A FINAL YEAR PROJECT REPORT

A WEB-BASED FINAL YEAR PROJECT MANAGEMENT SYSTEM

CASE STUDY: BUSITEMA UNIVERSITY

By

KITARA EMMANUEL OKELLO

(BU/UP/2023/1495)

This Final Year Project Report is submitted to the Department of Computer Studies in partial fulfilment of the requirement for the award of the degree of Bachelor of Information Technology of Busitema University.

SUPERVISORS

DR. NANDUTU IRENE

MR. WAMBI FRED

JULY, 2026.

Abstract

The Final Year Project Management System (FYPMS) is a comprehensive web-based application designed to streamline and enhance the management of student final year projects in Busitema University. The system provides a centralised platform that facilitates efficient interaction between students, supervisors and administrators throughout the project lifecycle.

FYPMS enables students to submit project titles, upload proposals and final reports, share GitHub repositories as well as snapshots, and track feedback in real time. Supervisors can review submissions, provide comments, approve and reject milestones, and monitor student progress effectively. Administrators are equipped with tools to manage users, assign supervisors, oversee submissions and analyse system-wide performance through interactive dashboard and visual analytics.

The system integrates modern technologies including Spring Boot for backend development, PostgreSQL for data persistence, and dynamic frontend visualisations using charting libraries. Key features such role-based authentication, deadline management and automated reporting improve transparency, accountability and decision-making.

By digitising and automating traditionally manual processes, FYPMS reduces administrative workload, minimises delays and enhances collaboration. The system ultimately contributes to a more structured, efficient and scalable approach to managing final year academic projects.

Declaration

I, **Kitara Okello Emmanuel,** Reg. No. **BU/UP/2023/1495** do hereby declare that this Project Report is original and hasn't been published or submitted for any other degree award to any other university before.

SIGNATURE:

.....

DATE

22nd July 2026

Approval

This Project Report has been submitted with the approval of the following supervisors.

SIGN:.....

DATE: 08/09/2026

DR. NANDUTU IRENE

Department of Computer Studies

Faculty of Science and Education

Dedication

Firstly, I thank Almighty God for enabling me to complete my project with a sound mind and good health.

I dedicate my report to my dear parents, **Prof. David Kitara Lagoro and Dr. Judith Aloyo Kitara** who didn't fail to give me moral and financial support. Sincere appreciation to **Stacy Martha Alowo Toto** who supported me. Thank you so much for all your support and may good Lord bless you.

I would like to extend special thanks to my supervisors **Dr. Nandutu Irene and Mr. Wambi Fred** who guided me throughout the course of my final year project and that has turned out to be a success.

Lastly, I would like to thank everyone who has helped me throughout my course especially my colleagues, **Wor Joshua, Pande Stephen, Angole Sharif and Kaswabuli Besweri** as they have been of great help and positive influence to me reaching this point in my course.

Acknowledgement

First and foremost, I would like to thank the Almighty God for walking with me and guiding me throughout my studies. Despite all the challenges I have faced, I have taken efforts in this project. However, it would have not been possible without the kind support and help from many individuals. I would like to extend my sincere thanks to all of them.

With utmost gratitude, I would like to thank my supervisors, **Dr. Nandutu Irene and Mr. Wambi Fred** for guiding me through my final year project from the start up to the point of writing the report. They always gave solutions to the problems or challenges I faced and this made everything no matter how hard it was look very easy and doable.

Furthermore, I would like to thank the students, lecturers and administrators of Busitema University for giving me the great hospitality as I was able to create this final year project from information that was provided by them. Such information was crucial and important in the making and coming up of my final year project.

Lastly, I would like to thank all my fellow students whom we've studied with to reach this far especially those with whom we share many course units and classes as they have given me advice on how my final year project would run and any errors that would need to be addressed and have helped my life to run quite smoothly during my stay at Busitema University.

Table of Contents

Contents

List of Figures

List of Acronyms

API	Application Programming Interface
CSS	Cascading Style Sheets
DFD	Data Flow Diagram
EMIS	Education Management Information System
ERD	Entity Relationship Diagram
FYP	Final Year Project
FYPMS	Final Year Project Management System
GB	Gigabyte
HTML	Hypertext Markup Language
HTTP	Hypertext Transfer Protocol
ICT	Information and Communication Technology
JPA	Java Persistence API
JS	JavaScript
JSON	JavaScript Object Notation
JWT	JSON Web Token
MIS	Management Information System
MoES	Ministry of Education and Sports
MVC	Model-View-Controller
RAD	Rapid Application Development
RAM	Random Access Memory
RBAC	Role-Based Access Control
REST	Representational State Transfer
UAT	User Acceptance Testing
UI	User Interface
UPS	Uninterruptible Power Supply
VS Code	Visual Studio Code

1. Chapter One: Introduction

1.1. Background of the Study

Final Year Projects (FYPs) are a core academic requirement in most universities and serve as a capstone for assessing students' ability to apply theoretical knowledge to real-world problems. Through final year projects, students demonstrate research competence, problem-solving skills, innovation, and technical proficiency acquired during the course of study. Effective management of final year projects is therefore essential for maintaining academic quality and institutional accountability.

Traditionally, the management of final year projects in many higher learning institutions has relied on manual and semi-manual approaches such as paper-based proposal submissions, physical logbooks, email-based supervision, and manual allocation of supervisors. In most universities and colleges, students' proposals and reports were collected, processed, and summarised using paper-based portfolios, a process that was time-consuming, inefficient, and disorganised (Suharno and Imraan, 2024). Supervisors often handle multiple students concurrently, progress tracking is done manually, and communication between students and supervisors is inconsistent. These practices frequently lead to delayed feedback, poor documentation, duplication of project topics, loss or misplacement of records, and limited transparency in supervision and assessment (Haruna et al., 2023).

With rapid advancements in Information and Communication Technologies (ICT), information has become more accessible and easier to manage. Educational institutions, as centres of knowledge creation and dissemination, are increasingly expected to adopt automated information systems to enhance academic administration and decision-making (Ahmad, 2024). Information systems enable institutions to efficiently collect, store, process, and disseminate data, thereby improving coordination, accountability, and service delivery (Masruri, 2025). Within the education sector, Educational Management Information Systems (EMIS) have been widely adopted to support academic planning, monitoring, evaluation, and evidence-based decision-making and institutional planning (Kebebe and Regassa, 2024).

Computers and electronic communication technologies play a vital role in ensuring faster flow of information in the rapidly developing world (Geymen *et al.*, 2008). The computerised

environment supports timely access to information, efficient record management, and improved communication among stakeholders (Durkaya and Durkaya, 2003). Final Year Project Management Systems form part of this broader shift towards automation in higher education institutions.

A Final Year Project Management System (FYPMS) is an automated web-based platform designed to manage the entire lifecycle of students' final year projects, from topic proposal and supervisor allocation to progress tracking, assessment, grading, and archival of completed projects. The system facilitates structured supervision, timely feedback, approval of proposals and reports, and secure storage of academic records, while also providing a centralised repository for final year projects and reports. Such systems enhance efficiency, transparency, and effective communication among students, supervisors, and administrators.

Final year project submission is a recurring academic activity in most universities and continues to receive increasing attention as institutions are required to demonstrate accountability, performance, and efficiency to stakeholders. Designing an effective final year project management system involves integrating multiple users, workflows and evaluation mechanisms into a secure digital environment (Sabo et al., 2025).

At Busitema University, the management of final year projects is still largely manual, presenting significant challenges in monitoring student progress, coordinating supervision, analysing submissions, and maintaining accurate project records. Manual paper-based systems are prone to data loss, omission of records, and delays in analysis and decision-making. This situation necessitates the development of an automated Final Year Project Management System to streamline project registration, supervision, evaluation, and report submission, thereby improving efficiency and effectiveness in the management of final year projects.

1.2.Statement of the Problem

The management of final year projects plays a critical role in supporting effective learning, research supervision, and institutional development within universities (Uttl, White and Gonzalez, 2017). However, the absence of an integrated and automated system for managing final year projects at Busitema University has resulted in several operational challenges. These include delays in project proposal approvals, poor communication between students and

supervisors, difficulty in tracking project milestones, and loss or duplication of project documents.

Currently, Busitema University–Nagongera Campus relies on a rudimentary, paper-based approach to manage final year projects. This manual system makes the retrieval, storage, and maintenance of project information difficult and inefficient. Supervisors and administrators are required to handle proposals, progress reports, and final submissions manually, a process that is time-consuming and prone to errors. As a result, analysing project data, monitoring student progress, and generating timely reports for decision-making and quality assurance becomes challenging.

Furthermore, the extensive paperwork involved in submitting proposals, reports, and feedback significantly delays the supervision and evaluation process. The lack of a centralised platform limits transparency, weakens coordination among stakeholders, and hinders effective documentation and archiving of final year projects.

Therefore, there is a need to design and develop an automated, web-based Final Year Project Management System to streamline project registration, supervision, progress tracking, evaluation, and documentation. Such a system will enhance efficiency, improve supervision, ensure proper record management, and support effective administration of final year projects at Busitema University.

1.3.Main Objective

The main goal of the study is to develop and design an automated Final Year Project Management System (FYPMS) that supports efficient management, monitoring, and evaluation of students' final year projects at Busitema University-Nagongera Campus, Department of Computer Studies.

1.4.Specific Objectives

1. To review the literature and determine the requirements of the Final Year Project Management System at Busitema University.
2. To design a web-based Final Year Project Management System tailored to the needs of Busitema University.

3. To implement the web-based Final Year Project Management System using modern technologies and best practices.
4. To test and validate the web-based Final Year Project Management System to ensure it meets the specified requirements and user expectations.

1.5.Significance

Upon the successful development and implementation of the automated web-based Final Year Project Management System, the study is expected to yield the following benefits:

- Improved efficiency: The system will streamline project proposal submission, supervision, progress tracking, and assessment, thereby reduce delays and improve overall project management efficiency.
- Enhanced communication: The system will facilitate effective and timely communication among students, supervisors, and administrators through a centralized platform.
- Centralised and secure data management: It will provide a secure and centralised repository for storing final year project proposals, reports, and related documents, minimizing the risk of data loss or duplication.
- Support for timely decision-making: Automated reporting and tracking features will enable administrators and supervisors to generate real-time reports to support informed decision-making and quality assurance.
- Reduced administrative workload: By minimizing paperwork and manual handling of documents, the system will significantly reduce administrative workload and operational costs.

1.6.Scope of the Study

1.6.1. Content Scope

The content scope of the study focused on the design and development of a web-based Final Year Project Management System (FYPMS) to support the management of final year projects at Busitema University.

The system will support multiple user roles, including students, supervisors, and administrators. Core functionalities of the system will include project topic submission, supervisor allocation, proposal and report submission, progress and milestone tracking, document upload, feedback provision, approval and rejection of submissions, and automated report generation. The system will utilize modern web-based technologies with a centralized database for secure data storage and retrieval. It will also analyse technological solutions that can be integrated into the FYPMS to enhance efficiency, learning outcomes, and user experience.

Administrators will be responsible for system management tasks such as user registration, supervisor allocation, milestone tracking, approvals, and overall system monitoring. Students will be responsible for logging into the system, submitting project proposals and reports, and tracking project progress. Supervisors will review submissions, provide feedback, and approve or reject milestones and reports.

In addition, the study will examine best practices in the implementation of Final Year Project Management Systems from other educational institutions and organisations. Particular attention will be given to the effectiveness of existing FYPMS tools, identification of gaps in user engagement, and the potential benefits of a centralized platform for managing learning and assessment processes.

1.6.2. Area Scope

The area scope defines the geographical boundary within which the study will be conducted and from which data will be collected for the system design and development.

This study was carried out at Busitema University-Nagongera Campus located at Nagongera Town Council in Tororo District about 17 kilometres from Tororo town along Tororo-Busolwe road.

2. Chapter Two: Literature Review

2.6. Introduction

This chapter presents a critical review of existing literature related to information systems, management information systems, educational management information systems, and project management systems within higher education. The purpose of this review is to establish a theoretical and empirical foundation for the design and development of the proposed Final Year Project Management System (FYPMS). The chapter also examines challenges associated with manual project management approaches and highlights the benefits of automated systems in managing final year projects.

2.7. Information Systems

An information system is a combination of people, hardware, software, data, and procedures that interact to collect, process, store, and disseminate information in order to support decision-making, coordination, and control within an organization (Masruri, 2025). Information systems are essential in improving organizational efficiency, reducing operational costs, and enhancing accuracy in data handling.

In higher education institutions, information systems are widely used to manage academic records, student information, course administration, and institutional reporting. The adoption of information systems enables universities to automate routine processes, improve access to information, and support evidence-based decision-making (Haruna et al., 2023). The proposed Final Year Project Management System is grounded in information system principles, as it seeks to integrate people, processes, and technology to manage final year project activities efficiently.

2.8. Management Information Systems

Management Information Systems (MIS) are a subset of information systems designed to provide managers with timely, accurate, and relevant information to support planning, monitoring, and decision-making (Ahmad, 2024). MIS typically generate structured reports based on data collected from transaction processing systems.

In academic institutions, MIS play a significant role in monitoring student performance, managing academic activities, allocating resources, and evaluating institutional performance. MIS enable administrators to track academic progress, identify challenges, and make informed decisions that improve institutional effectiveness. A Final Year Project Management System can be considered a specialized academic MIS, as it supports the management and monitoring of student research projects, supervision activities, and assessment outcomes.

2.9. Educational Management Information Systems

Educational Management Information Systems (EMIS) support planning, monitoring and evaluation in educational institutions. Research shows that EMIS implementation improves accountability, transparency and institutional coordination (Kebebe and Regassa, 2024). Furthermore, Masruri(2025) argues that EMIS frameworks optimise learning processes by enabling efficient data collection analysis.

In Uganda, the Ministry of Education and Sports (MoES) utilizes EMIS to facilitate evidence-based decision-making, policy formulation, and monitoring of educational performance at various levels (MoES, n.d.). EMIS have improved data accessibility, accountability, and transparency in the education sector.

Final Year Project Management Systems can be viewed as components or extensions of EMIS, as they focus on managing a specific but critical academic process—final year projects. By integrating FYPMS into institutional information systems, universities can enhance monitoring of student research activities, improve supervision quality, and maintain reliable academic records.

2.10. Projects Management Systems in Higher Education

Project Management Systems provide tools for planning, scheduling, monitoring, and controlling project activities. In higher education, automated project management systems have increasingly been adopted to manage student research projects, including final year projects (Suharno and Imraan, 2024). These systems facilitate structured supervision, timely feedback, and systematic documentation of project activities.

A Final Year Project Management System typically supports functionalities such as project topic submission, supervisor allocation, milestone tracking, proposal and report submission, feedback provision, grading, and archival of completed projects. Such systems enhance transparency, accountability, and collaboration among students, supervisors, and administrators.

According to Haruna et al., (2023), web-based academic management systems significantly improve communication and reduce administrative delays by providing a centralized platform for interaction and documentation. This makes automated FYPMS particularly suitable for institutions with large student populations and limited administrative resources.

2.11. Challenges of the Manual Project Management

Despite the importance of final year projects, many universities continue to rely on manual or paper-based systems to manage project activities. Manual project management systems are associated with poor documentation, delayed feedback and inefficient record retrieval (Ahmad, 2024). The absence of secure digital workflows may also lead to data inconsistencies and limited accountability (Sabo et al., 2025).

Paper-based systems are prone to data loss, duplication of project topics, inconsistency in documentation, and difficulty in retrieving historical records. Supervisors handling multiple students often struggle to track individual progress effectively, while administrators face challenges in generating timely reports for quality assurance and decision-making. These inefficiencies negatively affect the quality of supervision and overall student learning experience.

2.12. Benefits of an Automated Final Year Project Management System

Automated Final Year Project Management Systems offer numerous advantages over manual approaches. These include improved accuracy of records, real-time access to project information, enhanced communication among stakeholders, secure storage of documents, and efficient reporting mechanisms (Alruwais, Wills and Wald, 2018).

Electronic systems allow students and supervisors to interact through a centralized platform, reducing delays and improving transparency. Automated tracking of milestones and submissions ensures timely completion of projects and enables administrators to monitor progress effectively. Studies indicate that students and academic staff prefer electronic systems due to their convenience, flexibility, and reliability (Alruwais *et al.*, 2018).

By adopting an automated FYPMS, universities can improve supervision quality, reduce administrative burden, and enhance the overall effectiveness of final year project management.

3. Chapter Three: Methodology

3.6. Introduction

This chapter outlines the methodology that was employed in the development of the web-based FYPMS for Busitema University-Nagongera Campus, specifically within the Department of Computer Studies. The methodology detailed the research design, study population, sampling approach, and data collection methods. It also highlighted the system development process following the Rapid Application Development (RAD) model, which was structured into four key phases: requirements determination, design, implementation and testing. Each phase was designed to incorporate stakeholder feedback and ensure the system met the real needs of the university environment. Furthermore, this chapter covered the approach used for data analysis, ethical considerations upheld during research, and the limitations encountered throughout the study.

3.7. Research Design

The research employed a qualitative approach with the Rapid Application Development (RAD) framework. This design was chosen due to its ability to integrate academic inquiry with practical system implementation. The RAD methodology facilitated continuous stakeholder involvement through iterative cycles, ensuring that user feedback guided the development process. The process was divided into four progressive stages: The Requirement Determination Phase, which established the foundational needs and scope of the system; The User Design, which converted the requirements into tangible prototypes; The Implementation Phase, which involved the actual development of system components; and The Testing Phase, where the system was evaluated to ensure it met the intended functionality before deployment.

3.8. Area of Study

The study was conducted at Busitema University, situated along the Tororo-Busolwe road in Nagongera Town Council, 17 kilometres from Tororo Town in Tororo District, Eastern Uganda. The institution was selected due to its strong commitment to academic excellence and innovation, as well as its diverse student population drawn from multiple regions.

3.9. Population and Sampling

The study population comprised students, supervisors and an administrator from the Department of Computer Studies at Busitema University, Nagongera Campus. This population was selected due to its direct involvement with final year projects, including final year grading and validation, making them suitable participants for the research and system development. The population included system users such as final year computer students, lecturers and administrators , each playing a key role in the daily management and usage of the final year projects.

A population of 15 people were chosen but a sample of 9 were purposively selected based on their direct involvement in the final year project management system of submitting and approving final year projects.

