lanka, where the coconut industry significantly contributes to cuisine, foreign exchange, and employment, the unawareness and limited knowledge of growers regarding coconut diseases have led to devastating consequences for coconut lands. Similarly, in the software domain, inefficient risk management practices can hamper project success, especially in low to mid-level software companies [4].

2. Existing Literature

Risk Identification:

Researchers have explored various techniques for identifying risks early in the project lifecycle. These methods include:

Brainstorming Sessions: Collaborative sessions where project stakeholders identify potential risks based on their collective expertise.

Checklists: Predefined lists of common risks that project managers can use as a starting point.

Historical Data Analysis: Leveraging past project data to identify recurring risk patterns.

Expert Judgment: Relying on experienced professionals to assess risks based on their domain knowledge.

Empirical studies emphasize the importance of proactive risk identification. By identifying risks early, project teams can allocate resources effectively and implement targeted risk mitigation strategies [5].

Risk Mitigation Strategies:

Effective risk management involves more than just identifying risks; it requires tailored mitigation strategies. The literature highlights the following strategies:

Contingency Planning: Developing backup plans to address potential risks when they materialize.

Risk Transfer (Insurance): Transferring risk to external parties through insurance policies.

Risk Reduction: Taking actions to minimize the likelihood or impact of risks (e.g., process improvements, redundancy).

Risk Acceptance: Acknowledging certain risks as unavoidable and planning accordingly.

Case studies demonstrate how well-implemented mitigation strategies positively impact project outcomes. For instance, a mid-level software company that proactively addressed technical risks by diversifying its technology stack reduced the likelihood of project delays [6].

Agile Projects and Risk Management:

Agile methodologies, such as Scrum and Kanban, emphasize iterative development, flexibility, and collaboration. In agile projects:

Risk Identification: Agile teams continuously assess risks during sprint planning, daily stand-ups, and retrospectives.

Mitigation Strategies: Agile practices, such as frequent communication, adaptive planning, and incremental delivery, inherently address risks.

Client Collaboration: Agile principles encourage close collaboration with clients, enabling real-time risk assessment and adjustments.

3. Research Gap and Objectives

Despite existing research, there remains a gap in accessible risk management solutions tailored specifically to low to mid-level software companies working in agile environments. While some studies address risk management in general, few delve into the unique challenges faced by agile teams in smaller organizations. Our research aims to fill this gap by developing a user-friendly system that integrates risk identification, mitigation, and decision support, specifically tailored for agile software projects.

4. Significance and Contribution

Our proposed system will empower software companies by:

Proactively identifying risks early in the agile project lifecycle.

Providing context-specific mitigation strategies aligned with agile practices.

Improving project success rates and resource allocation in agile environments.

In summary, our research seeks to enhance risk-awareness and decision-making in agile software projects, ultimately benefiting the industry as a whole.

3. RESEARCH GAP

1. Research Gap

The existing literature on risk management in software projects has primarily focused on large enterprises and well-established organizations. However, there is a noticeable gap when it comes to risk management practices tailored specifically for low to mid-level software companies. These smaller organizations face unique challenges due to resource constraints, limited historical data, and the absence of dedicated risk management roles. Consequently, the available risk management solutions often fall short in addressing the specific needs of agile projects within these companies.

2. Missing Elements in Existing Research Literature

The following elements are notably missing from the existing research literature:
Agile-Specific Risk Management: While agile methodologies have gained prominence, there is a lack of comprehensive risk management frameworks specifically designed for agile software development. Existing models often do not fully align with agile principles, leading to suboptimal risk management practices.
Contextual Adaptation: Most risk management approaches assume a one-size-fits-all approach. However, low to mid-level software companies operate in diverse contexts, including varying project sizes, team structures, and client interactions. The literature lacks context-aware risk management strategies that can be adapted to specific project scenarios.
Decision Support Tools: While risk identification and mitigation strategies are well-documented, practical decision support tools are scarce. Agile project teams need real-time

insights to make informed decisions during sprints, releases, and retrospectives. Existing tools do not adequately address this need.

