

**DESIGN AND IMPLEMENTATION OF STUDENT PROJECT
MANAGEMENT SYSTEM FOR NILE UNIVERSITY FACULTY
OF COMPUTING**

BY

MUHAMMAD-NAZEER GARBA

201212079

**A PROJECT SUBMITTED TO THE DEPARTMENT OF
SOFTWARE ENGINEERING, FACULTY OF COMPUTING,
NILE UNIVERSITY OF NIGERIA, IN PARTIAL
FULFILMENT OF THE REQUIREMENTS FOR THE
AWARD OF DEGREE OF BACHELOR OF SCIENCE IN
SOFTWARE ENGINEERING,
NILE UNIVERSITY OF NIGERIA.**

JUNE, 2024



NILE UNIVERSITY OF NIGERIA

STATUS CONFIRMATION FOR BACHELOR'S PROJECT

DESIGN AND IMPLEMENTATION OF STUDENT PROJECT MANAGEMENT SYSTEM FOR NILE

UNIVERSITY FACULTY OF COMPUTING

ACADEMIC SESSION : 2023/2024

I, **Muhammad-Nazeer Garba** agree to allow this Degree's Project to be kept at the Library under the following terms:

1. This Degree Project is the property of Nile University of Nigeria.
2. The library has the right to make copies for educational purposes only.
3. The library is allowed to make copies of this report for educational exchange between higher educational institutions.
4. ** Please Mark (v)

☐

CONFIDENTIAL

(Contains information of high security or of great importance to Nigeria as STIPULATED under the

☐

RESTRICTED

(Contains restricted information as determined by

☐

FREE ACCESS

Approved by,

(WRITER'S SIGNATURE)

(SUPERVISOR'S SIGNATURE)

Permanent Address :

Supervisor's name:

JT Kolo road, Behind Zarumai
Model School, Minna, Niger
state.

Mrs. Hauwa Ibrahim Aminu

Date : 30th May, 2024.

Date: _____

CERTIFICATION

This is to certify that, the project titled “DESIGN AND IMPLEMENTATION OF PROJECT MANAGEMENT SYSTEM FOR NILE UNIVERSITY FACULTY OF COMPUTING” by MUHAMMAD-NAZEER GARBA 201212079 has been approved by the undersigned for meeting the requirements for the award of Bachelor of Science in Software Engineering (BSc. Hons in Software Engineering) by the department of Software Engineering, Nile University of Nigeria, Abuja.

.....

Mrs. Hauwa Ibrahim Aminu

PROJECT SUPERVISOR

.....

Dr. Ibrahim Anka Salihu

HOD SOFTWARE ENGINEERING

.....

Prof. Musa Muhammad

DEAN (FCOM)

.....

EXTERNAL EXAMINER

DESIGN AND IMPLEMENTATION OF PROJECT MANAGEMENT SYSTEM FOR NILE UNIVERSITY FACULTY OF
COMPUTING

Muhammad-Nazeer Garba

201212079

A project submitted in
partial fulfillment of the requirement for the award of the
Degree of Bachelor of Science (B.Sc.) in Software Engineering

Faculty of Computing, Department of Software Engineering
Nile University of Nigeria

JUNE, 2024

DECLARATION

I, Muhammad-Nazeer Garba declare that the project titled "Design and Implementation of a Project Management System" is my original work undertaken under the supervision of Mrs. Hauwa Ibrahim Aminu except quotations and summaries which have been duly acknowledged.

Student :

Date :

Supervisor :

Mrs. Hauwa Ibrahim Aminu

DEDICATION

I dedicate this project to the Almighty Allah for the gift of life and his immeasurable grace. I also dedicate this work to my parents for their unwavering support.

ACKNOWLEDGEMENT

I would like to express gratitude to express gratitude to my supervisor Mrs. Hauwa Ibrahim Aminu for going through this work despite her busy schedule. My appreciation also goes to my advisor Dr. Suleiman, my lecturers for all their support through the years. I would also like to recognize Mr. Peter Ntaimbu for his untiring efforts at correcting and proofreading this work and making sure I meet up with my deadlines. To my classmates and friends, your support is invaluable. Thank you.

ABSTRACT

This project talks about the design and implementation of a student project management system designed for use for Faculty of Computing, Nile University of Nigeria. The objective of this project is to develop a web application that enhances accessibility of past projects for students and an easier way for supervisors to supervise their students. The web application will consist of features such as login, download project and upload project for supervisors to correct. The project goal is to simplify looking searching for projects by students and make supervising easier. This was achieved this by using Html, CSS, and JavaScript. To ensure the system was functioning well several tests were conducted to make sure everything works correctly.

TABLE OF CONTENTS

Contents

CERTIFICATION	i
DEDICATION	iv
ACKNOWLEDGEMENT	v
ABSTRACT	vi
CHAPTER ONE	1
INTRODUCTION	1
1.1 BACKGROUND OF STUDY	1
1.2 STATEMENT OF THE PROBLEM	2
1.3 SIGNIFICANCE OF THE STUDY	3
1.4 AIM & OBJECTIVES	3
1.5 SCOPE & LIMITATION OF STUDY	3
1.6 ORAGANIZATION OF STUDY	4
1.7 DEFINITION OF OPERATIONAL TERMS	4
CHAPTER TWO	6
2.1 GENERAL INFORMATION	1
2.2 RELATED WORKS	2
2.2 SUMMARY OF REVIEW	2
CHAPTER THREE	15
3.1 SYSTEM ANALYSIS	15
3.1.1 ANALYSIS OF EXISTING SYSTEM	15
3.1.2 LIMITATION OF THE EXISTING SYSTEM	15
3.1.3 JUSTIFICATION FOR THE NEW SYSTEM	16
3.1.4 DESCRIPTION OF THE NEW SYSTEM	16
3.2 DESIGN OF THE PROPOSED SYSTEM	16
3.2.1 DATA MODEL	17
3.2.2 FUNCTIONAL REQUIREMENT	20
3.2.3 SYSTEM ARCHITECTURE	21
3.2.4 SOFTWARE STRUCTURE	23
3.2.5 WORKFLOW OF USE CASES	25
CHAPTER FOUR	24
4.1 SYSTEM REQUIREMENT FOR DEVELOPMENT	27
4.2 SYSTEMS MENUS AND IMPLEMENTATION	28