Out of the 9 participants chosen, 2 were lecturers (supervisors), 1 administrator and 6 students who were finalists ready to submit their final year projects.

3.9.2. Sampling Approach

The sampling approach employed in this study was no-probability sampling, specifically purposive sampling. This method was chosen because it allowed the researcher to use their judgement and expertise to select participants who were most relevant to the objectives of the research. Individuals selected were those who had prior information with the management of final year projects at the Department of Computer Studies, Busitema University-Nagongera Campus.

Purposive sampling enabled the collection of in-depth, context-specific insights from supervisors, administrators and students who directly prepared and submitted final year projects. Their informed perspectives, and usability expectations, which were essential for the successful development and validation of the proposed web-based Final Year Project Management System.

3.10. Data Collection Methods

In order to understand the operational context, challenges, and limitations of the current final year projects management practices at Busitema University-Nagongera Campus, Department of Computer Studies, the researcher employed a triangulated data collection approach. The methods included observation, document review, and semi-structured interviews, all tailored

to gather qualitative insights essential for system design and development under the RAD methodology.

3.10.2. Observation

Direct observation was conducted within the Department of Computer Studies to understand the practical workflow involved in submitting and approving final year projects using an observation checklist (see Appendix A). This non-intrusive method allowed the researcher to witness real-time processes without altering natural behaviours. The focus was on the use of manual logbooks, interaction patterns between students and supervisors, and how final year projects were updated, stored and accessed. Observational data revealed inefficiencies such as time delays in record retrieval, inconsistencies in data entry, and reliance on individual memory during project submission.

3.10.3. Document Review

The researcher reviewed various physical records used by the department, including project register books, submission books and maintenance books with reference to guiding questions (see Appendix B). These documents provided historical data on how final year projects have been managed over time. Through this review, it was evident that manual documentation methods lack data validation, version control, and backup mechanisms. In some cases, torn pages and faded ink hindered the ability to retrieve full records. This process emphasised the need for a digital system capable of secure, real-time updates with historical logging capabilities.

3.10.4. Interviews

Semi-structured interviews were conducted with key stakeholders including students, administrators and supervisors with guiding questions (see Appendix C). These interviews were designed to collect first-hand experiential insights regarding current final year project management practices, with particular focus on the use of books/manual systems

The questions focused on how supervisors, students and administrators interact when submitting and approval of final year projects and what they expected from the web-based system.

Findings

The researcher employed the above data collections methods to gather the necessary information that formed the requirements for the web-based Final Year Project Management System.

The key findings are summarized below;

- Manual and error-prone processes. Supervisors currently use physical logbooks, which are prone to errors, illegible handwriting, and difficult updates.
- Scattered and inconsistent records. Multiple paper registers lead to duplication, missing entries, and lack of a single source of truth.
- Difficulty tracking project submissions. Supervisors cannot easily monitor all project submissions due to the absence of reminders or alerts.
- Security and accessibility concerns. Physical records are vulnerable to loss, theft, or damage, and are slow to retrieve when needed.
- Slow approval process. Supervisors and students experience delays as supervisors manually check and update logbooks.
- Demand for transparency and automation. Respondents expressed a desire for clear workflows, automated approvals, and reliable digital tracking of project submissions and resources.
- Accountability issues. Some students have been wrongly recorded or forgotten in logbooks, causing disputes.

3.11. System Development Process

The development of the proposed web-based FYPMS followed the RAD methodology, which emphasises quick and iterative delivery of functional software with continuous user involvement. RAD was selected due to its adaptability, speed, and strong emphasis on prototyping and user feedback—making it ideal for developing a system tailored to departmental needs within a constrained academic timeline.

The RAD approach adopted in this study comprised four key phases;

3.11.2. Requirement Planning Phase

This initial phase laid the foundation for the entire system. It involved stakeholder engagement, data collection, and the modelling of functional and non-functional requirements.

Data collection

As outlined in section 3.5, data was gathered through observation, document reviews, and interviews with students, supervisors and administrators. This ensured a grounded understanding of the system context and challenges.

Requirements elicitation

Through interaction with stakeholders, key system requirements were identified based on the data collected. These included;

- Tracking of project submissions and resources.
- User authentication. Secure login for users.
- Role-based access for different users (admin, supervisors and students).
- Real-time status updates. Updates on submissions.

Requirement modelling

The requirements were structured into logical models using Entity-Relationship Diagrams (ERDs) to show the system's interaction with external entities, Data Flow Diagrams (DFDs) to illustrate data movement within the system, and use case scenarios, to represent user interactions with the system. These models provided a blueprint for the user interface, database structure, and system workflows.

3.11.3. User Design Phase

In this phase, the system began to take visual and structural form. User input was heavily relied upon to guide interface layout and system interactions.

Low-fidelity design: basic wireframes and sketches were created to visualise the user interface. This design focused on navigation flow, form structures and essential components like submitting proposals and final reports, status update on submission and analytics charts.

Usability testing: early prototypes were presented to students, supervisors and administrators for usability feedback. Suggestions such as form layout changes, colour preferences, and sidebar menu structuring were incorporated to enhance user experience. This step ensured that the design aligned with the actual needs and habits of intended users.

3.11.4. Implementation Phase

This stage involved actual system development based on the approved design models.

Front-end implementation:

The user interface was built using HTML5, CSS3, and JavaScript, ensuring a responsive, intuitive, and lightweight design. Tailored dashboards were created for different user roles (admin, supervisor, student).

- User login page for students, supervisors and admin.
- Project submission for students.
- Project preview and approval for supervisors.
- Dashboards displaying number of users, proposals and final reports.

Back-end implementation

The core logic and functionalities were developed using Spring Boot, a Java framework connected to a PostgreSQL database. This included modules for;

- Login authentication
- Project submission and management
- Submission tracking
- Grading and assessment

3.11.5. Testing Phase

The system underwent rigorous testing to ensure reliability, accuracy and usability prior to deployment.

Unit testing: Individual modules (e.g., login, submitting form, grading interface) were tested in isolation for functionality and correctness.

Integration Testing: the entire system was tested as a whole to verify that modules worked together smoothly, with attention to data integrity across transactions.

User Acceptance Testing (UAT): The prototype was deployed in a controlled environment where actual users from the department performed real-world tasks. Their feedback was recorded and used to refine final features and workflows.

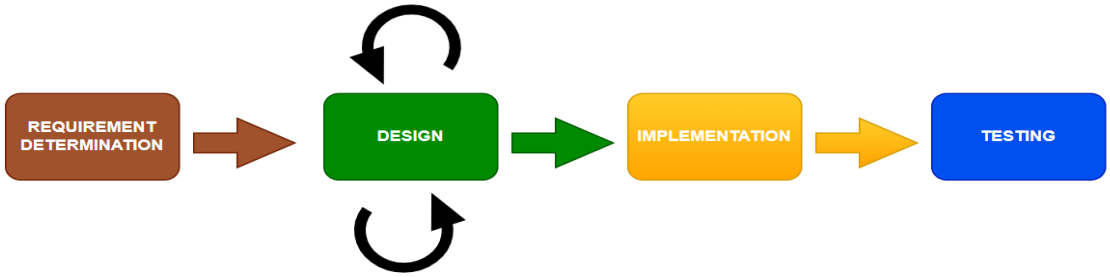


Figure 1: The Rapid Application Development methodology.

3.12. Data Analysis Approach

The qualitative data collected was analysed using thematic analysis. This process involved carefully reviewing interviews transcripts and observational notes to identify recurring patterns and key insights.

Lack of structure in final year project management. Current methods of submitting and acquiring comments for final year projects were manual and prone to errors, resulting in inconsistent, scattered, and sometimes duplicated records.

Inefficient and slow submission processes. Many Staff and students faced delays due to manual checks in books, causing slow approvals and sometimes wrong records of submissions.

Need for real-time tracking and accountability. Many respondents emphasised the importance of seeing project approval, maintaining clear project submission histories, and ensuring accurate accountability for each user.

These themes directly informed the design of the Web-Based Final Year Project Management System, guiding the inclusion of key features such as a centralized final year project database, real-time accountability dashboard, automated approval workflows, borrowing submission histories, secure role-based access, and notification systems for deadlines.

3.13. Ethical Considerations

In conducting this study, ethical principles were carefully observed to ensure integrity, confidentiality, and respect for all participants involved in the research process.

1. Informed Consent

Before any data was collected, the researcher obtained informed consent from all participants. Each participant was made aware of the purpose of the study, how their data would be used, and their right to withdraw from the study at any time without any consequences.

2. Confidentiality and Privacy

All information collected through interviews and document reviews was treated with strict confidentiality. Personal identifiers such as names or staff numbers were omitted or anonymized in order to protect the identity of participants. Data was stored securely and only used for academic purposes relevant to the system development.

3. Voluntary Participation

Participation in interviews and observations was entirely voluntary. No individual was coerced or compelled to provide information. The study ensured that participants were comfortable and aware that their input would directly inform improvements in the final year project management process.

4. Responsible Data Handling

Data collected from internal documents and observation of the university's final year projects records was handled responsibly and used solely to inform system requirements and validate the problem statement. No sensitive or institutional data was disclosed outside the research scope.

5. Academic Integrity

Throughout the development and documentation phases, the research adhered to principles of academic honesty, ensuring that all ideas, systems, and data sources were appropriately cited and attributed.

3.14. Limitations of the Study

While the research and system development were conducted rigorously, several limitations were encountered that may have influenced the depth and scope of the outcomes.

1. Time Constraints

The project was conducted within a limited academic timeframe, which restricted the duration available for in-depth user testing, long-term observation, and extended iterations of the system development phases.

2. Limited Sample Size

Due to availability and scheduling challenges, the number of interview participants was limited to a small subset of the ICT department staff. This could have influenced the diversity of feedback and generalizability of the findings.

3. Access to Historical Data

Accessing comprehensive historical records of final year projects was difficult due to incomplete documentation and inconsistent archival practices. This limited the system's ability to use past data for predictive analysis or trend modeling.

4. Chapter Four: Field Study Results and System Design

4.6. Introduction

This chapter presents the results of the system analysis and design activities carried out in the development of the Final Year Project Management System for Busitema University, Nagongera Campus, Department of Computer Studies. The chapter discusses the findings from the field study, which included interviews, observations and document reviews, followed by a structured interpretation and synthesis of the gathered data. These findings formed the functional and non-functional system requirements, which guided the modelling and design processes.

The chapter outlines the functional and non-functional requirements elicited from findings. The design section focuses on the transformation of these requirements into formal models using Entity Relationship Diagrams(ERDs), Data Flow Diagrams(DFDs), and process descriptions.

4.7. Current System Study

At Busitema University, specifically Nagongera Campus, a detailed study was carried out to understand how final year projects are currently managed using a paper-based system. The purpose of this study was to identify both the challenges and the strengths of the existing system in order to guide the development of a better digital solution.

The study mainly used interviews to collect information from students, supervisors, and administrators. This helped in understanding how the system works in real-life situations.

In the current system, most activities related to final year projects are done manually using paper. Students prepare and submit their project documents such as proposals and reports in printed form. The supervisor then reviews these documents by writing comments and feedback on paper.

After that, administrators collect all submitted documents and records. They are responsible for organising, storing, and reviewing them. This includes checking project progress, recording feedback, and managing final evaluations. All these tasks are manually done, which requires a lot of time and effort.

Although this system allows direct interaction between students and supervisors, it has several limitations. Documents can easily get lost or damaged, feedback may be delayed, and tracking

students' progress becomes difficult. In addition, storing large amounts of paper records is inefficient and hard to manage.

Despite these challenges, the existing system has some strengths. It is simple to use, does not require technical skills, and allows personal interaction between users. However, it is not suitable for handling large numbers of students and projects in a modern academic environment.

The findings from this study highlight the need for a more efficient and reliable system. As a result, the proposed Final Year Project Management System (FYPMS) aims to replace manual processes with a digital platform. The new system will allow online submission of documents, real-time feedback from supervisors, easy tracking of progress, and secure storage of records.

In conclusion, the study shows a clear transition from a traditional paper-based system to a modern digital system. This transformation is necessary to improve efficiency, accuracy, and overall project management within the university.

4.8.Strength of the Current System

At Busitema University, particularly Nagongera Campus, the current paper-based final year management system has played an important role over the years. Although it is traditional, it has several strengths that have made it reliable and widely used within the institution.

One major strength of the system is its **inclusivity**. The paper-based approach allows all students and supervisors to participate without needing any digital skills or access to technology. This is especially important in environments where some users may not have access to computers, smartphones, or stable internet connections. As a result, every student is able to submit their work and receive feedback without barriers.

Another strength of the system is its **resistance to technical failures**. Unlike digital systems, paper-based records are not affected by issues such as system crashes, server downtime, or cyber-attacks. This makes the system dependable, as information can always be accessed physically without worrying about technical disruptions.

The system also benefits from the **physical nature of records**. Printed documents such as proposals, reports, and supervisor comments can be stored and easily referenced when needed.

These physical files serve as proof of student work and feedback, and they can be archived for future use. This creates a sense of performance and reliability in record keeping.

In addition, the system promotes **direct interaction** between students and supervisors. Since much of the process is done physically, it encourages face-to-face communication, which can improve understanding and clarity in feedback. This human interaction is an important part of the academic experience.

Lastly, the paper-based system operates **independently of technology.** It does not require internet access, software systems, or electronic devices to function. This simplicity makes it easy to use and maintain, especially in situations where technological infrastructure may be limited.

In conclusion, while the current system has limitations, it remains valuable due to its simplicity, reliability, and inclusiveness. These strengths highlight why it has been sustained over time, even as the need for a more advanced digital solution continues to grow.

4.9. Weakness of the Current System

At Busitema University, particularly Nagongera Campus, the current paper-based final year project management system, despite its strengths, has several weaknesses that affect its efficiency and effectiveness.

One major weakness is the issue of **delays in processing information**. Students submit their project documents at different times, and supervisors also take time to review them manually. This results in slow feedback and delays in decision-making. Since there is no centralized system, tracking progress becomes difficult and time-consuming.

Another weakness is **administrative inefficiency**. Administrators are required to manually collect, organize, and review large amounts of paperwork, including proposals, reports, and feedback. This process consumes a lot of time and effort, making it difficult to manage multiple students effectively. It also increases the chances of human error during record handling.

The system also faces challenges related to **storage space**. As the number of students increases, the volume of physical documents grows significantly. Storing these documents requires large physical space, which becomes difficult to manage over time. Retrieving specific records can also be slow and inconvenient.

A critical weakness is the **risk of document loss or damage**. Paper documents can easily be misplaced, lost, or damaged due to poor handling or environmental factors such as moisture or fire. This can lead to loss of important academic records and affect the integrity of the project evaluation process.

In addition, the system lacks **real-time access to information**. Students and supervisors cannot easily track progress or access documents remotely. This limits communication and slows down collaboration between stakeholders.

Finally, the system is **not scalable**. As the number of students and projects increases, the manual processes become more difficult to manage. This makes the system inefficient for modern academic environments where large volumes of data need to be handled quickly and accurately.

In conclusion, the weaknesses of the current paper-based system highlight the need for a more efficient and reliable solution. These challenges, including delays, administrative burden, storage limitations, and risk of data loss, make it necessary to adopt a digital system.

4.10. System Analysis Findings/ Field Study Results

The system analysis findings presented in this section were derived through qualitative data collection methods as outlined in the research methodology (Chapter 3). These methods included semi-structured interviews, direct observations and document review, all aimed at gathering in-depth insights into the existing final year project management processes at Busitema University-Nagongera Campus, Department of Computer Studies.

The field study provided critical information about the operational inefficiencies, procedural gaps, and user expectations regarding the management, tracking and submitting of final year projects. Each method contributed uniquely to understanding the current system.

4.10.2. Functional Requirements

The functional requirements of the Final Year Project Management System (FYPMS) define the key services that the system provides to its users. These requirements focus on supporting the interactions between students, supervisors, and administrators, ensuring that the system meets their needs efficiently.

At Busitema University, especially Nagongera Campus, the system is designed to improve how final year projects are managed by replacing manual processes with a digital platform.

Student Functional Requirements

The student experience begins with **authentication**. Students log into the system using their email and password. Once successfully logged in, they gain access to a personalized dashboard.

From the dashboard, students are able to:

- View their assigned supervisor
- Submit project components such as:
    - Project title
    - Proposal document
    - Final report
    - GitHub repository link
    - Project snapshots/images
- Track the progress of their project
- View feedback and comments from their supervisor
- Monitor approval or rejection status of submissions

This design allows students to actively participate in managing their projects while receiving timely feedback from supervisors.

Supervisor Functional Requirements

Supervisors interact with the system through a dedicated dashboard that allows them to manage assigned students.

Supervisors are able to:

- View a list of assigned students
- Access student submissions (title, proposal, report, GitHub, snapshots)
- Approve or reject submissions

- Provide comments and feedback on each submission
- Monitor student progress through analytics
- Assign and manage grades based on:
    - Proposal
    - Progress
    - Final report
    - Presentation
- Upload and update their profile information

This ensures that supervisors can effectively guide and evaluate students throughout the project lifecycle.

Administrator Functional Requirements

Administrators oversee the overall system and ensure smooth operation. They are able to:

- Register and manage users (students and supervisors)
- Assign students to supervisors
- Manage deadlines for project activities
- Monitor overall system activity and performance
- Access reports and analytics on student progress

These capabilities help administrators maintain control and ensure that the project process runs efficiently.

System Functional Requirements

The system also provides core functionalities that support all users:

- Secure login and authentication using JWT
- File upload, storage, and retrieval
- Real-time feedback and comment system
- Progress tracking and analytics visualization

- Role-based access control (Student, Supervisor, Admin)

The functional requirements of the FYPMS are designed to create a smooth and efficient workflow between students, supervisors, and administrators. The system promotes transparency, accountability, and active participation in the management of final year projects.

By digitising key processes such as submission, feedback, and grading, the system improves efficiency and reduces the limitations of the traditional paper-based approach

4.10.3. Non-functional requirements

When designing the Final Year Project Management System (FYPMS), the focus goes beyond just what the system does. It also considers how well the system performs and how users experience it. These are defined as non-functional requirements, and they play a key role in ensuring the system is efficient, reliable, and easy to use.

At Busitema University, particularly Nagongera Campus, the system is designed with important qualities such as user-friendliness, performance, and data security.

