

PROJECT PROPOSAL
GROUP PROJECT
CST 392-2

Group No: CST 12

Project and Research Repository System for
Uva Wellassa University of Sri Lanka

Computer Science and Technology

Department of Computer Science and Informatics

Faculty of Applied Sciences

Uva Wellassa University of Sri Lanka

2025

DECLARATION

We hereby declare that the project will be developed by us and will be our own effort and that no part will be plagiarized without citations under the supervision of Dr. H.M.S.N. Ariyadasa, subhash@uwu.ac.lk. This Project Proposal is submitted for the partial fulfillment of the requirement of the course unit CST394-2, Project II for the degree of Computer Science and Technology Degree Program.

Group Details:

Group No: CST 12

No	Name of the Student	Index Number	E-mail address	Signature
1	R.K.C. Priyadarshana	UWU/CST/21/034	cst21034@std.uwu.ac.lk	
2	P.S.H. Gunasekara	UWU/CST/21/044	cst21044@std.uwu.ac.lk	
3	I.C. Dewamiththa	UWU/CST/21/038	cst21038@std.uwu.ac.lk	
4	C.C.R. Jayawardana	UWU/CST/21/060	cst21060@std.uwu.ac.lk	

Supervisor Details:

Name of the Supervisor	E-mail	Contact Number
Dr. H.M.S.N. Ariyadasa	subhash@uwu.ac.lk	+94 553560090

Name of the Co-supervisor	E-mail	Contact Number
Mr.		

Project Approval:

Name of the Supervisor:

Dr. H.M.S.N. Ariyadasa

Date:

Table Of Contents

Chapter 1: Introduction	1
1. Project title.....	1

2. Project Description	1
3. Background and Motivation	2
4. Problem in Brief	2
5. Proposed Solution	3
6. Project Aim and Objectives	4
Chapter 2: Methodology	5
Agile Methodology	5
Chapter 3: Requirements Identification	7
1. Functional and Non-functional requirements	7
I. Functional Requirements	7
II. Non-functional Requirements	11
2. System requirements	12
I. Hardware Requirements	12
II. Software Requirements	12
III. Technologies	12
3. User Roles	13
4. User Levels	15
Chapter 4: Project Plan (Gantt chart)	16
References	17
Appendixes	18

Chapter 1: Introduction

1. Project title

Project and Research Repository System for Uva Wellassa University of Sri Lanka

2. Project Description

The ‘Project and Research Repository System for Uva Wellassa University of Sri Lanka’ is a centralized digital platform designed to store, manage, and retrieve undergraduate and postgraduate project and research reports. The system enables students and supervisors to upload, access, modify, and manage academic documents efficiently. It provides essential functionalities such as smart search and filtering powered by Machine Learning, user management tied to the university domain, and a structured approval and feedback mechanism involving supervisors and contributors. By incorporating semantic search capabilities, the system allows users to retrieve relevant documents based on the meaning of their queries, thus enhancing research continuity and collaboration. This platform aims to minimize redundancy, improve project originality, streamline feedback and approval workflows, and create a culture of building upon existing research work within the university.

3. Background and Motivation

At present, Uva Wellassa University lacks a unified repository for storing completed academic projects and research reports. Once students finish their work, these documents often become scattered or inaccessible, making it difficult for new students and supervisors to access and refer to prior work. This gap results in repeated efforts on similar topics, project rejections due to lack of novelty, and general inefficiencies in managing academic output. Additionally, the manual nature of document handling and supervisor feedback creates bottlenecks in the project approval process. Motivated by these challenges, this project proposes a solution that not only centralizes document storage but also introduces modern retrieval techniques using Machine Learning, a structured feedback and approval system, and enhanced user management. Through this platform, the university community will experience improved academic collaboration, reduced redundancy, and a more effective research development cycle.

4. Problem in Brief

The absence of a centralized repository system at Uva Wellassa University gives rise to several interconnected issues affecting academic research and project development.

These problems are outlined below:

1. Fragmentation of Project and Research Reports

Currently, completed projects and research reports are dispersed across various platforms or stored in isolated locations. This fragmentation makes it extremely difficult for students and supervisors to retrieve past work when needed, leading to a lack of continuity in academic research.

2. Repetitive Work and Lack of Originality

Due to the unavailability of previous projects, students often unknowingly select topics that are too similar to earlier works. This redundancy not only wastes effort but also increases the risk of project rejection during evaluations, as originality and novelty are crucial academic requirements.

3. Inefficiency in Document Management

The management of project and research documents largely relies on manual processes, resulting in inefficiencies such as misplaced reports, delays in sharing documents, and challenges in maintaining proper records over time.

4. Difficulty in Supervisor Feedback and Approval

Without a structured digital workflow, obtaining feedback and formal approval from supervisors becomes time-consuming and disorganized. This often leads to communication gaps between students and supervisors, slowing down the project development and finalization process.

5. Limited Accessibility and Exploration of Existing Knowledge

The current system limits students' ability to easily explore and learn from previous research work. This restricts their understanding of existing gaps

and opportunities in research, ultimately affecting the quality and innovation of their own projects.