4.3 DATABASE IMPEMNTATION.....	31
4.3 SYSTEMS TESTING.....	33
CHAPTER FIVE	38
2.1 System Requirement for Development.....	38
2.2 Conclusion and Recommendations	38

LIST OF TABLES

TABLE 2.1 SUMMARY OF THE STUDY.....	9
TABLE 4.2 Functional Testing.....	9

LIST OF FIGURES

FIGURE 3.1 ENTITY RELATIONSHIP DIAGRAM	20
FIGURE 3.2 USE CASE DIAGRAM	21
FIGURE 3.3 DEPLOYMENT DIAGRAM.....	22
FIGURE 3.4 CLASS DIAGRAM.....	24
FIGURE 3.5 ACTIVITY DIAGRAM	26
FIGURE 4.6 ACTIVITY DIAGRAM	29
FIGURE 4.7 ACTIVITY DIAGRAM	30
FIGURE 4.8 ACTIVITY DIAGRAM	30
FIGURE 4.9 ACTIVITY DIAGRAM	31
FIGURE 4.10 ACTIVITY DIAGRAM	32
FIGURE 4.11 ACTIVITY DIAGRAM	32
FIGURE 4.12 ACTIVITY DIAGRAM	32
FIGURE 4.13 ACTIVITY DIAGRAM	35
FIGURE 4.14 ACTIVITY DIAGRAM	36
FIGURE 4.15 ACTIVITY DIAGRAM	27

CHAPTER ONE

INTRODUCTION

This chapter provides an overview of the project, highlighting the background of the study, problem statement, aim and objectives, scope and limitations, and significance of the study. This chapter sets the context of the research work and establishes the need for developing a project management system for Department of Computing of Nile University.

1.1 Background of the Study

Technology has made incredible progress in the past years, particularly in the field of Education [1]. As a result of technological advancements, our world has transformed dramatically in terms of living, working, and learning. Technology has revolutionized traditional teaching and learning methods in the education sector. One such technological advancement of the 21st century is the internet. The internet provides access to a wide range of online resources useful for independent learning and research [1].

Student projects are documents students create after thoroughly researching a particular topic. Projects excellently demonstrate students' creativity and knowledge gained. Project writing is a major requirement for final year students, evaluating their skills and experience over their university. Therefore, students must carry out and complete a research study or project to graduate. The project spans two semesters - first semester for writing and defending the chosen topic, and second semester for system development based on the selected topic. Students undertake research projects as part of their final year assessments to demonstrate their ability to conceptualize, organize, and solve problems. Graduating students should master research skills and approaches for solving societal issues. A research study can be described as the process of discovering new knowledge [2]. New knowledge can be generated, or existing knowledge creatively applied to

develop new concepts, methodologies, and understandings [2]. A.V. Afanasev et al. define research as the careful and detailed study of a specific problem, concern, or issue [2].

Project management involves applying knowledge, skills, and tools to project activities to meet specific project goals. Key components are project initiation, planning, execution, monitoring, control, and closing. Before committing to a project, a feasibility study should be undertaken to establish well-defined goals and expected benefits. Tools for project management include Gantt charts, project network diagrams, and critical path analysis.

The Student Project Management System is software specifically designed to simplify and enhance project management for both students and supervisors.

1.2 Statement of the Problem

The idea for the Student Project Management System arose from experiences in the school library. Going through multiple projects to find the specific one was a tedious task. An online system archiving and enabling search across all projects would make the process more efficient. Furthermore, supervisors could use such a system to remotely monitor and manage students, rather than students having to locate supervisors around campus.

1.3 Significance of the Study

The implementation of this project will assist the faculty in managing student project. It will provide a platform where students can readily obtain supervisor feedback, without

having to physically meet their supervisors. Students will also have access to prior works and projects posted on the platform, enabling them to learn from and build on previous efforts. Overall, the system will boost faculty coordination and oversight of student research and project work.

1.4 Aim and Objectives

The aim and objectives of this project is to create a web application for the Faculty of Computing that allows project supervisors to effectively monitor the progress of students while also giving students access to a project repository.

Objectives:

1. To develop a student project management web application that can be accessed from any device for Nile University of Nigeria's Faculty of Computing.
2. To develop a system that makes supervision of student's research work in Faculty of Computing easier.
3. To test and evaluate reliability of the system.

1.5 Scope and Limitation of the Study

The scope of this project is to develop a web application that can be used by both students and lecturers at Faculty of Computing, Nile University of Nigeria. In addition to browsing past student projects and download them, the lecturer will be able to monitor students' projects using the proposed system.

Limitations: The system will rely on internet connection. It will be limited to the Faculty of Computing; Nile University of Nigeria and the system will only accept .pdf files

1.6 Organization of the Study

The project is organized as follows:

1. Chapter One Introduction: Explain the significance of student project management system and the reasons why a university would implement one. Talk about the study's goals and the project's goals and scope.
2. Chapter Two Literature Review: Conduct a thorough review of the academic and research literature on student project management system, including their, design, implementation, and security measures. Draw insightful conclusions about the strengths and areas of that need improvement and summarize the major findings.
3. Chapter Three System Analysis and Design: Create a comprehensive design for the online voting system, including the database design, user interface, and architecture. During the design phase, keep important factors like scalability, security, and user-friendliness in mind.
4. Chapter Four Implementation and Testing: To ensure the project management system performs well and satisfies the usability requirements. It is implemented and tested using the appropriate tools and programming languages.
5. Chapter Five Conclusion and Recommendations: This talks about the importance to further research on the project management system. This chapter has conclusions from the literature review and suggestions for further study in the field together with a summary of the study's main conclusions.