3. Research Objectives

Our research aims to bridge the identified gap by achieving the following objectives:
Developing an Agile-Centric Risk Management Framework:
We will create a risk management framework specifically tailored for agile software projects in low to mid-level companies.
The framework will incorporate agile practices, such as iterative development, adaptive planning, and client collaboration.
Contextualizing Risk Management Strategies:
We will analyze contextual factors (e.g., project size, team composition, client engagement) and adapt risk management strategies accordingly.
The goal is to provide practical guidelines that resonate with the unique challenges faced by agile teams in smaller organizations.
Building a Decision Support System:
Our proposed system will offer real-time risk insights during sprint planning, daily stand-ups, and retrospectives.
By integrating risk data into agile ceremonies, we aim to enhance decision-making and resource allocation.

4. Expected Impact

Our research will contribute to the field of risk management by providing agile practitioners with actionable tools and guidelines. By addressing the specific needs of low to mid-level software companies, we anticipate improved project outcomes, reduced risks, and enhanced resource utilization.
References: “Risk Identification and Risk Mitigation Instruments for Global Software Development: Systematic Review and Survey Results,” 2011 IEEE Sixth International Conference on Global Software Engineering Workshop . “Risk Identification, Mitigation and Avoidance Model for Handling Software Risk,” 2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks . “Risk-Based Testing: Identifying, Assessing, Mitigating & Managing Risks Efficiently in Software Testing,” SSRN .

	RESEARCH [1]	RESEARCH [2]	RESEARCH [3]	SOLUTION
Risk Identification Technique - ML models	✓	✗	✗	✓
Risk Mitigation Approach	✗	✓	✗	✓
Aggregates Risk Reporting	✗	✗	✗	✓
Continuous Improvement	✗	✗	✓	✓

Comparison with existing research projects

4. RESEARCH PROBLEM

The research problem of enhancing risk management for agile software projects in low to mid-level companies is a critical area of study. While agile methodologies offer flexibility and iterative practices, existing risk management approaches often fall short in addressing the unique needs of smaller organizations. These challenges include the lack of agile-centric frameworks, context ignorance, and a real-time decision support gap. Specifically, most risk management models are designed for traditional waterfall projects, which don't fully align with agile principles. Consequently, risk management practices in agile contexts may be suboptimal. Additionally, current strategies lack contextual adaptation, assuming a one-size-fits-all approach and disregarding the varying project sizes, team compositions, and client interactions typical of agile environments. Furthermore, agile practitioners require real-time insights during sprint planning, daily stand-ups, and retrospectives, but decision support tools that seamlessly integrate risk data into agile ceremonies are scarce. Addressing these gaps is crucial for effective risk management in agile software projects.

The research problem of enhancing risk management for agile software projects in low to mid-level companies is a critical area of study. While agile methodologies offer flexibility and iterative practices, existing risk management approaches often fall short in addressing the unique needs of smaller organizations. These challenges include the lack of agile-centric frameworks, context ignorance, and a real-time decision support gap. Specifically, most risk management models are designed for traditional waterfall projects, which don't fully align with agile principles. Consequently, risk management practices in agile contexts may be suboptimal. Additionally, current strategies lack contextual adaptation, assuming a one-size-fits-all approach and disregarding the varying project sizes, team compositions, and client interactions typical of agile environments. Furthermore, agile practitioners require real-time insights during sprint planning, daily stand-ups, and retrospectives, but decision support tools that seamlessly integrate risk data into agile ceremonies are scarce. Addressing these gaps is crucial for effective risk management in agile software projects.

5. OBJECTIVES

5.1 Main Objectives

Main objective is to develop an agile-centric risk management framework.

Create a comprehensive risk management framework specifically tailored for agile software projects. Align the framework with agile principles, emphasizing iterative development, flexibility, and collaboration.

Develop a contextualize risk management strategies with analyze contextual factors (e.g., project size, team composition, client engagement) to adapt risk management strategies. Provide practical guidelines that resonate with the unique challenges faced by agile teams in smaller organizations.

5.2 Specific Objectives

1. Risk Identification Enhancement:

    Explore innovative techniques for early risk identification within agile projects.

    Address risks related to changing requirements, evolving technology, and client expectations.