User-Friendliness

User-friendliness is a key requirement of the system. The application is designed to be simple and easy to use for all users, including students, supervisors, and administrators.

The interface includes:

- Clear navigation menus
- Simple and understandable language
- Well-organized dashboards
- Easy-to-use forms for submission and feedback

This ensures that users can interact with the system without needing advanced technical skills, making the overall experience smooth and efficient.

Performance

Performance is another important requirement of the system. The FYPMS is a web-based application that depends on an internet connection to communicate with the backend server and database.

- With a **fast internet connection**, the system performs efficiently, allowing quick loading of pages, fast file uploads, and smooth interaction.
- With a **slow connection**, the system still functions but may respond more slowly.

The system is designed to handle multiple users at the same time while maintaining stable performance.

Data Integrity and Security

Data integrity is a critical requirement in the system. The FYPMS ensures that all data related to students, submissions, comments, and grades is stored securely.

- Data is stored in a structured database (PostgreSQL)
- Access is controlled using authentication and authorization (JWT)
- Only authorized users can view or modify specific data

The system minimizes the risk of data loss by ensuring that records are safely stored and only accessible by permitted users. Data can only be altered or removed through authorized actions, which helps maintain accuracy and reliability.

Reliability

The system is designed to be reliable and consistently available for users. It ensures that:

- Users can access the system whenever needed
- Operations such as submissions and grading are completed successfully
- Errors are handled properly without crashing the system

Scalability

The FYPMS is built to handle an increasing number of users and data over time. As more students and supervisors use the system, it can scale without affecting performance.

Security

Security is enforced through:

- Secure login authentication
- Role-based access control (Student, Supervisor, Admin)
- Protection of sensitive data such as grades and submissions

This ensures that the system is protected from unauthorized access.

The non-functional requirements of the FYPMS ensure that the system is not only functional but also efficient, secure, and user-friendly. These requirements support a smooth user experience while maintaining the reliability and integrity of the system.

They form the foundation that allows the system to operate effectively in a modern academic environment, improving the overall management of final year projects.

4.11. Hardware/Software Requirements

4.11.2. Hardware requirements

The hardware requirements for the Final Year Project Management System (FYPMS) ensure that the system runs smoothly and reliably for both students and supervisors.

First, a **computer or laptop** is required. This is the main device used to access the system. It should have at least **8GB of RAM** to allow smooth performance when running browsers, backend services, and development tools. With sufficient memory, users can interact with dashboards, upload files, and review submissions without lag.

Secondly, **storage space** is important. A minimum of **100GB hard disk space** is recommended. This storage is used for saving project files such as proposals, reports, snapshots, and system-related data. As more students use the system, the storage ensures there is enough room for growth.

Another important component is a **stable internet connection**. Since the system communicates with a backend server and database, internet access is required for uploading documents, fetching student data, and submitting feedback.

Additionally, a **power backup system (UPS)** is recommended. This helps prevent data loss and system interruption during power outages, especially when uploading files or saving important information.

In summary, these hardware components provide a stable environment that supports the performance, reliability, and scalability of the FYPMS

4.11.3. Software requirements

The software requirements define the technologies and tools used to build and run the Final Year Project Management System.

At the base level, the system requires an **operating system** such as Windows 10 or higher. This provides a stable platform for running development tools, browsers, and backend services.

For the backend, the system is built using **Java with Spring Boot**. This framework handles core functionalities such as authentication, API development, business logic, and communication with the database. It ensures that the system is secure, scalable, and efficient.

The database used is **PostgreSQL**, which stores all system data including users, submissions, comments, grades, and project details. PostgreSQL provides strong data integrity and supports complex relationships between entities.

On the frontend, the system uses **HTML, CSS, and JavaScript**. These technologies create an interactive user interface where students and supervisors can log in, upload files, review submissions, and manage projects.

A **web browser** such as Google Chrome or Microsoft Edge is required to access the system. The browser acts as the interface between the user and the application.

For development and testing, tools like **IntelliJ IDEA or VS Code** were used for coding, while **Swagger UI** can be used to test API endpoints.

Conclusion

The hardware and software requirements of the FYPMS work together to create a reliable and efficient system. The hardware provides the physical support needed to run the system, while the software defines how the system operates and delivers its services. Together, they ensure that students and supervisors can interact with the system smoothly, securely, and effectively.

4.12. System Design

This is the process of defining the architecture, modules and interfaces of the system to realise the functionality of the system. The purpose is to translate user and system requirements into standard design artifacts such as system architecture, context diagram, DFD and system modelling using case diagrams.

4.12.2. System Architecture

When designing the FYPMS, the architectural framework chosen played a crucial role in shaping the structure and functionality of the web-based system. In this context, the system opted for the Model-View-Controller (MVC) architecture; a sophisticated and strategic choice that only enhances the system's efficiency but also elevates the overall user experience.

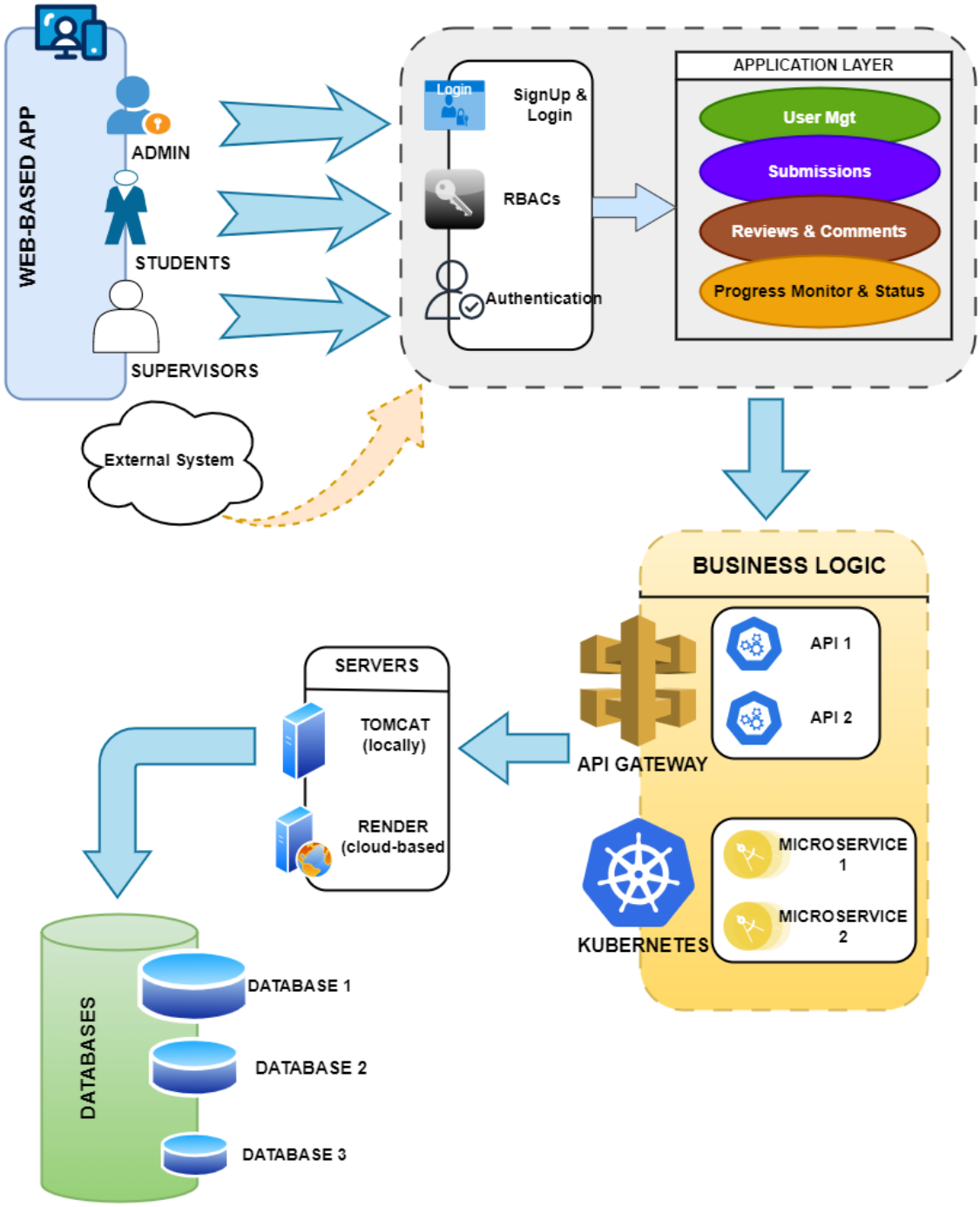


Figure 2: The FYPMS System Architecture.

4.12.3. Context Diagram

The context diagram provides a high-level overview of the FYPMS. It shows the system as a single process with external entities interacting via data flows. The student could log into the system and edit profile, submit project requirements, check submission status and supervisor comments. The administrator could log into the system and edit student information, assign

supervisors and system overview. The supervisor could log into the system and edit profile, view student project submissions, approve and reject submissions and make comments.

This model simplifies the understanding of the system boundaries and external communication channels.

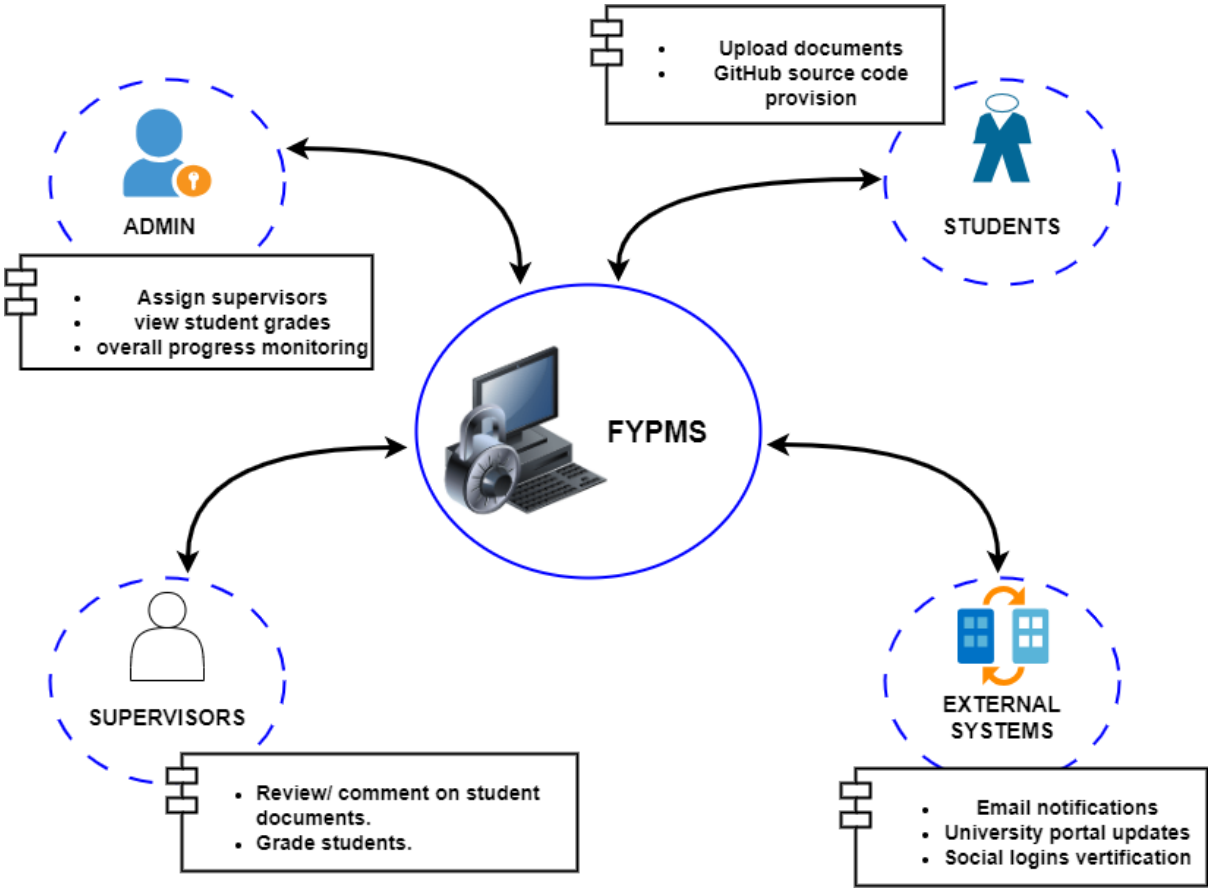
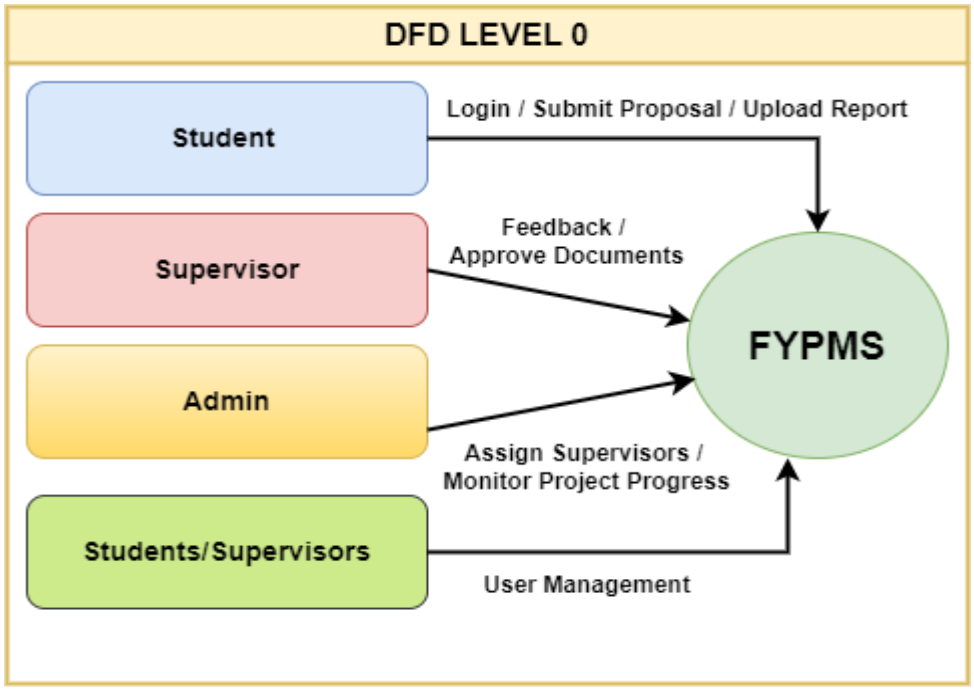


Figure 3: The FYPMS Context Diagram.

4.12.4. Data Flow Diagrams (DFD)

The DFD illustrates how data moves within the Final Year Project Management System (FYPMS). It shows the interaction between users (students, supervisors and administrators) and the system processes



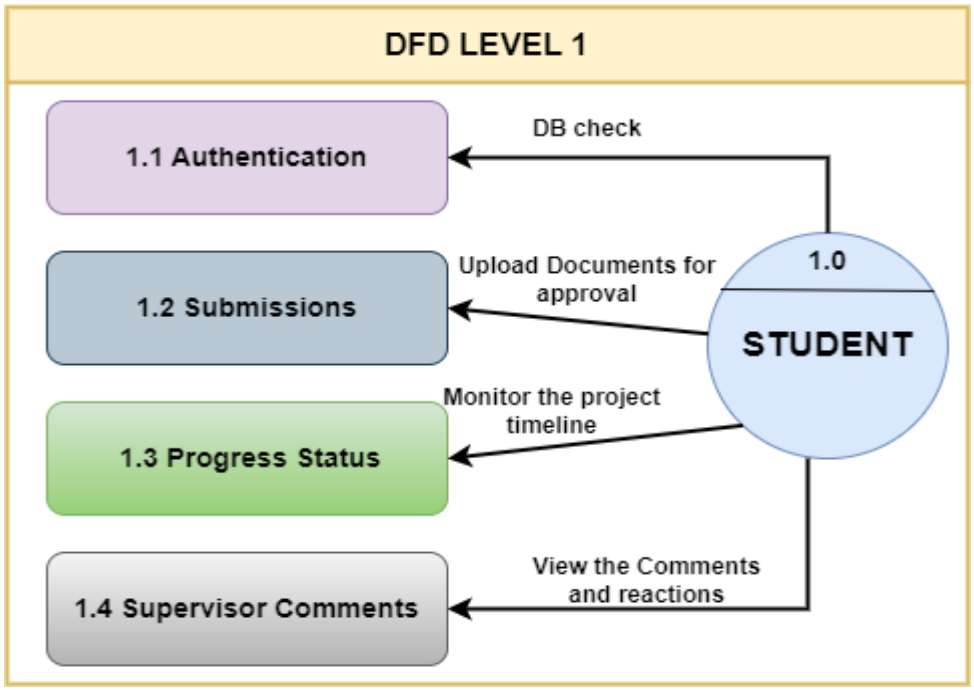


Figure 4: The FYPMS Level 0 & Level 1 DFDs.

Level 0 DFD

The Level 0 Data Flow Diagram provides a high-level overview of the FYPMS as a single process. It illustrates how external entities interact with the system without delving into internal workings. Students submit project titles, proposals, and final reports, while supervisors provide

feedback, approve or reject submissions, and track student progress. Administrators oversee user management, assign supervisors, and ensure smooth operation. The FYPMS acts as the central hub, processing all requests and storing the related data.

Level 1 DFD

The Level 1 DFD decomposes the system into its core internal processes, offering a more detailed view of its functionality. Key processes include Authentication, Submissions, Progress Tracking, and Supervisor Comments. Each process interacts with users and the database to maintain the system's operations effectively.

Process 1.1: Authentication

This process handles user login and verification. Students and supervisors provide their email and password, which the system checks against stored credentials. Valid credentials grant access, while invalid attempts trigger error messages, ensuring secure system entry.

Process 1.2: Submissions

Submissions manage all student uploads, including project titles, proposals, final reports, and snapshots. Each submission is stored and linked to the respective student, allowing supervisors to review and approve the documents efficiently.

Process 1.3: Progress Tracking

Progress Tracking monitors the different stages of a student's project, such as proposal submission, GitHub activity, snapshot uploads, and final report completion. The system calculates progress and presents it through dashboards and charts for both students and supervisors to track development.