1.7 Definition of Operational Terms

1. Student: A person that is currently studying at a university or any place of higher education.

2. Project: A research in which a student analyses information to be used for educational purposes.
3. Project Management: It is the applying of processes, methods, and skills to achieve specific project objectives.
4. Supervisor: It is someone that oversees a project of its inception to completion.
5. Web application: It is an application software that is accessed through a web browser.
6. Repository: It is a data structure that stores metadata for a set of files.
7. Research: It is the organizing and analysis of data to increase understanding of a topic

CHAPTER TWO

LITERATURE REVIEW

This chapter provides a review of existing literature related to the management of student projects. The review encompasses research on the challenges faced in capstone projects, prior work on digital systems for managing such projects, key features and functionalities that have been implemented, development methodologies utilized, and overall outcomes or assessments.

2.1 General Information

Student Project Management Systems come in different forms. There are systems that archive students' final year projects for other students to view. Another allows supervisors and students to communicate and make corrections to their projects [3]. These systems do away with the necessity for printing projects and storing them. The systems allow students access to multiple projects without the need to go to the school library. Student project management system can be developed using various technologies including HTML, PHP, MySQL and CSS.

2.2 Related Works.

According to A.V. Afanasev and Natalia Gorkushkina in 2023 [1], they propose a system that allows for management of students' projects in the information technology (IT) sphere, allowing coordination, monitoring, and evaluation of student contributions. The system allows for the coordination of actions between students of different courses and groups fostering a collaborative learning environment. It tracks and evaluates the progress of students' work and their contributions.

Yahya L in 2023 [2] describes the development of a web-based system that automate the management of final year projects. The goal was to automate the management and allocation of final year student projects in the Department of Mathematics at Gombe State University in Nigeria. The system allows students to submit project topics. The coordinator then verifies topics against past projects to prevent duplication before allocating topics. Supervisors can also view allocated topics and students. The system was developed using PHP, MySQL, JavaScript, HTML and CSS. The waterfall model was the methodology used for developing the system.

In 2017, Soyemi J and Isinkaye F [3] developed a web-based system to detect duplication in final year student projects. It was developed for the department of Computer Science at The Federal Polytechnic in Ilaro, Nigeria. The aim was to reduce instances where students copy projects from previous years without doing original work. The system allows supervisors to upload and access previous student projects to check for duplication. The system was developed using HTML, JavaScript, CSS, PHP, and MySQL.

Nwangwu E and Asuquo G [4], developed a computerized project management system to improve student research and final year projects at public universities in Awka Ibom State, Nigeria. The goal was to transform the manual approach to managing student projects which has various drawbacks. The study used a research and development design with four phases: needs assessment, system development, validation, and trial testing. The system was developed using Agile software development model.

In 2018, Malik S et al[5] developed a web-based system called “GPOrganize” to help manage and archive final year projects (FYP) for students in the Faculty of Computing and Information Technology, Rabigh at King Abdulaziz University. The GPOrganize system was aimed to facilitate supervisor communication, provide access to previous projects, give guidance/templates to students, and manage schedule. Open-source technology like XAMPP was used to create a prototype system. Interfaces were created for functions like creating student groups, assigning projects, uploading reports, communication tools etc.

In 2021, V. Vijayakumar[6] developed a software called “ProTrack”. It combines project management and student management systems, providing an industrial workspace for students to work on innovative projects and allowing teachers to support and supervise their teams.

Imed Romdhani et al proposed an integrated online supervision system for final year and dissertation projects in 2011[7]. It was achieved by conducting surveys on undergraduate students and academic staff at Edinburgh Napier University to collect feedback on current supervision practices and ideas/requirements for proposed online system. The survey feedback was used to identify key features to implement in an initial prototype of the online Student Project Performance Management System.

Chikwendu et al in 2021[8] proposed web-based student project management system for final year students at Akanu Ibiam Federal Polytechnic Unwana. The aim was to provide a system to mitigate challenges like poor quality project, duplication, prolonged research writing, and late submissions that exist with current manual processes. The system automates activities like project topic listing/selection, approvals, writing guidelines, students/supervisors

communication, assessments, and report submission/management. It uses technologies like Java, Apache Tomcat, and MySQL to eliminate duplication by automatically rejecting already done projects, and ensure timely, quality project completion.

Bhatt, Rahul et al proposed a system in 2008[9]. The system was a Student Project System software to help students manage their college projects. The system has separate student and mentor modules for students to create/edit projects and mentors to access assigned projects. Key features of the system are user login, project creation/editing/deletion, assigning tasks to group members. The system was developed using HTML, CSS, MySQL, and Ruby on Rails.

2.3 Summary of the Review

Table 2.1 below gives a summary of some of the literature reviewed related to design and implementation of a Student Project Management System.