5. Mitigation Strategies Customization:

    Tailor risk mitigation strategies to agile contexts.

    Consider adaptive planning, continuous feedback, and incremental delivery.

6. Decision Support System Implementation:

Develop a real-time decision support system.

Integrate risk insights seamlessly into agile ceremonies (e.g., sprint planning, daily stand-ups).

7. Validation and Evaluation:

Validate the proposed framework and tools through case studies or simulations.

Evaluate their effectiveness in improving project outcomes and reducing risks.

These objectives will guide our research efforts toward enhancing risk management practices in agile software projects within low to mid-level companies.

6. METHODOLOGY

The objective as mentioned of this component is to identify risks and provide mitigation strategies for them. To achieve this first the project details are extracted from the project form that filled by project manager. Then analyze it with the historical data. Then system will identify whether there is risk assigned with project or not. The analysing model will be trained using a dataset of different software projects that have done in different domains and the risks and mitigation strategies assign for them. Then after identify those risks system will provide mitigation strategies for them.

6.1 System Architecture

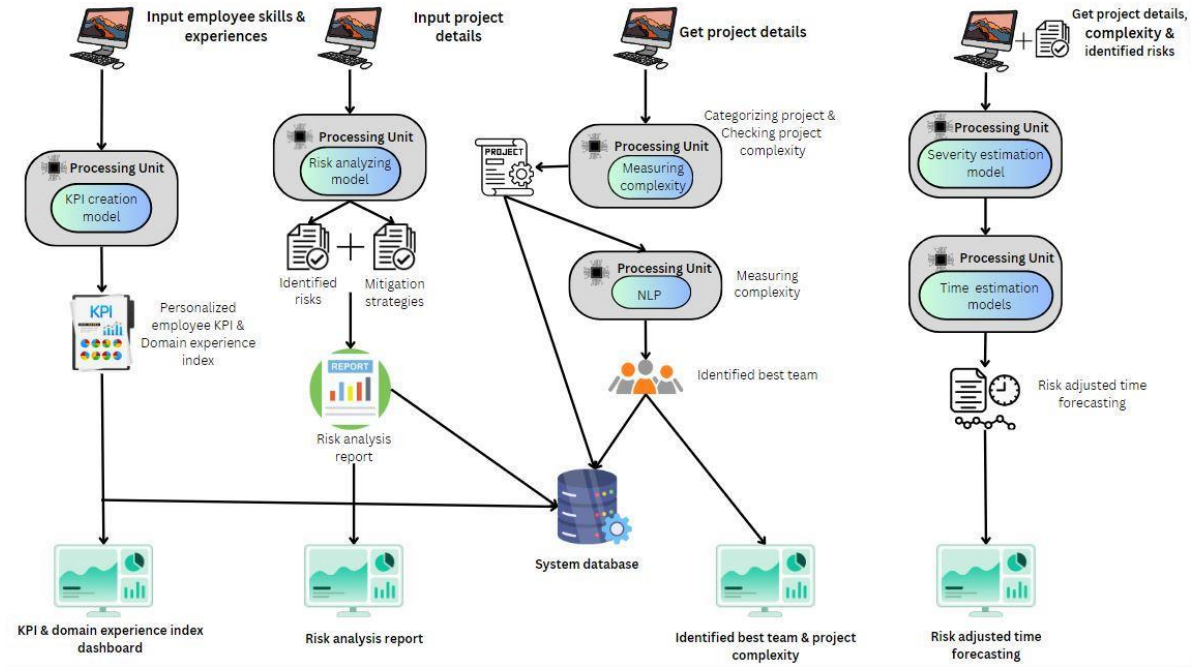


Figure 6-1System Diagram

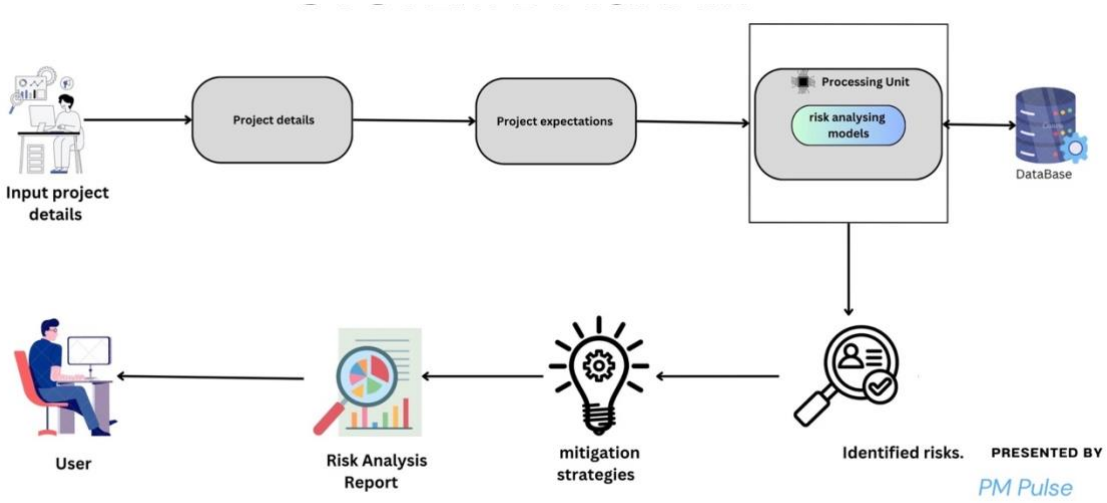


Figure 6.2Flow of the component – Risk identification and mitigation

6.2 Software solution

We are using the System Development Life Cycle (SDLC) technique, which is a wellknown and well-established development strategy. This method should be used because this is a research project. As a result, everything must be created step by step in a straightforward manner, with each step building on the one before it. Our research team intends to clearly develop this finished product, which consists of four outstanding components. The SDLC's primary steps are listed below.

Figure 6-3Stages of Software development life cycle

6.2.1 Requirement Gathering

The SDLC's initial stage is this one. Getting a clear grasp of the entire project is the goal of this stage. It involves planning, requirement identification, and requirement gathering, and all fundamental requirements-related data will be gathered. To collect information on the area, articles which are based on existing researches were referred in order to identify the research problem and to identify a research gap. For data gathering, I started having a database of software projects through a survey.