Process 1.4: Supervisor Comments

This process manages feedback from supervisors. Supervisors can comment on submissions, approve or reject documents, and provide suggestions for improvement, helping students refine their work while keeping records of evaluations.

These comments are stored in the system and made visible to students, allowing continuous interaction and improvement.

The DFD for the FYPMS shows how the system supports interaction between students, supervisors, and administrators. It highlights how data flows from submission to review,

feedback, and progress tracking. This structure ensures efficiency, transparency, and better management of final year projects.

4.12.5. Use-Case Diagrams

A use-case diagram is a visual representation of the interactions between user, supervisors , administrator and the system, showing how users interact with the system to achieve specific system objectives. A use case typically includes actors, use cases and relationships between actors and the use cases

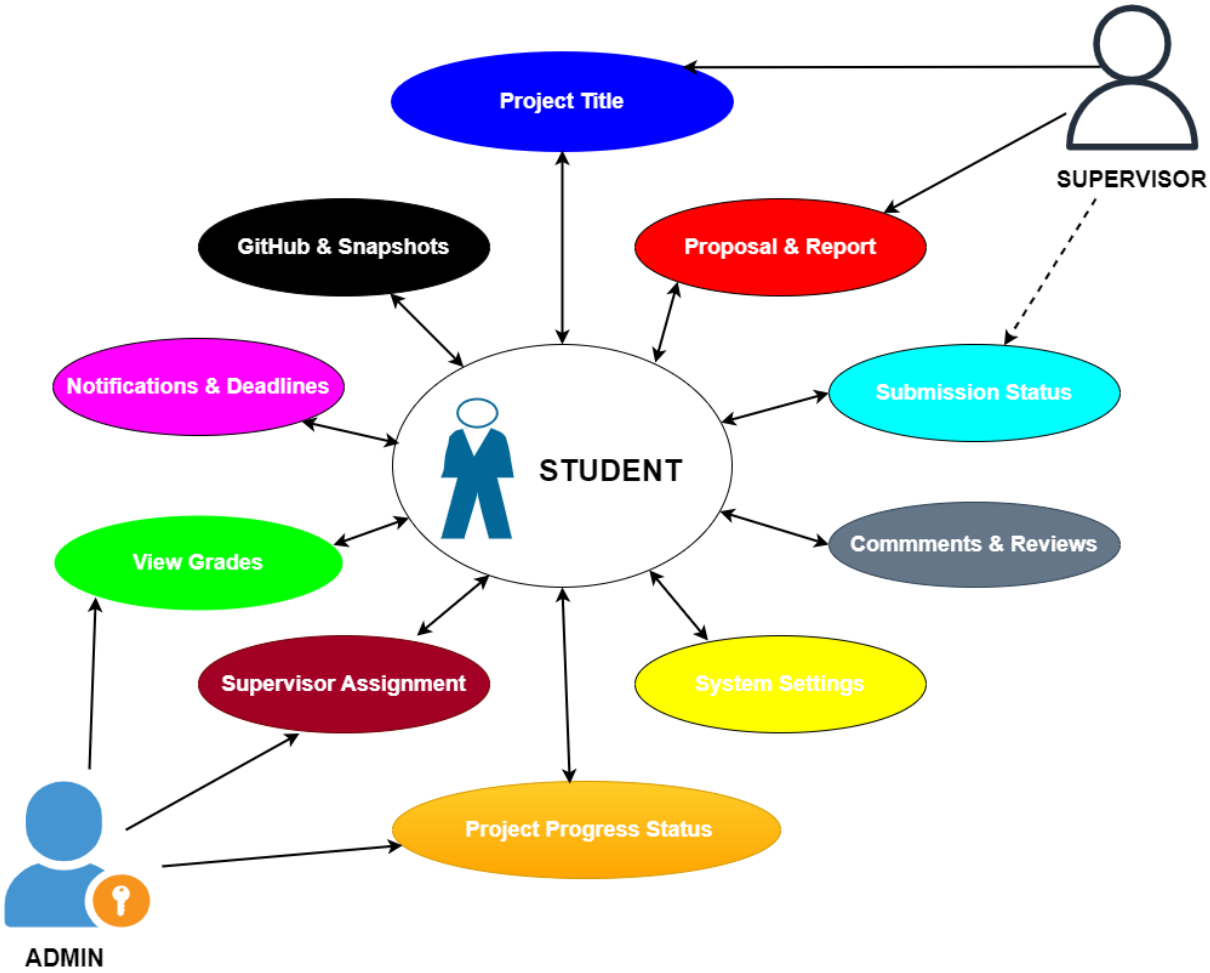


Figure 5: The FYPMS Use-Case Diagram.

4.12.6. Entity Relationship Diagram

The ERD (Entity Relationship Diagram) technique was employed to represent the logical structure of the database. The ERD was designed to capture entities, their key attributes, and the relationships between them to support the management of project submissions and tracking

files. The entities about the data include Students, Admin, Supervisors, Submissions, Comments and Grades.

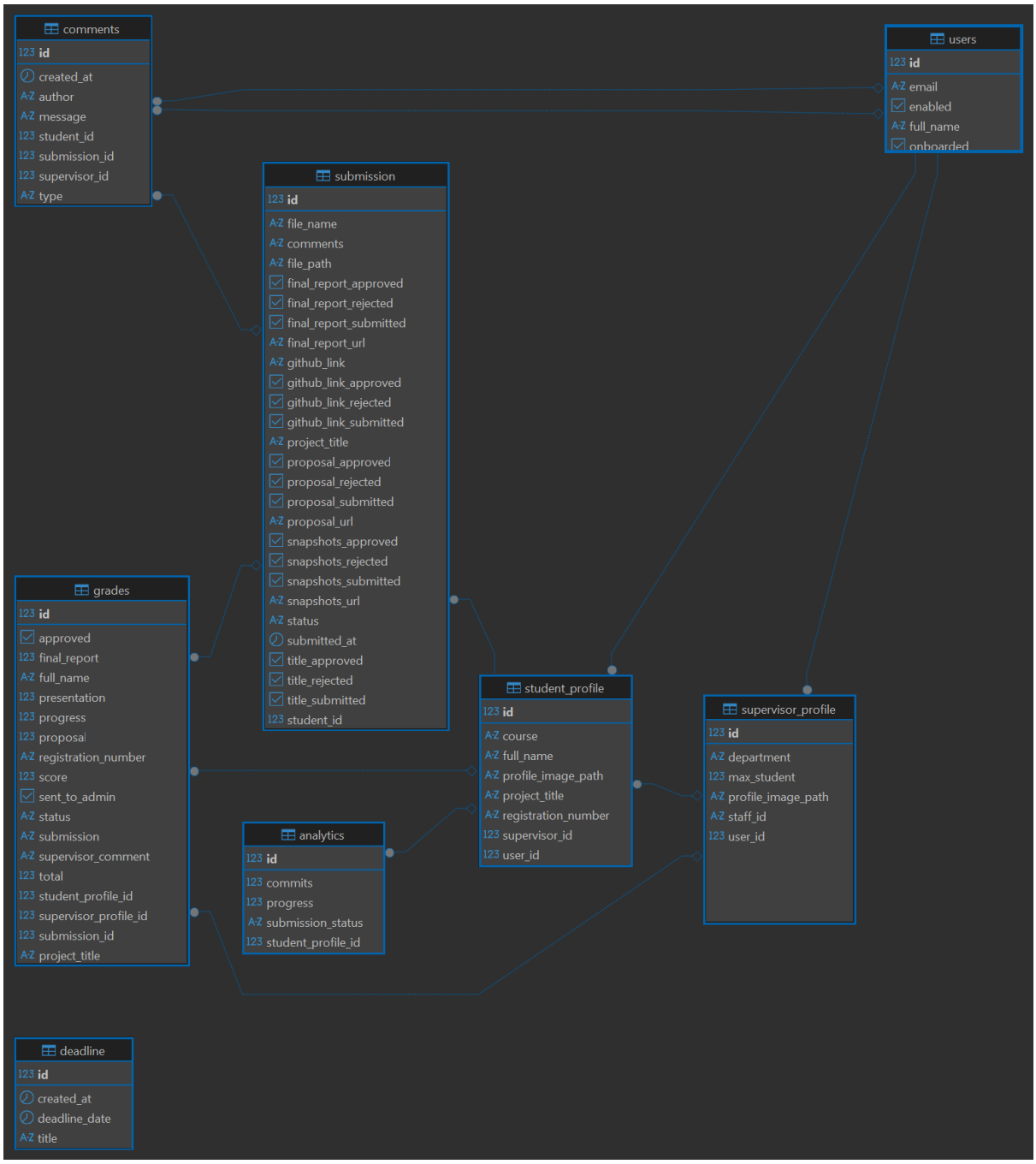


Figure 6: The FYPMS Entity Relationship Diagram.

5. Chapter Five: System Implementation and Testing

5.6. Introduction

This chapter presents the implementation and testing of the developed web-based Final Year Project Management System for Busitema University, Nagongera Campus, Department of Computer Studies. The implementation process was guided by the system design specifications and requirements gathered during the analysis phase. It involved converting design artifacts into a fully functional software product using modern web development technologies including HTML, CSS, JavaScript, Spring Boot and PostgreSQL.

Furthermore, testing was conducted to ensure the system met the expected functional and non-functional requirements. Various testing techniques were applied to identify and resolve defects, verify system performance, and validate user satisfaction. The testing phase also included unit testing, integration testing, system testing and User Acceptance Testing (UAT), aiming to guarantee the reliability, security and usability of the system before deployment.

5.7. System Implementation

This section details the practical development of the FYPMS. The system was implemented as a web-based application using a combination of front-end and back-end technologies. The development followed the design models and requirements identified in previous phases to ensure a functional and user-friendly system.

5.8. Frontend Implementation

The frontend of the Final Year Project Management System (FYPMS) provides an interactive and user-friendly interface that enables users (students, supervisors, and administrator) to interact with the system. It is responsible for rendering dynamic content, handling user interactions, and communicating with backend services through REST APIs. The design focuses on usability, responsiveness, and clarity to support academic workflows.

It communicates with the backend through REST APIs and presents data dynamically. Technologies used include HTML5, CSS3, Vanilla JavaScript (ES6), Chart.js(for analytics visualisation), and Fetch API (for backend communication).

5.8.2. Authentication System (Login, Signup & Onboarding)

5.8.2.1. Login

Users authenticate through a login interface where credentials are validated via the backend. Upon successful authentication, a JWT is returned and stored in the browser. Unauthorised users are automatically redirected to the login page to restrict access.

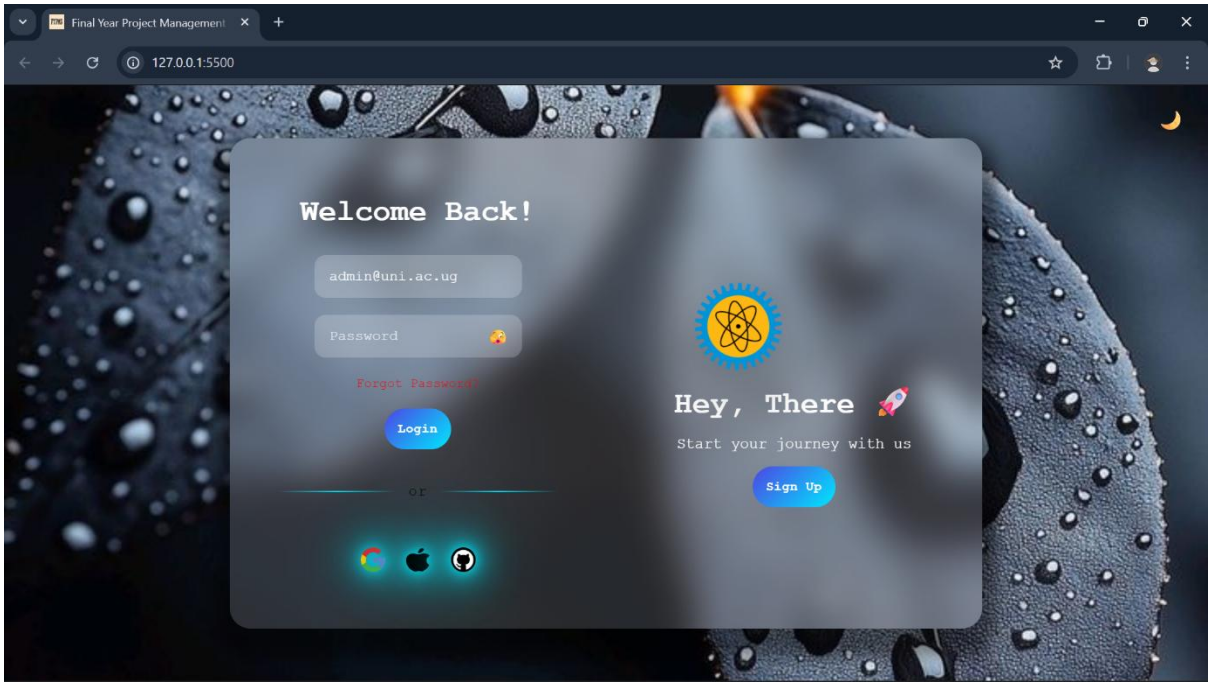


Figure 7: The Login Interface for All Users.

5.8.2.2. Signup

New users register through a signup interface where they provide required details such as full name, email and select a role. The frontend validates inputs before sending them to the backend for account creation. This ensures data consistency and prevents invalid entries.

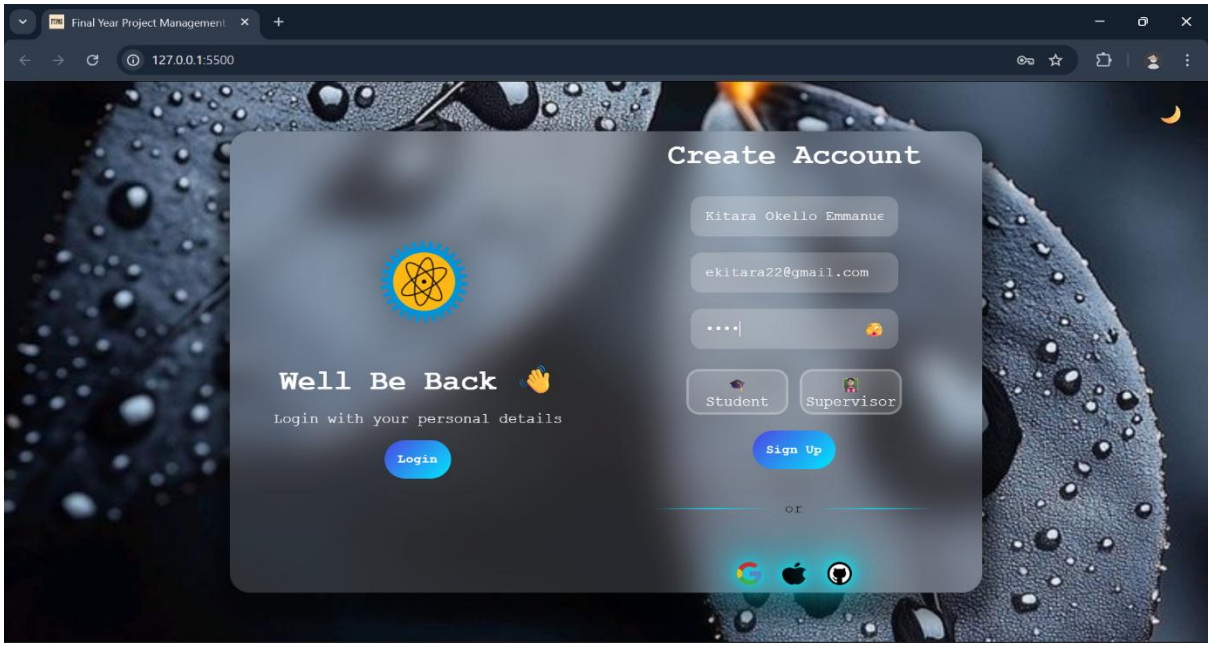


Figure 8: The Signup Interface for Users.

5.8.2.3. Onboarding (Role-based Redirection)

Upon signing up, users are redirected to respective dashboard to complete their profile details and credentials such as staff Id for supervisors and registration number for students.

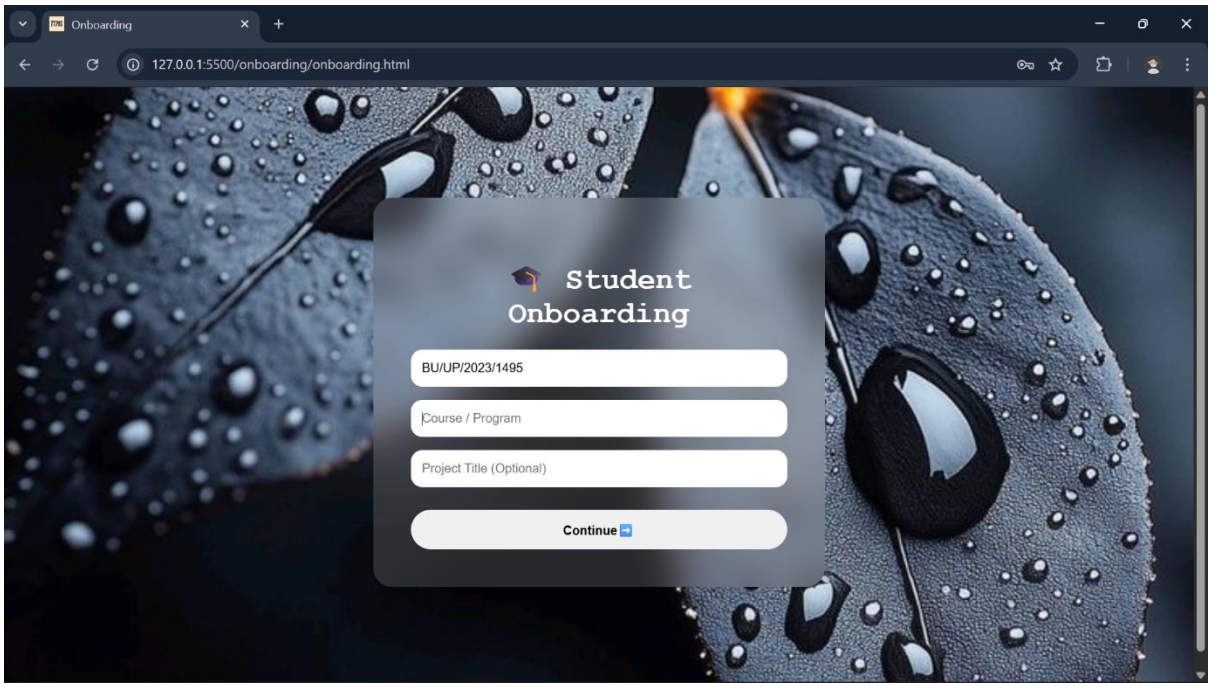


Figure 9: RBAC Dashboard for Students.