5. Proposed Solution

To address the current challenges, we propose the development of a Project and Research Repository System for Uva Wellassa University of Sri Lanka. This system will serve as a centralized digital platform to store, manage, and retrieve project and research reports efficiently.

The platform will offer advanced functionalities, including:

1. Project and Research Management

Allowing users to upload, modify, and delete their own documents, with administrative oversight for quality control.

2. Smart Search and Filtering

Using Machine Learning to enable semantic keyword searches and intelligent filtering, improving the relevance and accuracy of search results.

3. User Management

Ensuring secure access by restricting full project views to users with a university domain, while allowing limited previews to guest users.

4. Supervisor and Contributor Interaction

Facilitating easy addition of supervisors and contributors to projects, and enabling them to verify, accept, or reject submissions with notification support.

5. Document Approval and Feedback System

Enabling supervisors to provide structured feedback and approve documents through a streamlined digital signature process.

The system will enhance the accessibility of past projects, minimize redundancy, facilitate better collaboration between students and supervisors, and streamline the overall document management and project approval process within the university.

6. Project Aim and Objectives

Our aim is to design and implement a centralized repository system for Uva Wellassa University of Sri Lanka, enabling efficient management, retrieval, and approval of student projects and research reports while enhancing collaboration, research continuity, and academic innovation.

The objectives of the ‘Project and Research Repository System for Uva Wellassa University of Sri Lanka’ can be defined as follows:

- To develop a centralized digital platform that securely stores and manages student projects and research documents.
- To integrate an advanced smart search functionality utilizing Machine Learning, allowing semantic search and filtering based on project titles, categories, and document content.
- To implement structured user management ensuring secure access control, with verified users enjoying full access and guests limited to previews.
- To provide a supervisor and contributor management feature, enabling verification, feedback, and acceptance or rejection of submitted documents.
- To streamline the feedback and approval process by allowing supervisors to comment on documents and approve them through an in-system digital signature.
- To enhance research continuity and minimize redundancy, promoting the reuse and building upon previous academic works to encourage originality and innovation.

Chapter 2: Methodology

Agile Methodology

Agile Methodology is an iterative and flexible approach to software development that emphasizes continuous improvement, collaboration, customer feedback, and rapid

delivery of functional software. Instead of delivering a complete product at the end, Agile divides the development process into small, manageable units called sprints or iterations, each resulting in a working product increment. Agile encourages frequent reassessment of project goals and priorities, enabling teams to respond quickly to changes and evolving requirements. Core principles of Agile include transparency, adaptability, early and continuous delivery, and a strong focus on individuals and interactions over rigid processes and tools.

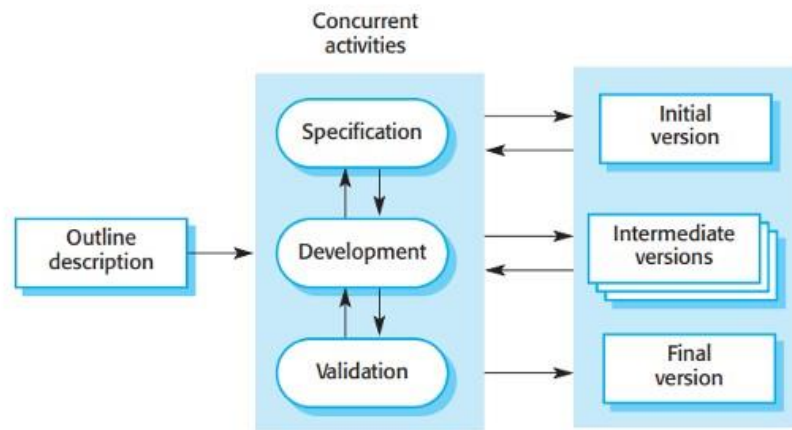


Figure 1

The expense of making changes to requirements is reduced. When compared to the waterfall process, the amount of analysis and documentation that needs to be rebuilt is far less. And also, it is very easy to get customer feedback on the currently developed part of the system and make the changes that customers want. Even if all of the functionality has not been implemented, early delivery and deployment of valuable software to the client are achievable.

Chapter 3: Requirements Identification

1. Functional and Non-functional requirements

I. Functional Requirements

1. User Registration and Authentication

The system must allow users to register and log in using their Uva Wellassa University domain email accounts. This ensures that only authorized individuals, such as students and supervisors, can access full project details while maintaining system security.

2. Upload, Edit, and Delete Projects and Research Reports

Students must be able to upload their projects or research reports, as well as modify or delete them when necessary. This functionality ensures that users have control over the content they contribute to the system.

3. Admin Management of Uploaded Content

Administrators should have the capability to monitor all uploaded documents. They can restrict, delete, or view any project or research report to maintain the integrity and quality of the repository.

4. Smart Search and Filtering

The system will provide an intelligent search mechanism using semantic keyword matching powered by Machine Learning. Users can quickly and accurately retrieve relevant documents based on titles, categories, and content without relying solely on exact keyword matches.

5. Supervisor and Contributor Addition with Notifications

Students must be able to add supervisors and contributors to their projects. When added, these individuals will receive system notifications prompting them to review the submissions and verify or reject them as appropriate.