Table 2.1: Summary of the Review

Title of Paper/Year	Author	Methodology	Summary of Result	Limitation
Web-based Archive Management and Student Guidance for Final Year Project [5]	S. Malik, S. Abdulla Al-youbi et al	The method used to make this system was prototyping. The software tool used was an open-source technology	This project was able to create an interface that allows students to be added to the group. It also allows for project	This system did not make monitoring of students by supervisors possible

		called XAMPP.	supervisors to send messages to the students.	
Final Year Supervision Management System as a Tool for Monitoring Computer Science Projects[10]	M. A. Bakar, N. Jailani, Z. Shukur, and N. F. Mohd Yatim	The web-based system was developed using PHP, Dreamweaver CS3 for user interface, and MySQL for the database system.	A prototype was developed which can create modules for the student and a lecturer profile for the supervisor. These are used to make appointments.	This system does not archive student project.
International Conference on Multimedia Communications: Mediacom[11]	International Communication Sciences Association	An iterative approach was taking in developing this system	This system was able to allow students to login and get guides from their project supervisor. It has modules for user registration and login, documentation upload and download and	This system does not archive projects.

			progress report generation	
Development of Computerized Students' Project Management System (CSPMS) for Enhancing Students' Research in Public Universities in Akwa Ibom State[12]	E. C. Nwangwu and G. E. Asuquo	Agile methodology was used in creating this system. A questionnaire was also used for data collection. Java script, XAMPP, HTML, and CSS were the tools used for developing the system.	The system was validated by ICT experts and evaluated the system and approved its functional elements. Successful registration of students' records, functional login, easy and fast downloading and uploading of projects, simple and flexible change of user records, and error-free approval of projects.	This system does not allow archiving of student's project
Final Year Student Project Allocation	A. Adamu	The system was developed using	The developer of this system was	The system did not allow students

and Managing System [13]		Hypertext Preprocessor (PHP) programming language, ASP.NET for the graphical user interface and MySQL for database.	able to create a system where supervisors can accept or reject topics from students. Additionally, students can submit projects, view projects, request changes to project topics, and schedule meetings with supervisors.	to chat with their supervisors on the system
Student Project Management System [14]	N. Soms	HTML5 and CSS were used to develop the design for the system. JavaScript and jQuery were used for client-side scripting. PHP and MySQL were used for the back end	The system developed had key features like student registration, login authentication, group formation and communication through email alerts for project-	The system does not allow archiving for other students to viewing.

			related notifications.	
Students Final Year Project Management System [15]	J. T. Fakoya, M. A. Ibiyomi, and A. Akeem	Visualbasic.net and Microsoft Access were used to create this system. Visualbasic.net allows developers to target windows, web, and mobile devices. Microsoft access was used for the database.	The result of the developed system is registration of students, uploading completed students project and downloading existing student project.	The system does not allow project supervisors to communicate with each other.
Final Year Students' Projects Allocation and Management System [16]	L. Yahaya	Waterfall model was used in creating this system. HTML, PHP, CSS, MySQL, and WAMP server.	The project aimed to automate project topic allocation and verification replacing the manual method	The system does not make possible communication between students and supervisors

Research Gap

From the existing system reviewed there have been attempts at developing a project management system. Most of the existing software has focused on project allocation,

archiving of projects, group project communication and supervision of projects. The system that is being developed will allow for archiving of projects which students can use for their project research and allows supervision between supervisors and students. This system is not available in Nile University of Nigeria.

CHAPTER THREE

SYSTEM ANALYSIS AND DESIGN

This chapter will talk about the System Analysis and Design for the proposed topic. System analysis and design is replacing an existing system by defining its components. The issues with the current system will be addressed in the design of the system. The chapter will also discuss the system architecture and user interface design of the system.

3.1 System Analysis

System analysis is the detailed examination and study of a current system architecture, with the goal of identifying opportunities for improvement. It involves investigating how an existing system functions in terms of its component elements and interactions, to determine requirements and guide changes for enhancing overall performance through a new or upgraded system.

3.1.1 Analysis of the Existing System

Student Project management is currently done manually in Nile University of Nigeria. Hard copies of projects are received, sorted, and processed by each department in the university and then stored in the library.

With the existing system students have to search around the university for their supervisors, and sometimes their supervisors will not be available on campus which could make the process frustrating.

3.1.2 Limitation of the Existing System

- i. Projects are not always available in the library.
- ii. It is time consuming to look through multiple projects.

- iii. The library is not always open.
- iv. Inefficient office space management

3.1.3 Justification for the New System

- i. Projects can be easily accessed as it will be on the web.
- ii. Supervisors can easily monitor their students' work.
- iii. The supervisor can give prompt feedback to the students.
- iv. It is easy to operate and maintain.
- v. The danger of losing projects is lessened.

3.1.4 Description of the New System

The system will have three main users: students, supervisors, and the admin. The admin is the user that has full access to the system. The admin will create accounts for users. They will be able to upload projects onto the repository. The admin will also register new students and supervisors. Overall, all system monitoring will be done by the admin. The students will be able to login and view projects uploaded by the admin. Students can search for specific projects in the repository. They will also be able to view their supervisors' comments. Supervisors will be able to login and manage assigned students, and review student project submissions. Supervisors can also provide feedback to students and track their progress. The system will be accessible through web browsers. All projects will be stored in a central database for easy search and retrieval.

3.2 Design of the Proposed System

A web application for archiving and viewing student's projects. Supervisors can also monitor their students. The proposed system will be developed using HTML, CSS, JavaScript, Node.js, and MySQL.

3.2.1 Data Model

In figure 3.1 below, the 'admin' entity has six (6) attributes: 'admin_ID', 'user_name', 'password', 'first_name', 'last_name' and 'email'. The 'student' entity has six (6) attributes: 'student_ID', 'user_name', 'password', 'first_name', 'last_name' and 'email'. The 'supervisor' entity has six (6) attributes: 'supervisor_ID', 'user_name', 'password', 'first_name', 'last_name' and 'email'. The entity 'project' has three attributes: 'project_ID', 'project_name', and project_description. The last entity 'comment' has three attributes: 'comment_ID', 'comment_Text', 'date'. These entities will have several relationships between each other.

- i. Admin and Project – An admin can upload many projects. This would be a one-to-many relationship.
- ii. Students and Project – Students can view projects, these projects are uploaded by the admin. This would be a many-to-many relationship.
- iii. Student and Search project – This is a functionality so there is no direct relationship.
- iv. Student and supervisor comment -Students can view supervisor comments on their project. This would be a one-to-many relationship.
- v. Supervisors and Student – One supervisor can manage multiple students. This will be one-to-many relationship.
- vi. Supervisors and Project submissions - One supervisor review multiple projects submissions. This will be a one-to-many relationship.