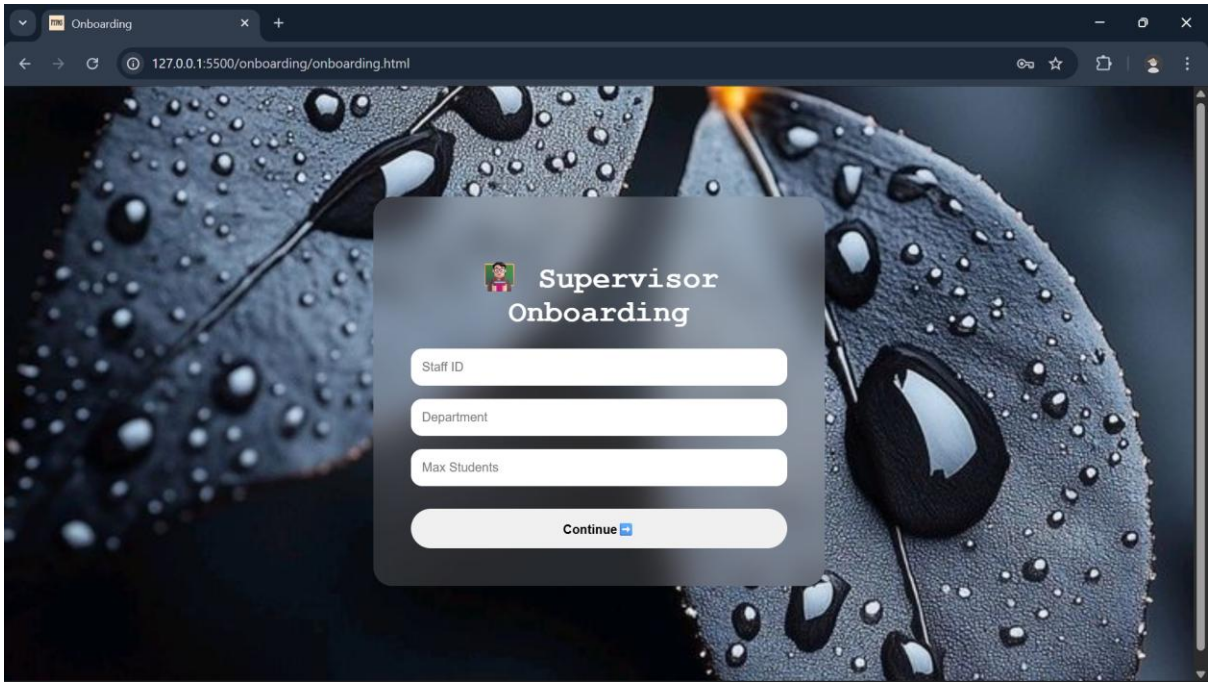


Figure 10: RBAC Dashboard for Supervisors.

5.8.3. Admin Dashboard Implementation

5.8.3.1. Overview

The admin dashboard provides full system control, allowing administrators to monitor users, different project submissions , and monitor overall system activity.

It acts as the central control panel of the system.

Figure 11: Admin Overview Dashboard.

5.8.3.2. User Management

Administrators can manage and update users (students and supervisors). The interface dynamically loads user data and displays it in structured tables and cards.

This ensures efficient management of all system participants.

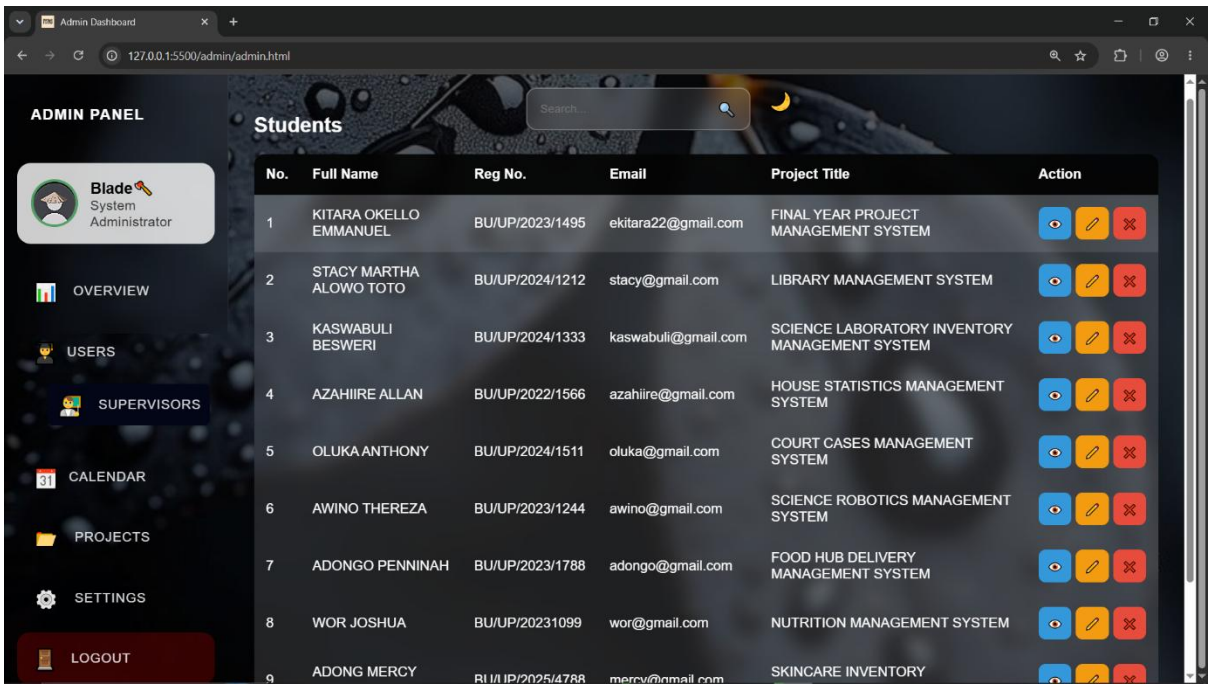


Figure 12: Admin Student Management Interface.

5.8.3.3. Supervisor Assignment

Admins assign supervisors to students through a controlled interface. The system ensures proper linking between entities using backend relationships

This maintains structured supervision workflow.

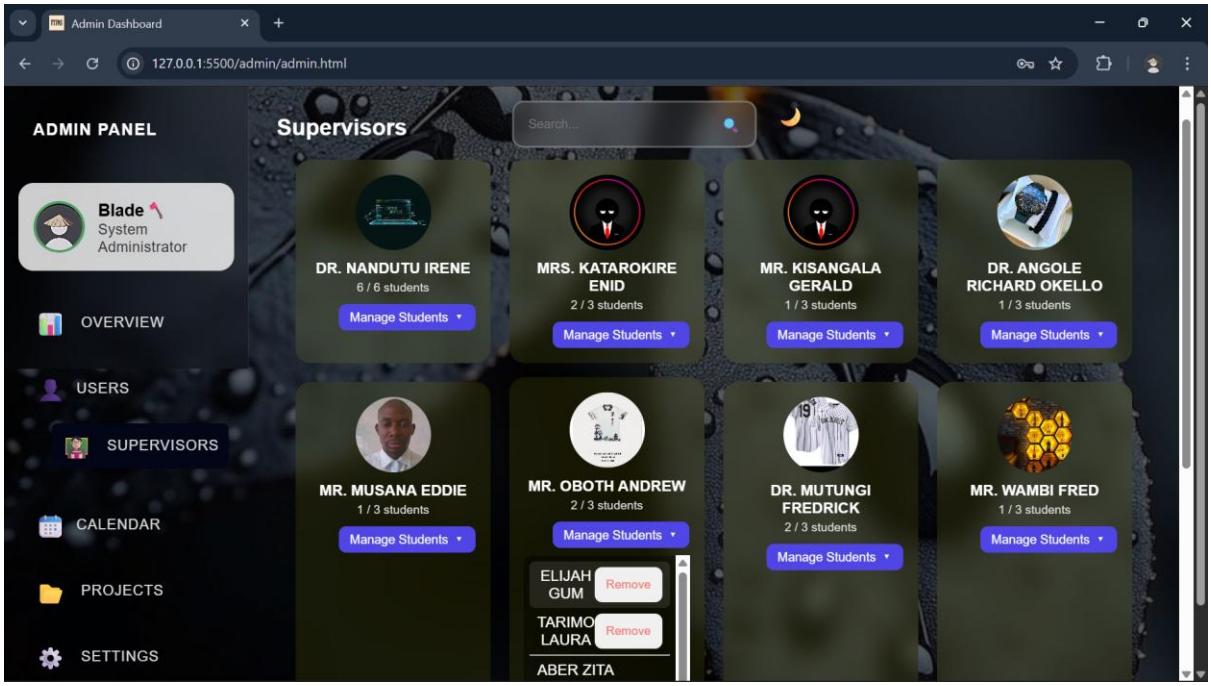


Figure 13: Admin Supervisor Assignment Interface

5.8.3.4. Deadline Selection

Admins can set and manage specific deadlines for tasks or submissions through an intuitive interface. The system ensures proper tracking and enforces timelines, maintaining an organised and efficient workflow.

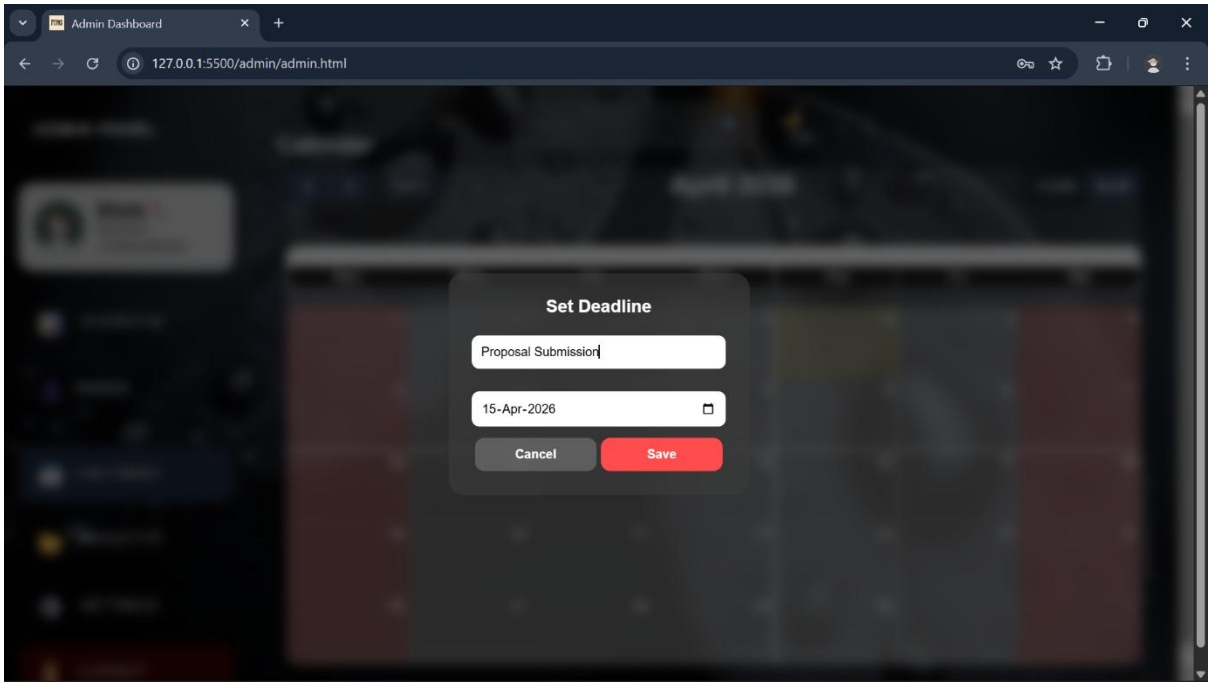


Figure 14: Admin Deadline Selection.

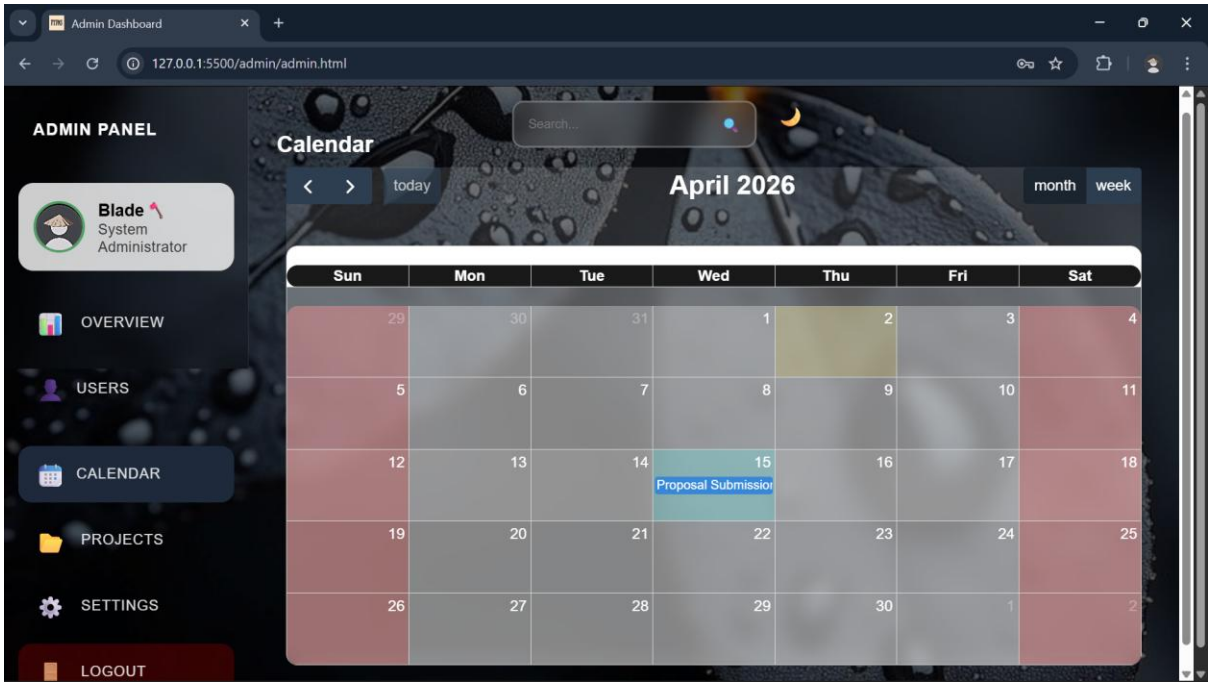


Figure 15: Admin Deadline Calendar.

5.8.4. Student Dashboard Implementation

5.8.4.1. Profile Management

Students can view and update their profile information and assigned supervisor, including uploading a profile image. The system provides instant preview functionality before upload.

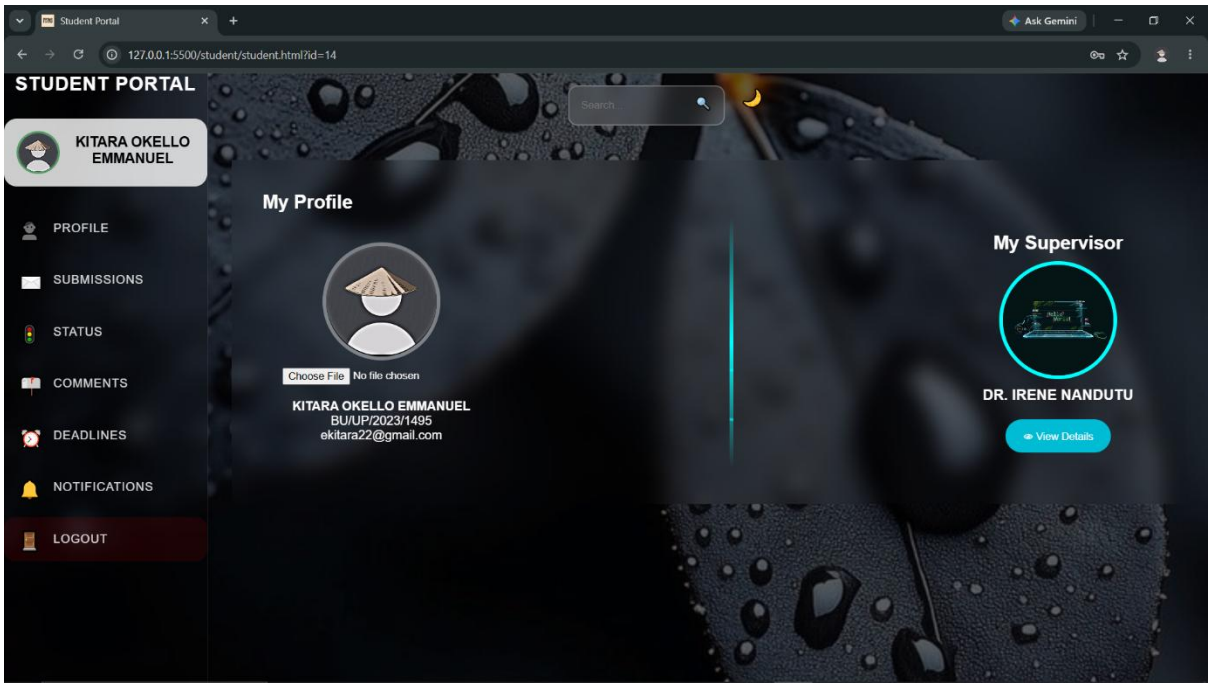


Figure 16: Student Profile Interface.

5.8.4.2. Project Submission Interface

Students can submit project components such as proposals, final reports, and GitHub links. Files are uploaded and previewed dynamically.