6.2.2 Feasibility study (Planning)

- **Economic feasibility**

  The benefit and project development cost are covered in the economic feasibility report. When a good economic feasibility plan is in place, the procedure would be successful. As a result, the suggested solution ought to be efficient and affordable.

- **Scheduled feasibility**

  Scheduled feasibility in a feasibility study assesses the timetable (period) for the planned, proposed project and directly affects the project because the project's intent would fail if the deadline is not met.

- **Technical feasibility**

  Technical feasibility is the assessment of the expertise, resources, and skills needed to create the proposed web application, as well as the knowledge of the system architecture and the communication abilities needed to comprehend the demands from the stakeholders to complete the proposed project solution.

6.2.3 Design (system and software design documents)

After the planning phase, which consists of requirement gathering and feasibility study, proposed system and software documents related to system design are created, contributing to the overall system diagram.

6.2.4 Implementation (Development)

The implementation was discussed in the methodology, which consist of the development of below mentioned functionally to satisfy user requirement providing the proposed system with a high level of reliability and accuracy.

- Implementation of the program which identify risks and provide mitigation strategies.
- Implementation of the model which does the analysis to identify risks.

- Implementation of the model which classifies what strategies good to avoid/control/accept those risks.

6.2.5 Testing (Track and Monitor)

At this phase, the developed system will be tested to ensure that it meets all criteria as planned and that there are no system gaps. When testing and ensuring the quality of the generated software, it is also important to check for any system errors and correct any defects that are discovered. To ensure the proposed system's quality, a variety of testing techniques, including component, integration, unit, system, and user acceptability testing, will be used.

6.3 Commercialization and Business plan

Our web application has the significantly increase project management efficiency. It eliminates a large portion of the manual labour associated with traditional project management by automatically matching the best teams to projects based on skill sets, risk assessments, and timeline projections. This is beneficial to any organization that oversees several complex projects.