- vii. Supervisors and Students - One supervisor provides feedback to multiple students. This will be a one-to-many relationship.
- viii. Supervisors and Student progress – This is a functionality so there is no direct relationship.
- ix. Projects and Comments: One project can have many comments and each comment is for one project. This will be a many-to-one relationship.

TRANSFORMING THE ER DIAGRAM TO RELATIONAL ALGEBRA

Relational algebra:

- i. Admin (admin_ID, user_name, password, first_name, last_name, email)
- ii. Student (student_ID, user_name, password, first_name, last_name, email)
- iii. Supervisor (supervisor_ID, user_name, password, first_name, last_name, email)
- iv. Project (project_ID, project_name, project_description)
- v. Comment (comment_ID, comment_Text, date)

NORMALIZATION

1. 1NF: To transform the ER diagram to 1NF, the tables need to have atomic values and no repeating groups. 1NF needs every attribute in a table to contain atomic values as seen below.
 - i. Admin (admin_ID, user_name, password, first_name, last_name, email)
 - ii. Student (student_ID, user_name, password, first_name, last_name, email, supervisor_ID (Foreign key))
 - iii. Supervisor (supervisor_ID, user_name, password, first_name, last_name, email)

- iv. Project (project_ID, project_name, project_description, admin_ID (Foreign Key), supervisor_ID (Foreign Key), student_ID (Foreign Key))
 - v. Comment (comment_ID, comment_Text, comment_Date, comment_Month, comment_Year, supervisor_ID (Foreign Key), student_ID (Foreign Key), ProjectID (Foreign Key))
2. 2NF: To transform the ER diagram to 2NF, we need to make sure there are no partial dependencies. In the given ER diagram, all tables have all the non-key attributes fully dependent on the primary key. The ER diagram is already in 2NF.
3. 3NF: To transform the ER diagram to 3NF, we need to make sure there are no transitive dependencies. Transitive dependency occurs when a non-key attribute depends on another non-key. In the table there is a transitive dependency on the entity 'Student' where 'first_name' will depend on 'email'. To resolve this, we create a new entity for student details:
- i. Admin (admin_ID, user_name, password, first_name, last_name, email)
 - ii. Student (student_ID, user_name, password, first_name, last_name, supervisor_ID (Foreign key))
 - iii. Student Details (student_ID, email)
 - iv. Supervisor (supervisor_ID, user_name, password, first_name, last_name, email)
 - v. Project (project_ID, project_name, project_description, admin_ID (Foreign Key) supervisor_ID (Foreign Key), student_ID (Foreign Key))

- vi. Comment (comment ID, comment_text, date, supervisor_ID (Foreign Key), student_ID (Foreign Key), project_ID (Foreign Key))

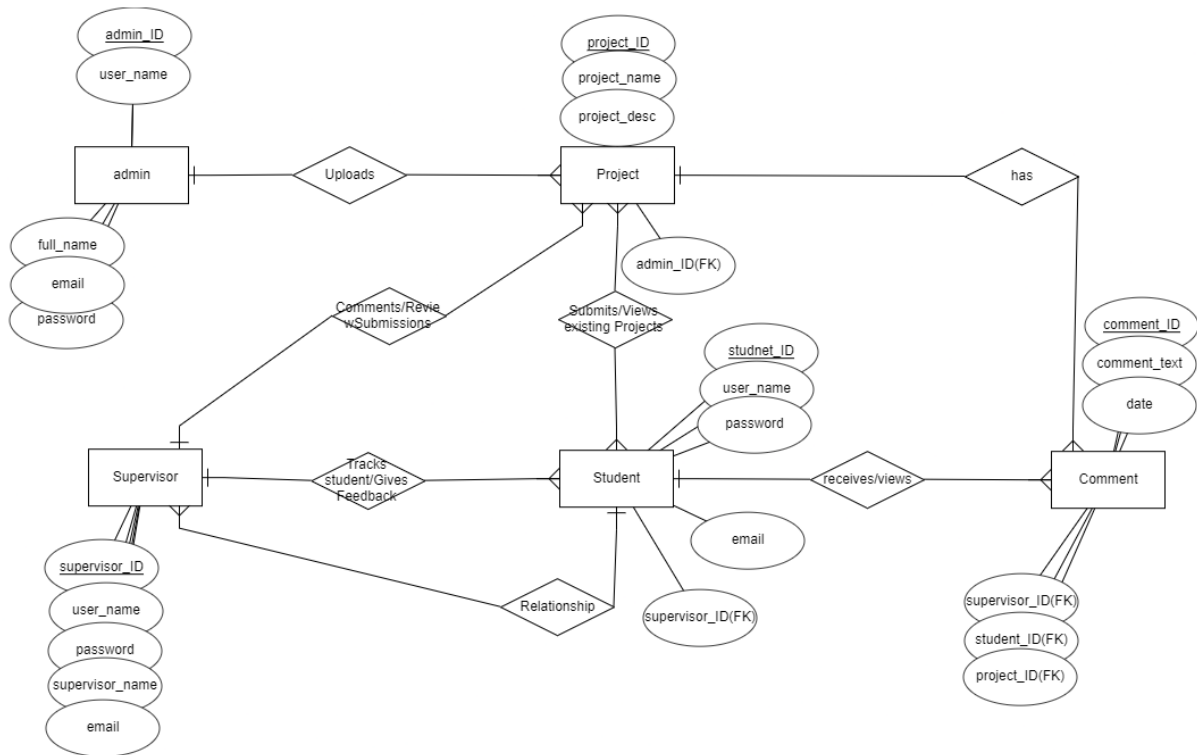


Figure 3.1: ER Diagram

3.2.2 Functional Requirement

In Figure 3.2, there are three users' actors which are: student, supervisor, and the admin. The admin has two use cases which are: "login" and "upload project". The use case "login" refers to the admins ability to access the system with an account. The use case "upload project" portrays the ability of an admin to upload projects on to the system. The users will be able to register.