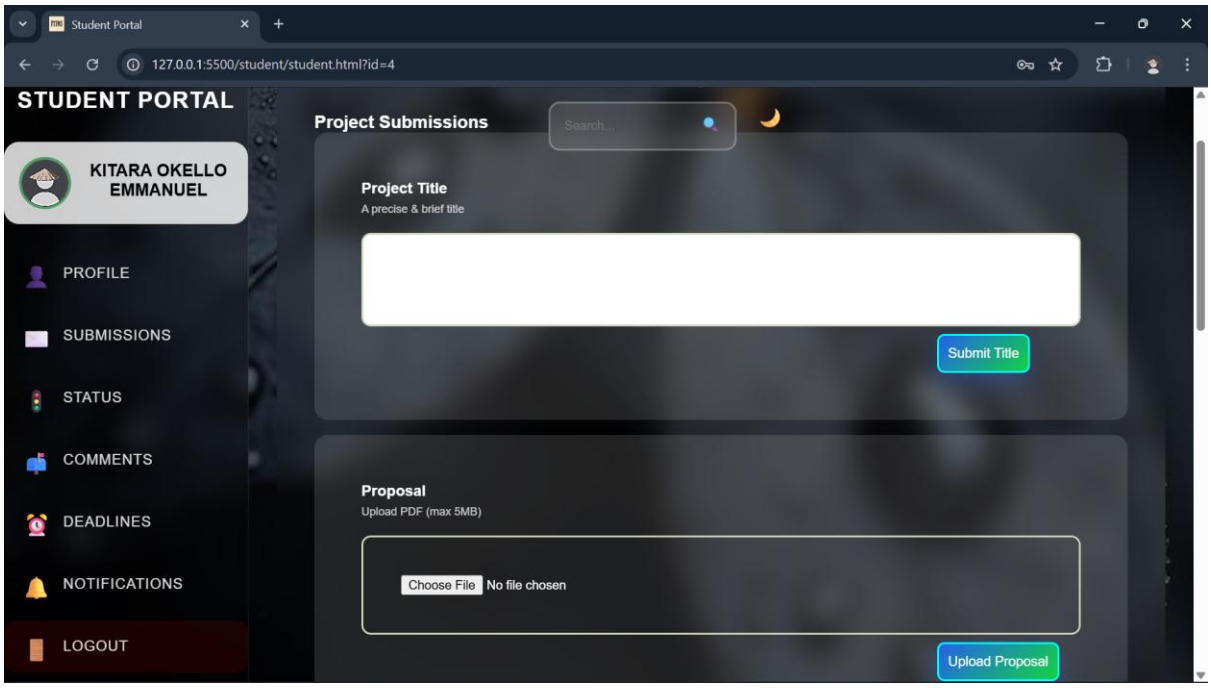


Figure 17: Student Project Submission Interface.

5.8.4.3. Progress Tracking Interface

Students can monitor their progress through visual indicators such as progress bars and battery status updates. This helps them stay aligned with deadlines and progress.

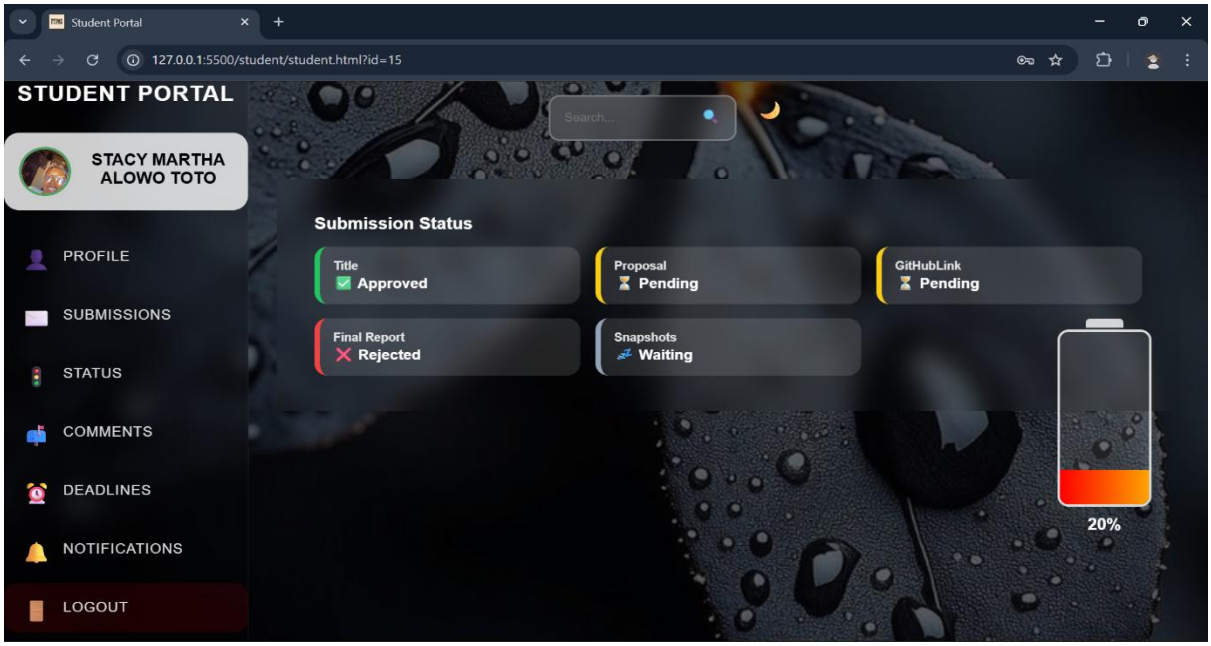


Figure 18: Different Project Submission Status of a Student.

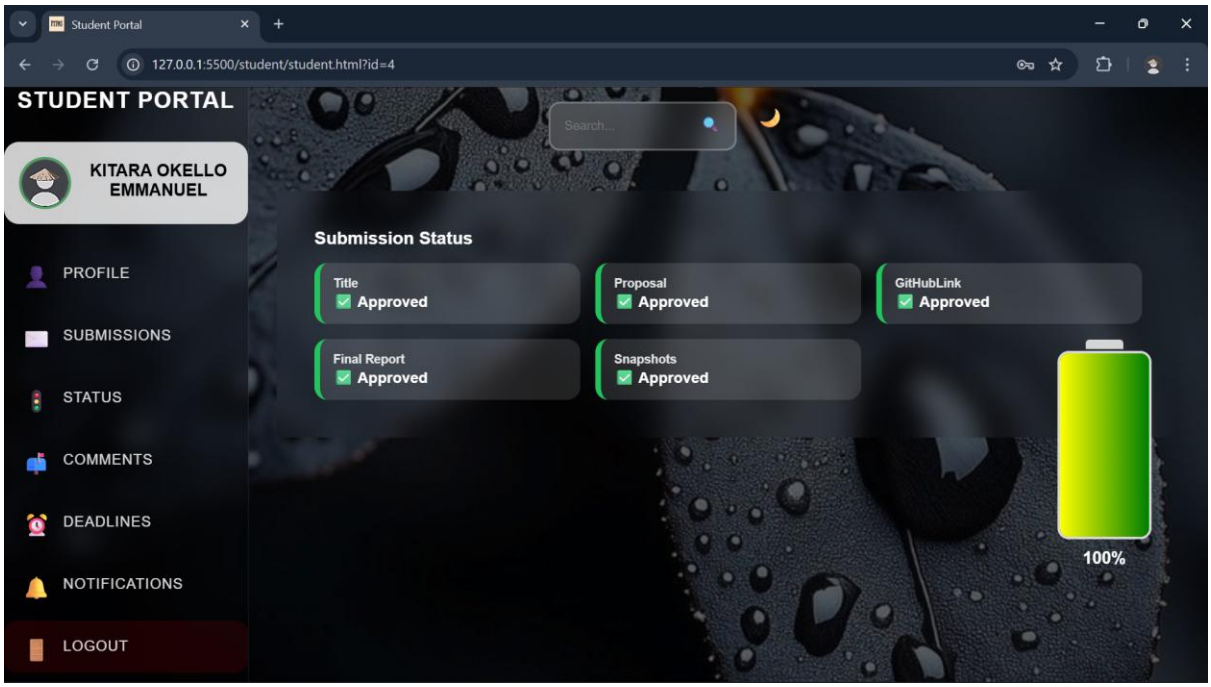


Figure 19: A Complete Project Submission Status of a student.

5.8.4.4. Deadline Management

Deadlines are dynamically displayed and formatted for readability.

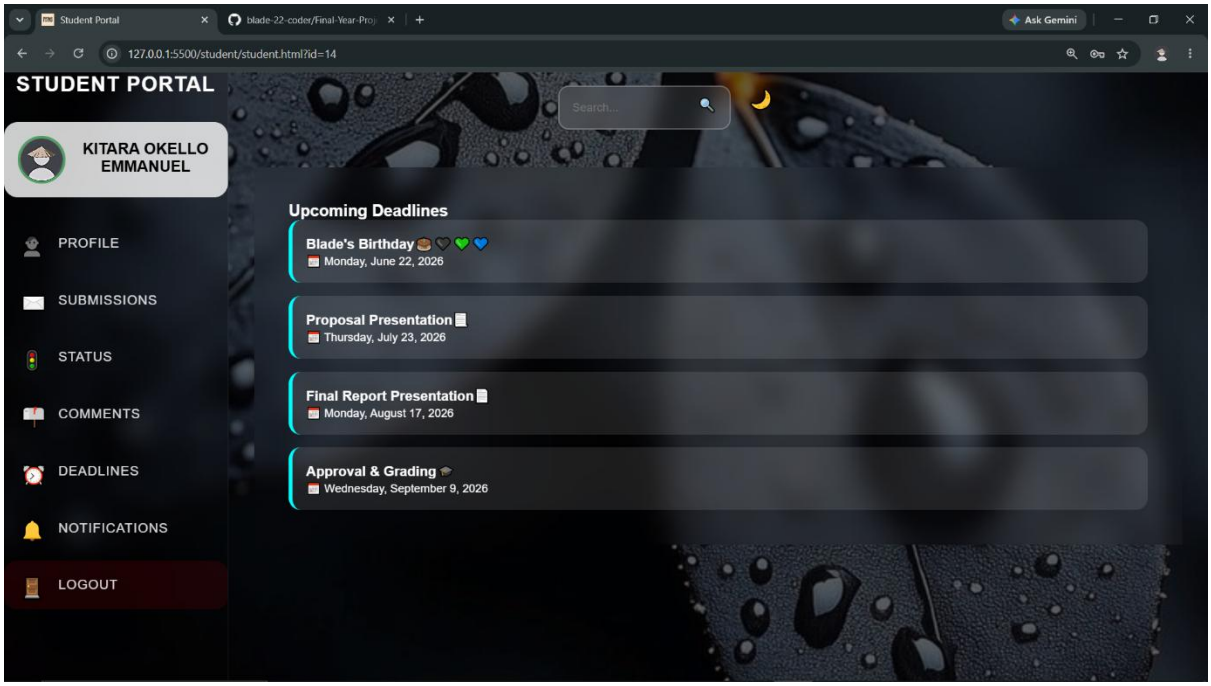


Figure 20: Student Deadline Interface.

5.8.5. Supervisor Dashboard Implementation

5.8.5.1. Profile Management

Supervisors can view and update their profile information, including uploading a profile image. The system provides instant preview functionality before upload.

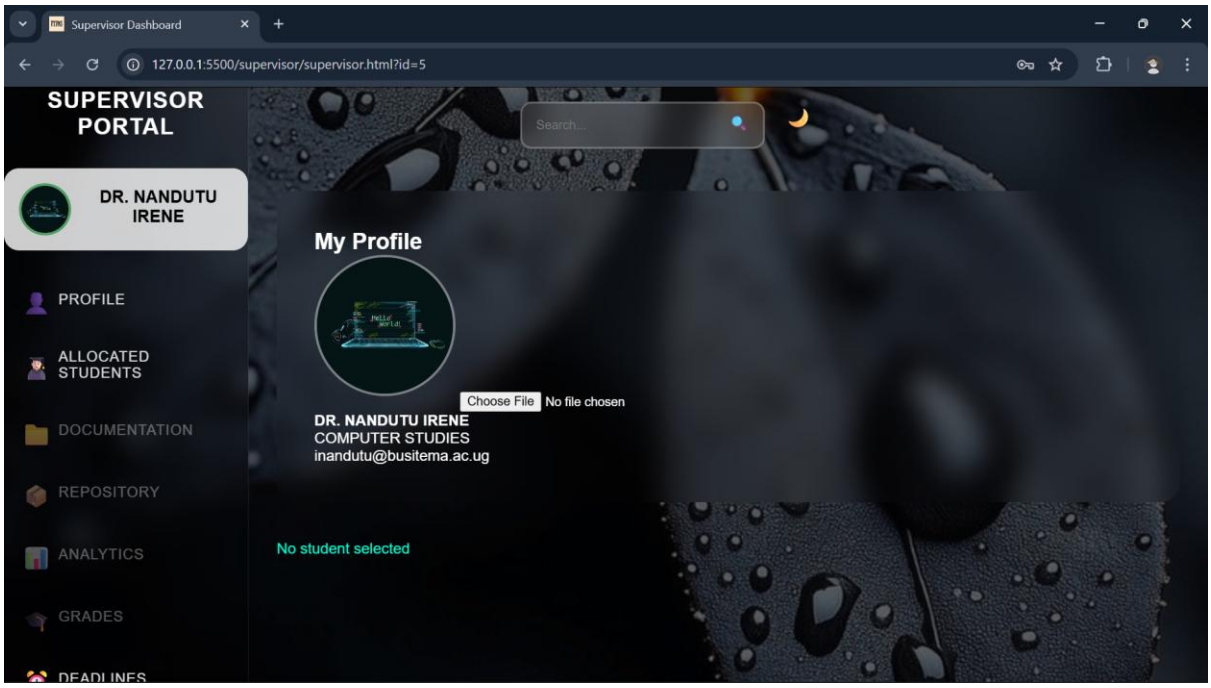


Figure 21: Supervisor Profile Interface.

5.8.5.2. Student Management Interface

Supervisors can view a list of assigned students displayed in a card-based layout. Each card contains essential information such as student name, registration number and project progress. The interface allows supervisors to select a student, which loads detailed project data dynamically

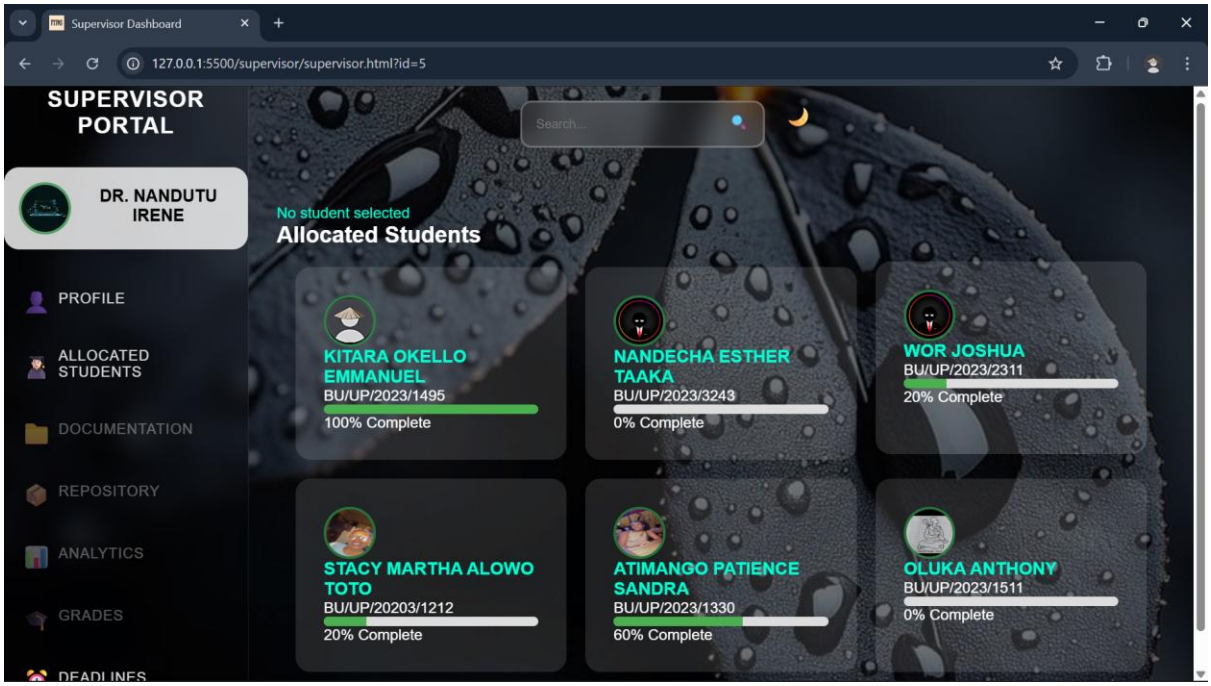


Figure 22: Supervisor's Assigned Students Interface.

5.8.5.3. Navigation and Access Control

The supervisor dashboard uses a sidebar navigation system for switching between sections. Certain sections remain locked until a student is selected, ensuring a proper workflow order.

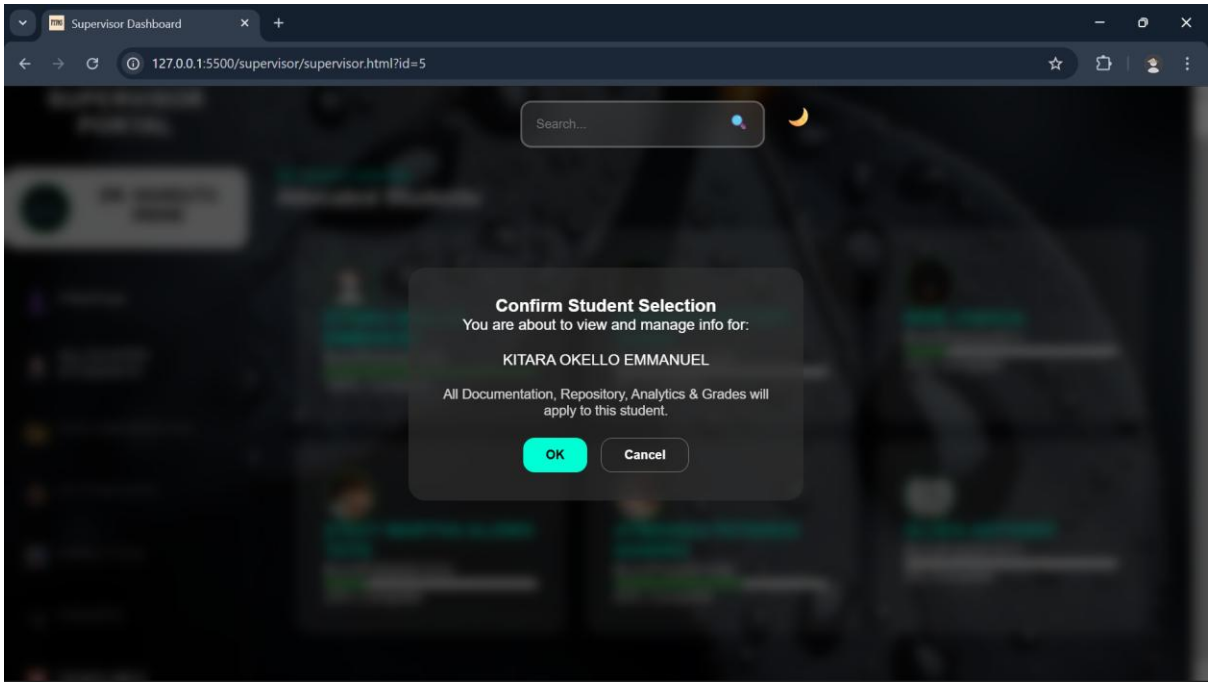


Figure 23: Assigned Student Selection.