This application's main target market would be low to medium-sized organizations that manage several projects at once across departments and distant staff. This covers, among other things, consultancy, marketing, IT, Health, Education likewise software companies. These fields obviously need intelligent project management systems.

Create a business-to-business software-as-a-service strategy and collect monthly fees per user. Provide categories of pricing and free trials. To contact clients, form connections with appropriate systems integrators and consulting businesses. Attend conferences and trade exhibitions to demonstrate the product and create interest.

• Using the proposed application, we hope to provide software companies considering each side.

• Market the app to companies, especially in the software/IT services and project management spaces, as a tool to manage their web development projects.

• Offer customized versions and premium support. Using the proposed app as the platform for an agile web project management service.

• Charge clients a fee to manage their projects using the app and your project management methodology. Offer in-person or online training programs that certify people on using the app to manage agile web projects. Charge fees for training.

7. PROJECT REQUIREMENTS

In order to successfully develop and implement our proposed Smart Surveillance System for Risk Identification and Mitigation in Agile Software Projects, we need to define the project requirements. These requirements can be categorized as follows:

7.1 Functional Requirements

Risk Identification Module:
The system must allow agile teams to identify risks early in the project lifecycle.
It should provide mechanisms for brainstorming, historical data analysis, and expert judgment.

Risk Mitigation Strategies Module:
The system should offer context-specific risk mitigation strategies.
Strategies must align with agile practices, such as adaptive planning and continuous feedback.

Customization and Adaptation:
Agile teams should be able to customize risk management approaches based on project context (e.g., team size, client engagement).

7.2 Non-Functional Requirements

Usability and User-Friendliness:

The system interface should be intuitive and easy to navigate.
Agile practitioners with varying technical backgrounds should find it accessible.
Performance and Responsiveness:
The decision support system must provide real-time insights without significant delays.
It should handle concurrent user interactions efficiently.

Security and Privacy:
The system should protect sensitive risk data.
Access controls and encryption mechanisms are essential.

7.3 System Requirements

Technology Stack:
The system will be developed using modern web technologies (e.g., React, Node.js).
Database management systems (e.g., PostgreSQL) will store risk-related data.
Scalability:
The system should accommodate growing project teams and increasing data volumes.
Scalability considerations should be part of the architecture.
Integration with Existing Tools:
The system may need to integrate with existing project management tools (e.g., Jira, Trello).
APIs or connectors should facilitate seamless data exchange.
Documentation and Training:
Comprehensive documentation should guide users on system usage.
Training sessions may be necessary for agile teams during adoption.
By addressing these requirements, we aim to create a robust and effective system that enhances risk management practices in agile software projects.