The student has four use cases which are: "login", "upload project", "search project", "download project". The use case "login" refers to the student's ability to access the system with an account. The "upload" use case portrays the ability to upload projects

on the system for supervisors to see. The “search project” use case portrays the ability to search for projects uploaded by the admin. The “download” use case portrays the ability to download projects uploaded by the admin.

The supervisor has two use cases which are: “login”, and “correct project”. The use case “login” refers to the supervisor’s ability to access the system with an account. The “correct project” use case refers to the ability of a supervisor to correct projects by making comments on projects uploaded by students. The supervisor will be able to correct whatever their student post.

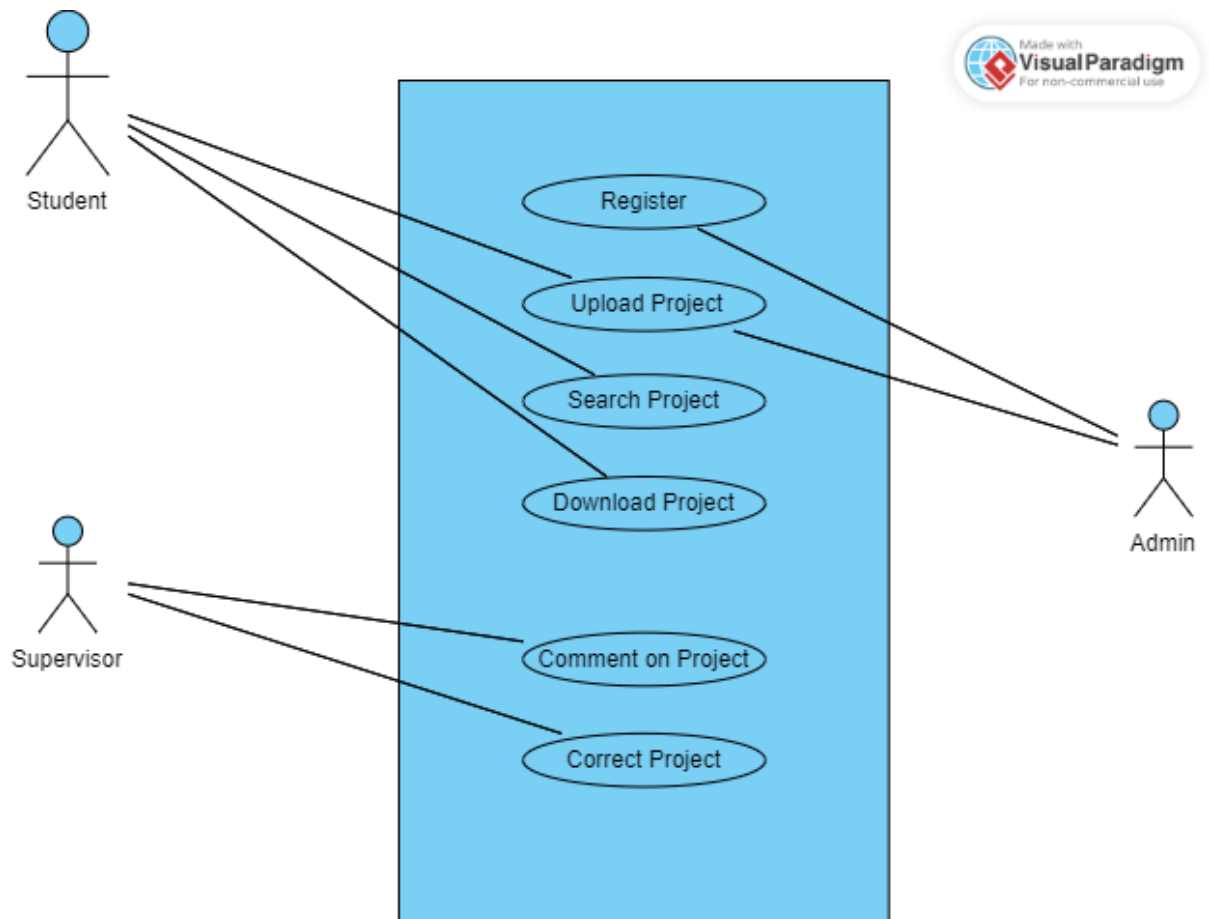


Figure 3.2: Use Case Diagram

3.2.3 System Architecture

In Figure 3.3 below, the deployment diagram consists of:

- i. Client: The client is the device that users will use to access the student project system. It can be a desktop, tablet, laptop, or smartphone.
- ii. Web Server: The web server oversees providing web pages to clients and managing HTTPs requests. It connects to the database to obtain and save data.
- iii. Database: The database serves as a repository for storing the projects uploaded by the admin. It is also used to store the login details for the users.