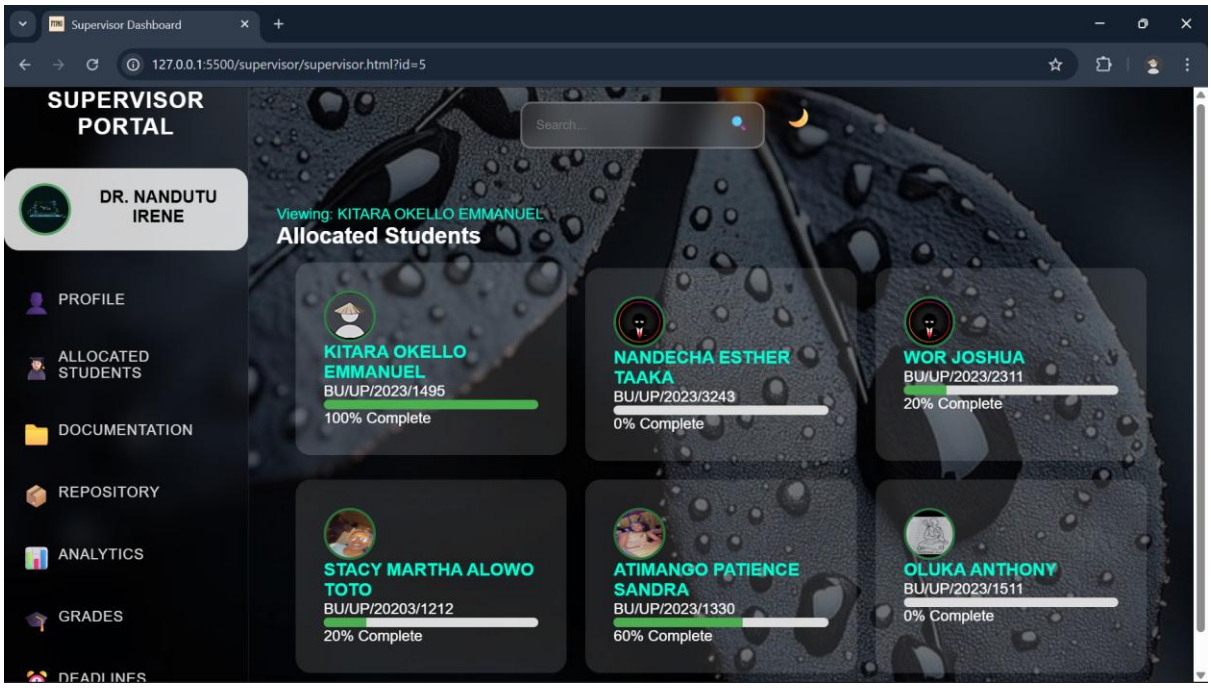


Figure 24: Unlocked Sections per Assigned Student.

5.8.5.4. Document and File Management

The system allows supervisors to view, preview and download documents such as proposals, reports and snapshots. Files are handled dynamically to ensure smooth interaction as they can be approved or rejected based on the project progress and quality.

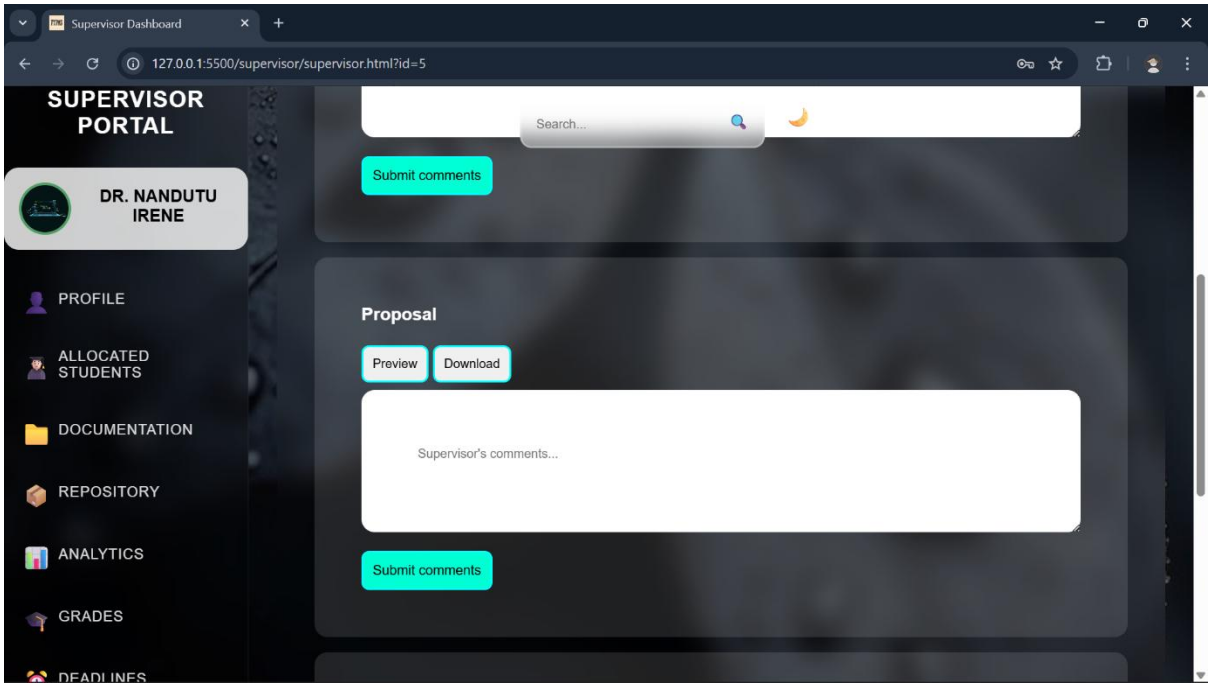


Figure 25: Submitted Project Documentation Interface.

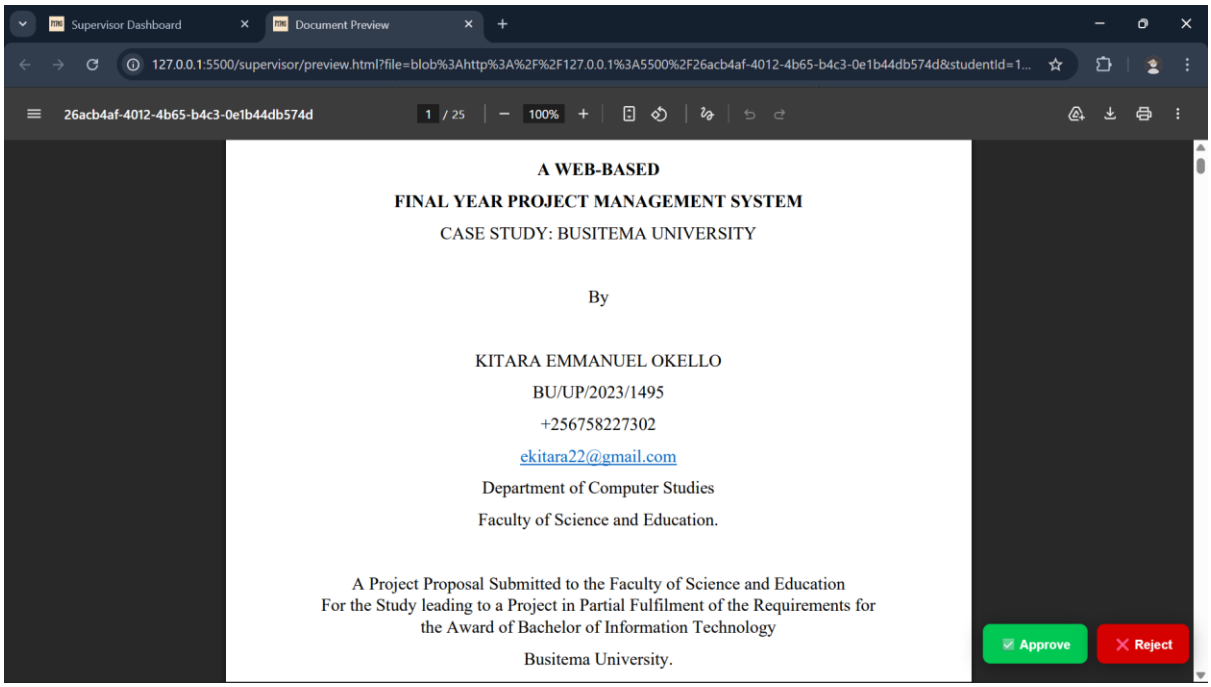


Figure 26: Project Document Preview.

5.8.5.5. GitHub Repository Review

Supervisors can review student-submitted GitHub repositories directly by clicking the links. They can approve or reject submissions based on project progress and quality.

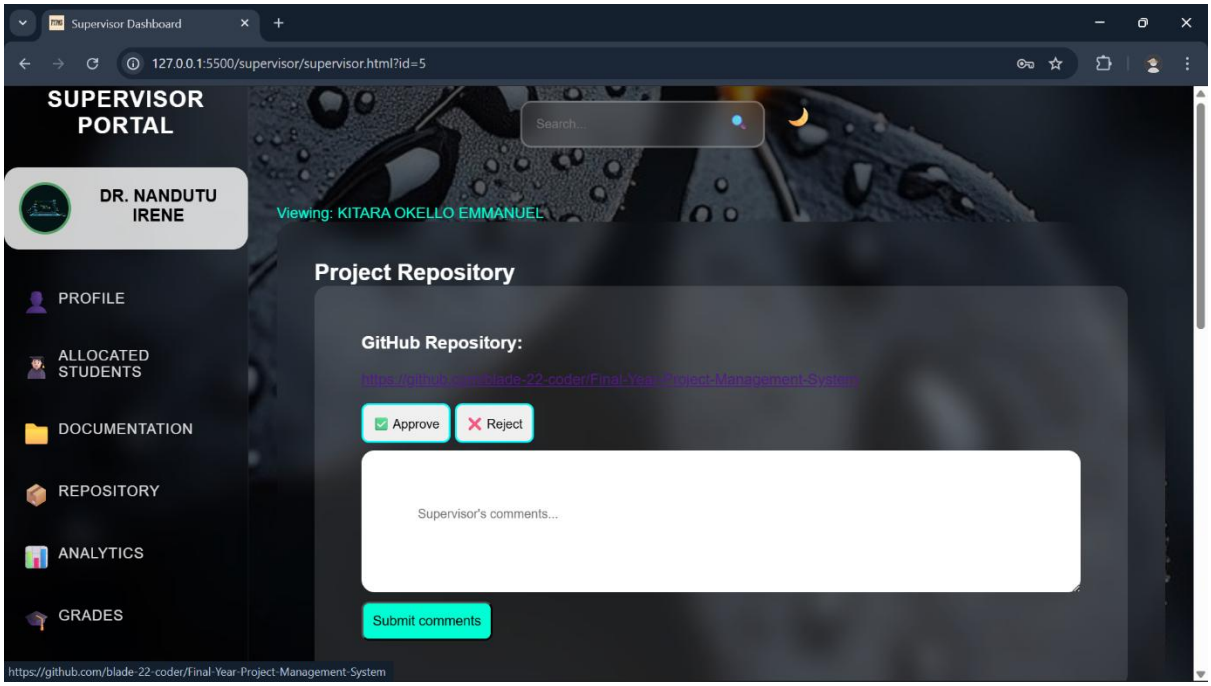


Figure 27: Submitted GitHub Link Interface.

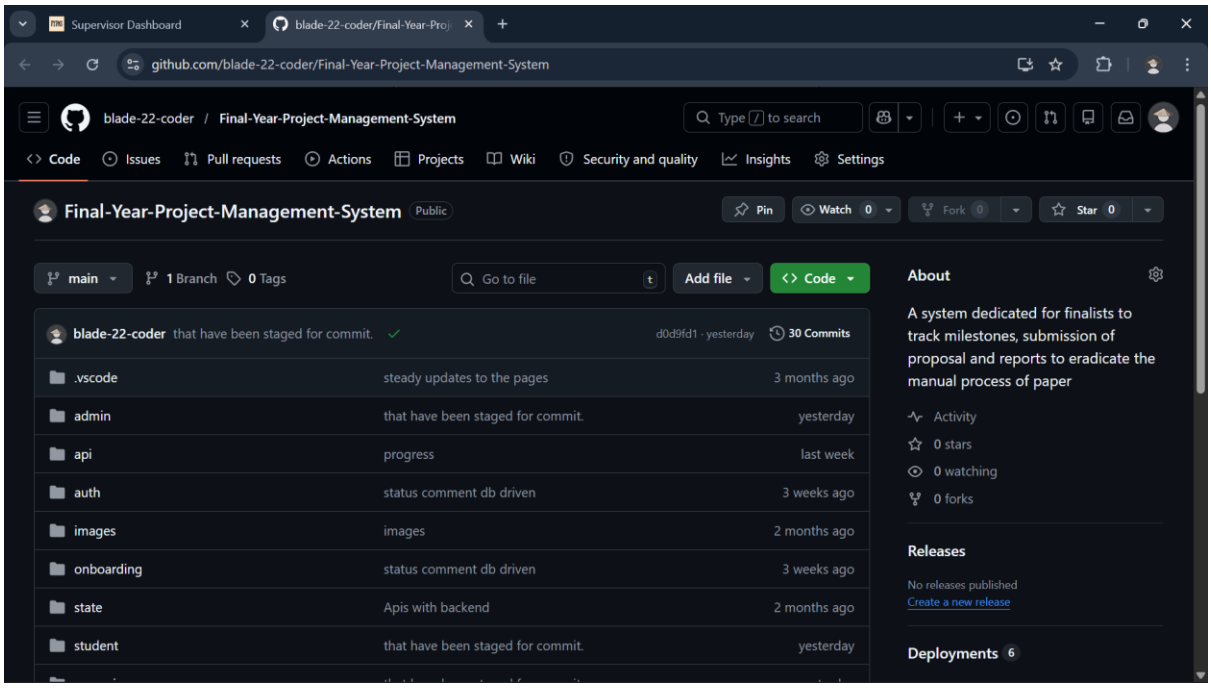


Figure 28: GitHub Dashboard per Submitted Link.

5.8.5.6. Snapshot Management

Students upload project snapshots (screenshots), which supervisors can view in a grid layout. Clicking on an image opens a preview for a detailed inspection.

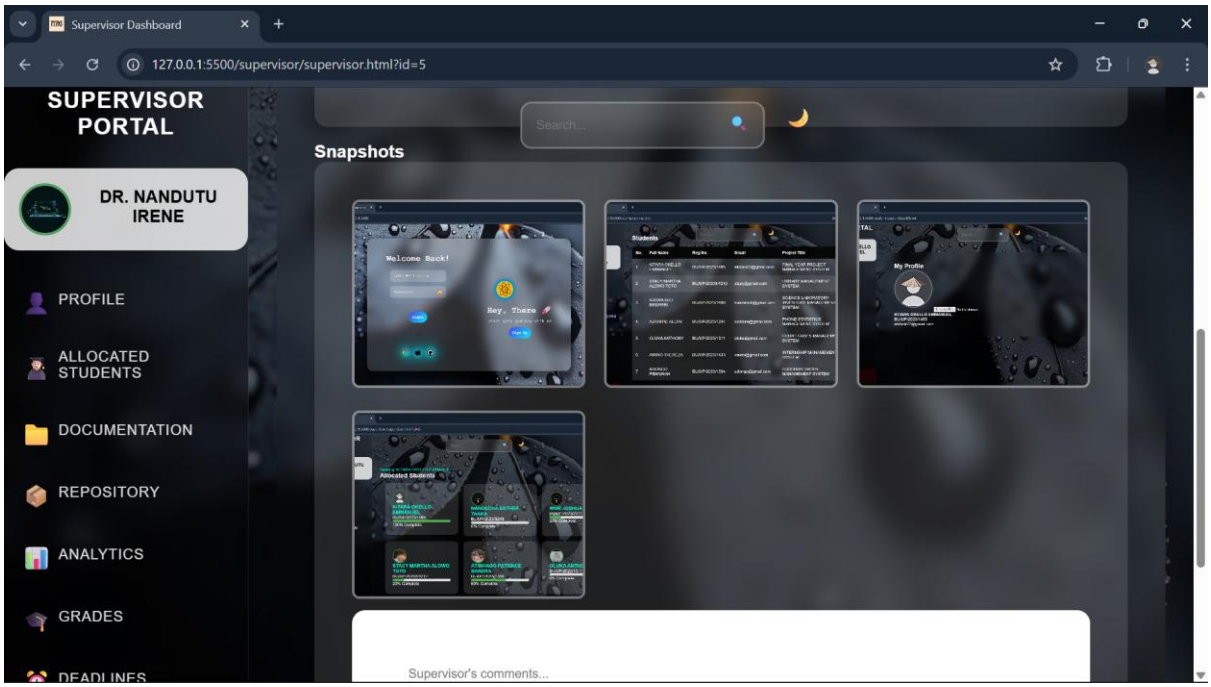


Figure 29: Snapshots Preview Interface.