6. Project Verification by Supervisors

Supervisors are responsible for verifying submitted projects. They can either accept or reject the submissions, ensuring that only quality and original work is officially added to the repository.

7. Feedback System for Supervisors

Supervisors should be able to provide structured feedback on documents through the system. They can comment on necessary corrections or suggest improvements before accepting a project.

8. Digital Approval Mechanism

Once supervisors are satisfied with a project or research document, they can provide digital approval using an in-system signature. This formalizes the acceptance process and provides an official record of document verification.

9. Role-based Access for Different User Types

The system must enforce role-based access control, ensuring that students, supervisors, administrators, and guests have different levels of permissions and access according to their roles.

10. Preview Mode for Guests

Guest users will only have limited access, allowing them to view basic previews (such as project titles and abstracts) without accessing full project documents, thus protecting sensitive academic content.

II. Non-functional Requirements

1. Security

The system must implement strong security measures, including user authentication and secure storage, to protect sensitive academic data and prevent unauthorized access.

2. Performance

The repository must ensure fast response times for searches, uploads, and downloads, providing a smooth and efficient experience even during peak usage.

3. Scalability

As the number of users and documents increases over time, the system must scale accordingly without affecting performance or accessibility.

4. Usability

The system must have an intuitive and user-friendly interface that accommodates both tech-savvy and non-technical users, ensuring that students and supervisors can use it without extensive training.

5. Reliability

The platform must be consistently available, with minimal downtime or system failures, to ensure uninterrupted academic activities.

6. Maintainability

The codebase and infrastructure should be designed for easy maintenance, allowing developers to make updates, bug fixes, and feature additions with minimal disruption.

7. Compatibility

The repository must be accessible across various modern web browsers and devices, including desktops, laptops, and mobile phones, ensuring flexibility for users.

2. System requirements

I. Hardware Requirements

2.1.1. Server Requirements

Processor: Intel Xeon / AMD Ryzen 5 or better

RAM: 16GB or higher

Storage: SSD with minimum 500GB (expandable)

Network: High-speed Internet connection

2.1.2. Client Requirements

Processor: Intel Core i3 or higher

RAM: 4GB or higher

Storage: Minimum 256GB HDD/SSD

Browser: Latest versions of Chrome, Firefox, Edge

II. Software Requirements

2.2.1. Server-Side

Operating System: Ubuntu Server / Windows Server

Web Server: Apache / Nginx

Database: MySQL / PostgreSQL

Backend Environment: Node.js / PHP / Django (depending on selection)

2.2.2. Client-Side

Web Browser: Latest Chrome, Firefox, Edge

Frontend Frameworks: React.js / Angular / Vue.js

2.2.3. Development Tools

- Visual Studio Code Editor

Visual Studio Code is a streamlined code editor that is used for developing various kinds of projects, running tasks, and includes a version control system. It allows the user to build more complex and attractive websites, software, as well as many applications with debugging options.

- Git and GitHub

GitHub is a web-based platform that provides a hosting service for version control using Git. It is primarily used for managing and sharing source code and collaborating on software development projects. GitHub offers a wide range of features and tools

that facilitate collaboration among developers and enable efficient project management.

- Figma

Figma is a collaborative web application for interface design, with additional offline features enabled by desktop applications for macOS and Windows.

- Postman

Postman is a tool used to test and manage APIs easily. It allows developers to send requests, view responses, and check if APIs are working correctly without writing code. Postman helps speed up development, debugging and ensures smooth communication between the frontend and backend systems.

III.

Technologies

Frontend: React.js / HTML5 / CSS3 / JavaScript

Backend: Node.js with Express.js / PHP with Laravel / Django (Python)

Database: MySQL / PostgreSQL

Machine Learning Model: For semantic search (could use TensorFlow.js or a simple custom-trained model)

Authentication: OAuth 2.0 with UWU domain email

APIs: RESTful APIs for communication between client and server

Deployment: Cloud services (AWS, Heroku) or University-hosted server

3. User Roles

1. Student

1. Upload, edit, and delete their own projects/research reports
2. Add supervisors/contributors to their projects
3. View verified projects

2. Supervisor

1. Review projects assigned to them
2. Provide comments and request modifications
3. Accept or reject project submissions
4. Approve documents with an in-system signature

3. Administrator

1. Manage user accounts and permissions
2. Monitor all projects and research reports
3. Delete or restrict projects if necessary
4. Ensure system maintenance and updates

4. Guest User

1. Browse and search for projects
2. View only limited previews without full access

4. User Levels

1. Registered users
 - a. Administrator
 - b. Supervisor
 - c. Registered Students
2. Guest users

Chapter 4: Project Plan (Gantt chart)

[illegible]

References

1. O'Regan, G. 2022. Agile Methodology. In: springer, 247–255.
2. S. Ambler, Agile Modeling: Best Practices for the Unified Process and Extreme Programming. John Wiley & Sons, 2002.
3. S. Newman, Building Microservices: Designing Fine-Grained Systems. O'Reilly Media, 2015.
4. I. Sommerville, Software Engineering, 10th ed. Pearson, 2015.