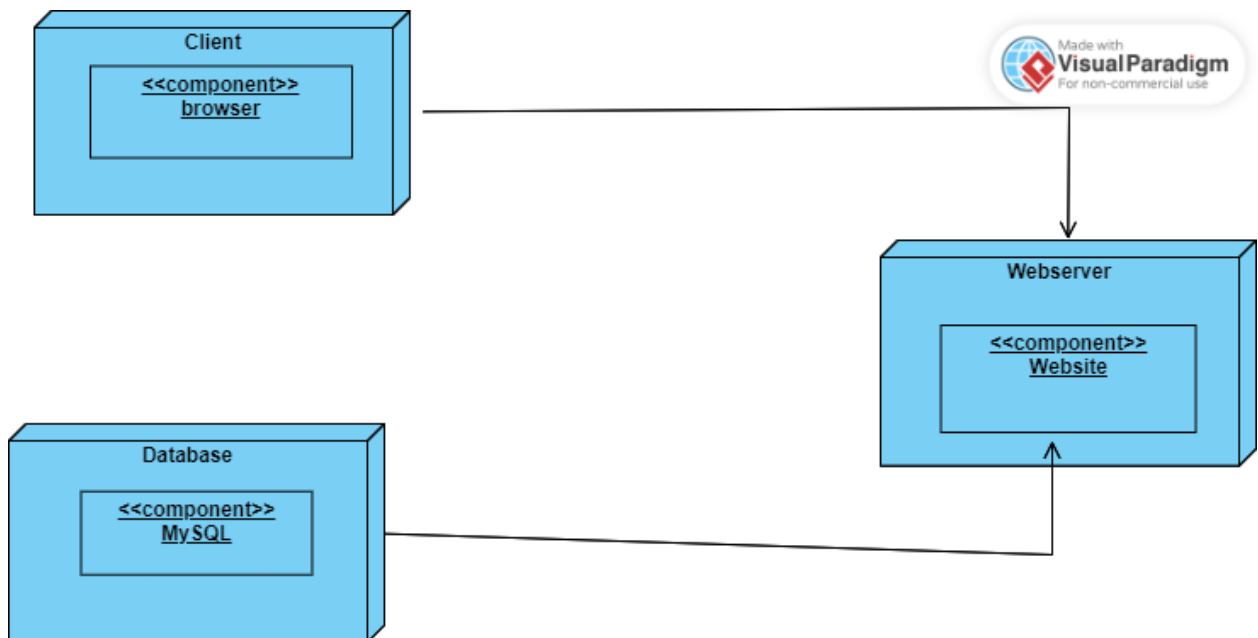


Figure 3.3: Deployment Diagram

The relationships between these components are as follows:

- i. Client to Web Server: The clients interact with the web server through network communication, typically utilizing the HTTPs protocol. This enables users to access the system.
- ii. Web Server to Database: The web server communicates with the database to retrieve relevant data. The interaction between the web server and the database

usually occurs using a database protocol like SQL. This ensures proper management and storage of data.

These relationships ensure the system functions effectively. It also safeguards the integrity of the data.

3.2.4 Software Structure

Here are the relationships between the classes in 3.4:

- i. In the class diagram there is a relationship called “association”. This relationship is denoted with a line connecting the two entities with a “1” or “*” at the end of the line. The classes with this relationship are:
 - a) Admin - Project
 - b) Student - Project
 - c) Student - Comment
 - d) Supervisor - Student
 - e) Supervisor – Project
- ii. The other relationship in the class diagram is “aggregation” which is represented by a line with an unfilled circle on one end. This shows that one class contains or owns the other class as a part of it. The classes with this relationship are:
 - a) Project - Comment