8. GANTT CHART

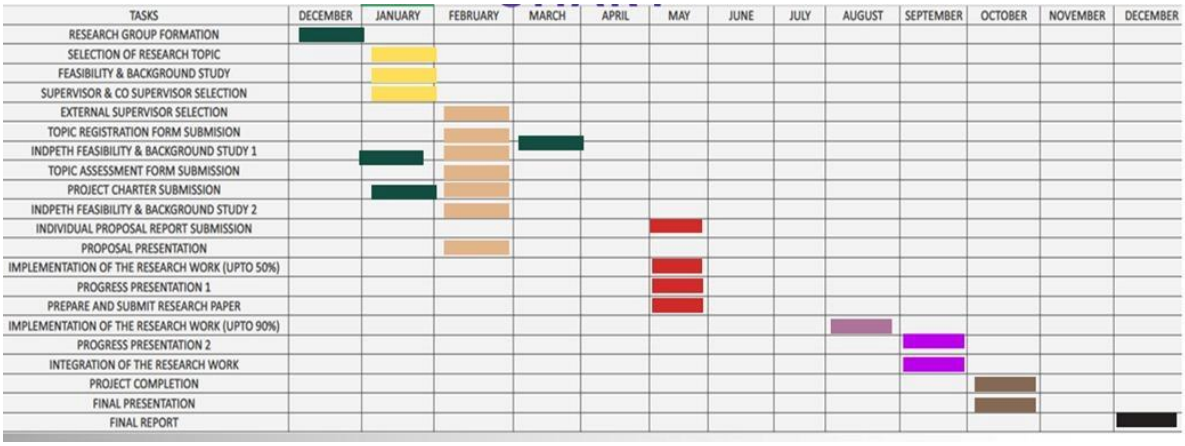


Figure 8-1Gantt chart

8.1 Work Breakdown Structure (WBS)

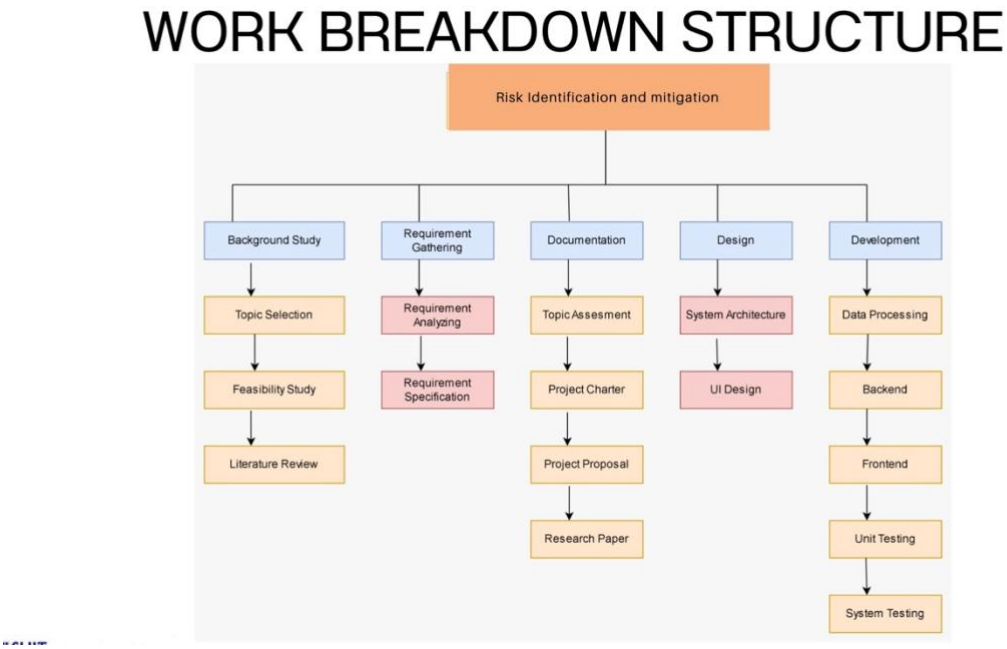


Figure 8-2Work Breakdown Chart of detection of Hate speech in Sinhala YouTube videos

9. BUDGET AND BUDGET JUSTIFICATION

The below table 9.1 shows the overall budget for the overall proposed system

Table 9.1Expenses for the proposed system

		Unit Cost	Total Cost
Cloud Price	*3	6000	18,000
Developers' value	*4	50,000	200,000
Database Price		5000	5000
AWS		7000	7000
Marketing and Advertisements		10,000	10,000
Total Value			240,000

10. REFERENCES

[1] - Liu, J., Chen, C., and Chan, C. (2021). Intelligent Risk Assessment Framework for Software Projects Using Machine Learning Techniques. Journal of Software Risk Management, 27(8), 558-580.

[2] - Hertzfeld, A.R. and Weiss, L.A. (2020). A Software Project Knowledge Graph Representation for AI-Driven Risk Mitigation. Proceedings of the International Conference on Software Risk Management (SCRIM 2020). 10-19.

[3] - Lee, J.Y., Dwivedi, A., and Kim, S. (2018). Dynamic Risk Management Platform for Software Project Health Monitoring. Software Quality Journal 26(3), 1077–1106.

[4] "Risk Identification and Risk Mitigation Instruments for Global Software Development: Systematic Review and Survey Results," *2011 IEEE Sixth International Conference on Global Software Engineering Workshop*

[5] "Risk Identification, Mitigation and Avoidance Model for Handling Software Risk," *2010 2nd International Conference on Computational Intelligence, Communication Systems and Networks*

[6] "Risk-Based Testing: Identifying, Assessing, Mitigating & Managing Risks Efficiently in Software Testing," *SSRN*