5.8.5.7. Analytics Dashboard

Supervisors have access to analytics that visualise student performance and activity trends. Charts provide insights into progress and repository activity.

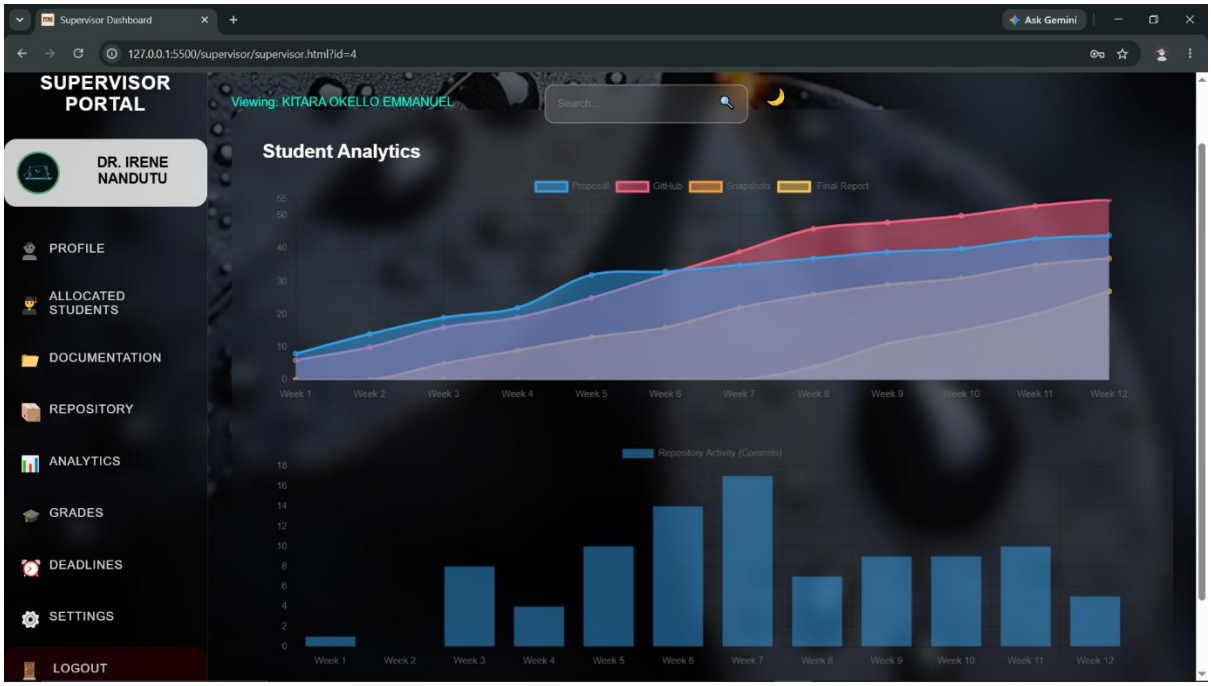


Figure 30: Analytics Dashboard Interface

5.8.5.8. Grading System Interface

The grading interface allows supervisors to assign marks across multiple evaluation criteria such as proposal, progress, report and presentation

The system automatically calculates total scores and supports both draft saving and final submission

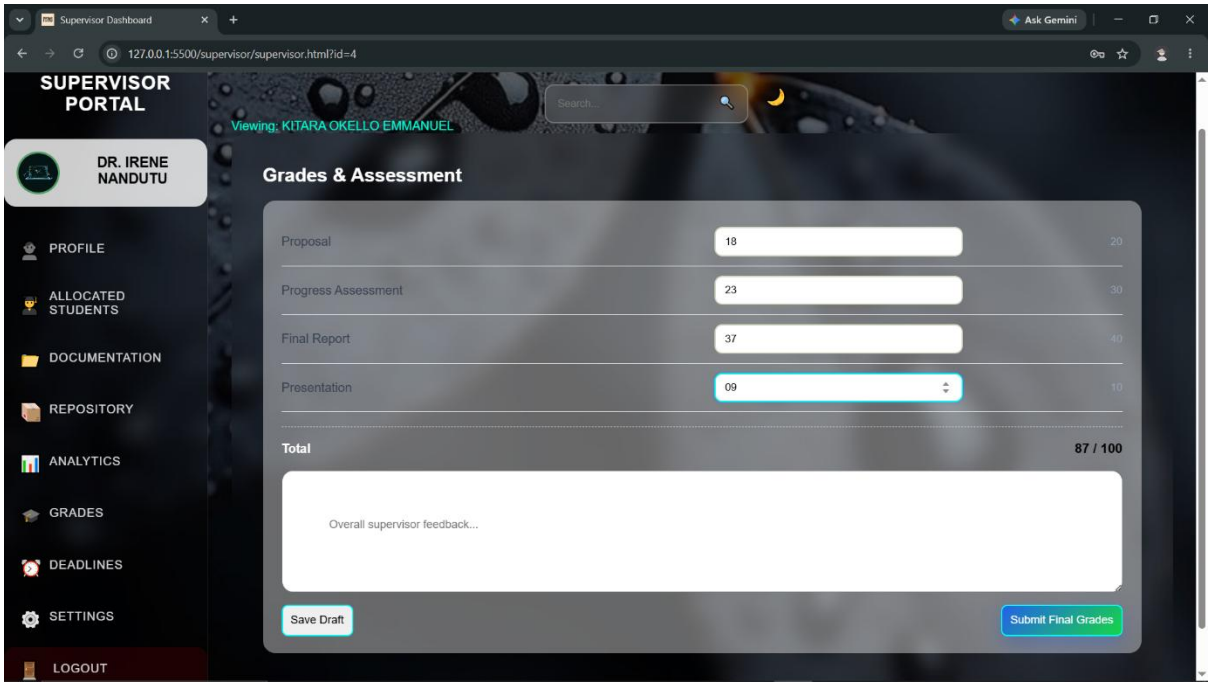


Figure 31: Submission Grading Interface.

5.9. Backend Implementation

The backend of the Final Year Project Management System (FYPMS) is responsible for handling business logic, data processing, authentication, and communication with the database. It exposes RESTful APIs that allow supervisors, students, and administrators to interact with the system efficiently.

The backend was implemented using: Spring Boot(Java), Spring Data JPA(Hibernate), PostgreSQL Database, JWT-based Authentication and REST APIs.
The backend followed a layered architecture:

1. Controller layer: handling HTTP requests and responses. For example, *SupervisorProfileController.*
2. Service layer(Logic layer): Contains business logic (validation, processing) for handling approvals, analytics, grading. For example, *AdminService.*
3. Repository layer: Interfaces for database access using JPA. For example, *SubmissionRepository.*
4. Entity layer: represents tables using JPA annotations. For example, *Comment, Grades*.

The system uses PostgreSQL with relational schema design with key entities such as *User, StudentProfile, SupervisorProfile, Comment, Grades and Submission* e.tc.

The system uses JSON Web Token (JWT) for secure access; when user logs in—server generates JWT—token stored in frontend (*localStorage*)—and Spring Security: Role-based access (admin, supervisor, student) and unauthorised access returns 403 Forbidden.

The backend exposes RESTful endpoints structured such as;

1. Profile
    - *GET /api/supervisor/me*
    - *PUT /api/supervisor/upload-profile-image*
2. Students
    - *GET /api/supervisor/my-students*
3. Project Title
    - *GET /api/supervisor/title/{studentId}*
    - *PUT /api/supervisor/{studentId}/title/approve*
    - *PUT /api/supervisor/{studentId}/title/reject*
4. Documents
    - *GET /api/supervisor/docs/{studentId}*
    - *POST /api/supervisor/docs/{docId}/comment*
5. GitHub
    - *GET /api/supervisor/github/{studentId}*
    - *PUT /api/supervisor/{studentId}/githubLink/approve*
    - *PUT /api/supervisor/{studentId}/githubLink/reject*

The backend also supports file operations:

1. Upload; profile images and submissions (proposal, final report and snapshots).
2. Download and Preview; files fetch using a RESTful API and files returned as JSON.

Furthermore, business logic features such as project approval system, where supervisors approve/reject titles, documentation, GitHub links and snapshots, comment system, where supervisors provide feedback per submission linked to submission and users. Analytics system, where both supervisors and admin track project progress, GitHub commits and submission activity, grading system, where there is multi-component grading, auto total calculation on frontend, draft and final submission supported.

5.10. System Testing

The testing phase aimed to validate the functional and non-functional correctness of the system and ensure it meets the user requirements. The following testing techniques were applied:

- Unit Testing: Individual modules such as login, project submission and approval and rejection status were tested independently to ensure correct output.
- Integration Testing: Interaction between front-end and back-end components was tested, ensuring seamless data flow between forms and the database.
- System Testing: The entire system was tested as a whole to validate whether all components functioned together as expected.
- User Acceptance Testing (UAT): Selected end-users including students and supervisors interacted with the system and provided feedback on usability, accuracy, and performance.

Key outcomes:

- All critical functions worked as intended.
- Minor UI issues were identified and resolved during UAT.
- The system was confirmed to be ready for deployment in a real operational environment.

5.11. System validation

Validation of the system was done in order to confirm whether the system met its specified requirements and functionalities as intended. It involved a series of evaluations to ensure that the system is correct, consistent, complete, traceable and secure. Functional validation was done in order to test the system's core functionalities and ensure they meet the specified requirements.

Here 7 members participated in the validation of the system and out of them, 5 of the members agreed that the system meets all the functionalities intended. This shows that 71.4% of the people agreed that the system is deemed fit for its use.

6. Chapter Six: Discussion, Conclusion, Recommendations, Limitations and Future Work

6.6. Introduction

This chapter presents a discussion of the results obtained from the design and implementation of the web-based Final Year Project Management System. The chapter also draws conclusions from the study, offers recommendations for system improvement, highlights limitations encountered during system development, and suggests areas for future research. The discussion is organised based on the study's specific objectives and is supported by insights from the reviewed literature.

6.7. Discussion of results

The findings from the system development and testing processes are discussed in this section in relation to the specific objectives and the existing document reviewed in Chapter Two.

6.7.2. Objective 1: To review the literature and determine the requirements for the Final Year Project Management System at Busitema University.

The study began by investigating the current final year project management practices at Busitema University. This involved identifying how final year projects are submitted, monitored, and how information flows among students, supervisors and administrative personnel. The investigation uncovered significant challenges: inconsistent record-keeping, delays in tracking submission status and supervisor comments, difficulty aggregating data for reports and limited accountability for submissions and comments. Stakeholders (students, supervisors and administration) highlighted needs for real-time updates, role-based user access, notifications for submissions and status, and secure project handling.

These findings align with (Oyekan & Ikuomola, 2024), who observed that universities often suffer from delayed reporting, lack of accurate inventory data, and require features such as alert/threshold notifications. Similarly, (Febrina & Utami, 2022) noted that manual processes in academic laboratories frequently lead to errors, delays and poor data retrieval, motivating the development of digital systems with features like reporting, tracking and logouts. Non-functional requirements such as usability, scalability and security also emerged, consistent

with (Afolabi, 2023), who emphasized speed, accuracy, and secure handling of final year projects in academic systems.

6.7.3. Objective 2: To design a web-based Final Year Project Management System tailored to the needs of Busitema University.

Using the requirements, the design phase focused on creating modules such as user registration and logins, role-based access, threshold alerts, reporting dashboard, project submissions and real-time tracking. Prototypes were iterated with feedback from supervisors and students. Key considerations included interface clarity, minimal input burden, fast loading times and secure access control.

The design approach is consistent with (Oyekan & Ikuomola, 2024), whose adaptive system incorporated alerts for submissions and status, performance metrics and dashboards for decision making. Similarly, (Febrina & Utami, 2022) emphasised that designing interfaces and workflows based on user feedback enhances system usability and acceptability. These studies confirm that user-centred design ensures Final Year Project Management Systems remain practical, efficient and sustainable.

6.7.4. Objective 3: To implement the web-based Final Year Project Management System using modern technologies and best practices.

The system was implemented using Spring Boot, PostgreSQL, HTML, CSS and JavaScript. Key features included role-based access, JWT authentication, responsive interfaces and real-time data synchronisation (e.g., status approval and rejection of submissions, alerts). Input validation and reporting modules were also included.

These choices are consistent with prior research. For example, (Afolabi, 2023) implemented their academic warehouse system using Spring Boot, HTML and JavaScript, emphasising secure and timely data handling. Likewise (Oyekan & Ikuomola, 2024) recommended Spring Boot and PostgreSQL as robust, cost-effective technologies for university systems

6.7.5. Objective 4: To test and validate the web-based Final Year Project Management System to ensure it meets the specified requirements and user expectations.

Testing phases included unit testing (on individual modules), integration testing (how modules interoperate), and User Acceptance Testing (UAT) with actual users (students, lecturers and administrators). Feedback indicated improved speed, easier supervisor feedback and better

accountability of submissions. Weakness were mainly minor UI issues or performance lags with large datasets

(Febrina & Utami, 2022) found that web-based systems reduced data entry errors and enhanced efficiency in retrieving records. (Oyekan & Ikuomola, 2024) also validate their system with users and found that it successfully met requirements for alerts, reporting and performance.

6.8.Conclusion

The study successfully achieved its objectives by conducting a thorough analysis of existing Final Year Project Management practices, establishing system requirements, designing, implementing and testing a web-based Final Year Project Management System tailored to the needs of Busitema University.

The system addressed the challenges of the manual final year project processes and rudimentary paperwork, including inconsistent record-keeping, delays in asset tracking, and limited accountability. By providing a secure, accessible, and efficient digital platform, the system improves operational efficiency, enhances data accuracy, facilitates real-time monitoring, and strengthens submission management and reporting practices. Overall, the web-based system ensures that Final Year Project resources are properly managed, monitored and utilised effectively within the university.

6.9.Recommendations

Based on the system performance, stakeholder feedback, and observations during testing, the following recommendations are proposed:

- Officially adopt the web-based Final Year Project Management System for managing all Final Year Projects and resources at Busitema University.
- Conduct regular training sessions for students, supervisors and administrative personnel to ensure effective system usage and understanding of its features.
- Integrate the system with the university's main Academic Information Management System and procurement modules to enable seamless final year project tracking and reporting.

- Regularly review and update system functionalities based on user feedback, evolving Final Year Project Management needs, and technological advancements.
- Explore the development of a mobile-accessible version of the system to allow supervisors to update and monitor final year project submissions remotely, improving real-time visibility and management of the resources.

6.10. Limitations and Future Work

While the web-based Final Year Project Management System developed in this study successfully addresses the challenges of manual projects management at Busitema University, several limitations were observed:

- Limited Mobile Accessibility: the current system is primarily web-based and may not be fully optimised for mobile devices, which could limit accessibility for staff working remotely or in different campus locations.
- User Adoption: Despite training sessions, some users may require additional time and support to adapt fully to the new digital workflow, especially those accustomed to manual processes.
- Integration Constraints: Integration with existing university systems, such as procurement and accounting modules, was not fully implemented during the study due to time and resource constraints.
- Scalability for Large Data Sets: While the system performs well with current Final Year Projects volumes, further testing is required to ensure efficiency with larger datasets as the university's student population expand.

6.11. Future Work

- Develop a fully mobile-compatible version of the system to enhance accessibility and real-time management capabilities.
- Implement full integration with the university's Academic Information Management System and procurement platforms to streamline final year project lifecycle management.
- Introduce advanced reporting, analytics, and predictive final year project management features to support strategic planning and resource optimization.

- Conduct periodic user feedback sessions and system audits to continuously refine and improve system performance and usability.

References

Alruwais, N., Wills, G., & Wald, M. (2018). Advantages and challenges of using e-assessment. *International Journal of Information and Education Technology*, 8(1), 34–37.

Ahmad, I., 2024. Evaluation of the implementation of Educational Management Information Systems in supporting strategic decision-making in higher education. *Contemporary Journal of Applied Sciences*, 2(6). Available at: https://doi.org/10.55927/cjas.v2i6.12314.

Haruna, I.U., Awang, H., Mansor, N.S., Ibrahim, H.H., Azni, S. and Khamis, S., 2023. Prototyping a web-based application for remedial education data management. *Sinergi International Journal of Education*, 2(2). Available at: https://doi.org/10.61194/education.v2i2.256.

Kebebe, M. and Regassa, T., 2024. An empirical research on the practices of EMIS in South West Showa education sectors. *American Journal of Education and Information Technology*, 8(2), pp.69–72. Available at: https://doi.org/10.11648/j.ajeit.20240802.11.

Masruri, M.N., 2025. Conceptual analysis of Educational Management Information Systems for optimizing the learning process. *JEDMI: Journal of Education and Multidisciplinary Studies*, 1(1), pp.11–21. Available at: https://journal.megantaraabdinusa.org/index.php/jedmi/article/view/58

Suharno, F.S. and Imraan, M., 2024. Development of a website-based project management information system. *The International Journal of Education Management and Sociology*, 3(2), pp.67–77. Available at: https://doi.org/10.58818/ijems.v3i2.117.

Appendix

This appendix contains the guiding questions used during interviews with students, supervisors, and administrators. Questionnaires, observation and literature view questions are also included.

Appendix A: Observation

Observation Checklist:

1. How students submit proposals and reports
2. Time taken to receive supervisor feedback,
3. Communication methods used between students and supervisors
4. Record-keeping practices (manual files, emails, logbooks)
5. Challenges faced during submissions and approvals

Appendix B: Document review

Guiding Focus Questions:

1. What existing systems are used to manage final year projects in universities?
2. What challenges are associated with manual project management systems?
3. What technologies are commonly used in FYP management systems?
4. What best practices exist for supervising student projects digitally?

Appendix C: Interviews (Students, Supervisors, Administrators)

Guiding Questions:

1. How are final year projects currently managed at the University?
2. What challenges do you face in supervising or submitting final year projects?
3. How do you track project progress and milestones?
4. What delays are commonly experienced during project supervision?
5. What features would you expect in an automated project management system?
6. How do you currently store and retrieve project documents?

Appendix D: Questionnaires (Mainly Students)

Guiding Questions:

1. How do you currently submit your project proposals and reports?

2. How often do you receive feedback from your supervisor?
3. What challenges do you face during project supervision?
4. Do you experience delays in proposal approval or grading?
5. Would you prefer an online system for managing final year projects? (Yes/No)
6. What features would you find most useful in such a system?