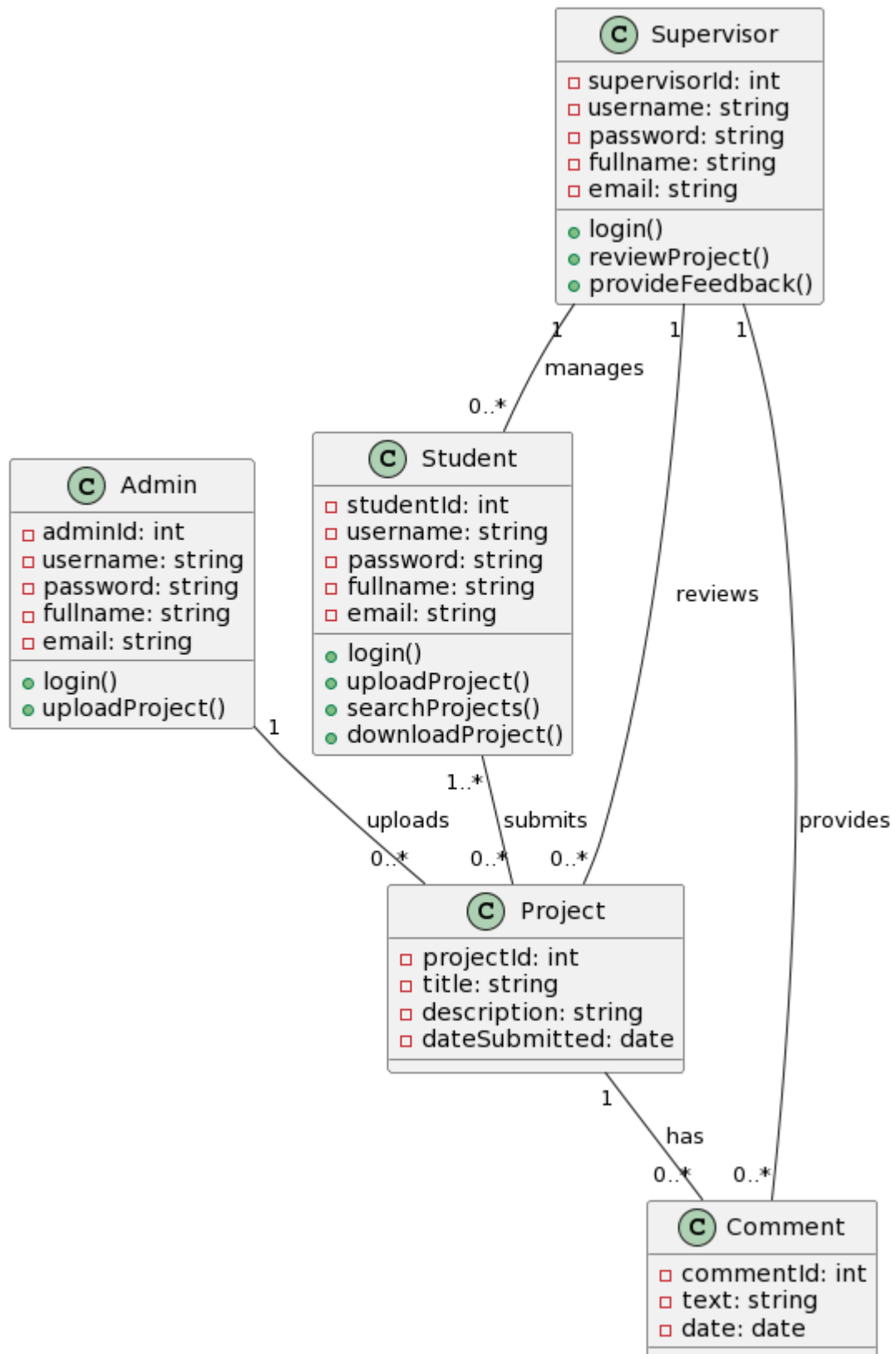


Figure 3.4: Class Diagram

3.2.5 Workflow of Use Cases

In Figure 3.5, the admin logs into the system. After logging into the system, the admin clicks the “upload project” option. The system displays a project form. The admin fills in the form with details of the project and attaches the project file. The admin clicks submit. The system validates the details, if valid the system saves the project information and file in the database. The system shows a message saying the upload was successful.

The student logs into the system. After logging the student can “search project” or “upload project”. When the student clicks on “search project” the system displays a search form. The student enters keywords to search for a project. The student clicks search. The system searches the database based on the keywords given. The system displays the search results list. If the project exists, the student will click on the project to view details. If the student wants to keep a copy of the project, they can download it to their local device. After downloading or viewing, they can log out to exit the system. If “upload project” is clicked, a window will open asking them to attach the project. After attaching the project, it will be uploaded for their supervisor to review and correct. When done the students will log out.

The supervisor logs into the system. After logging into the system, they can view their assigned students. They can select a student and review their project submissions. Once reviewing a project, the supervisor can add feedback, comments, and corrections to provide guidance to the student.

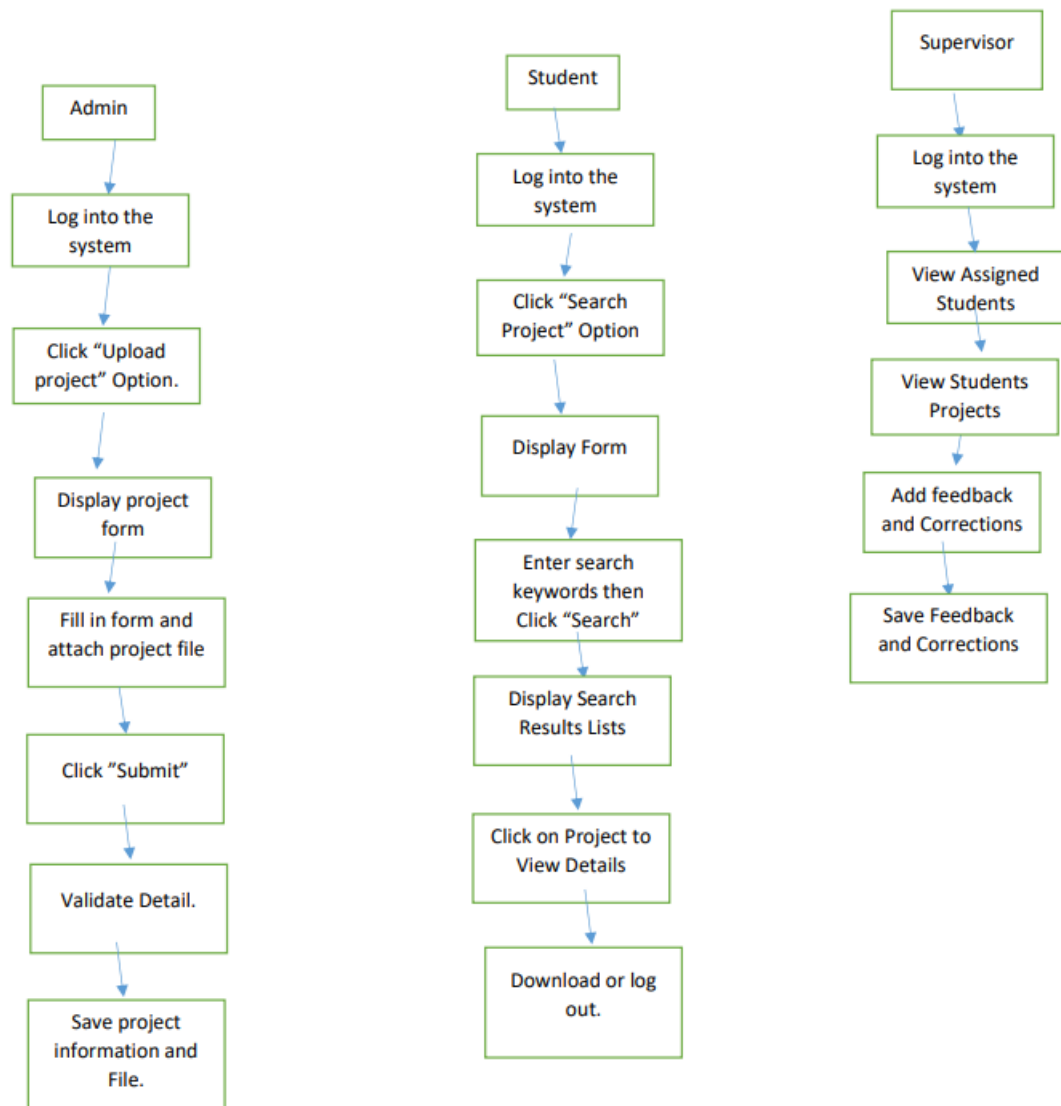


Figure 3.5: Activity Diagram

CHAPTER FOUR

IMPLEMENTATION AND DISCUSSION

This chapter deals with developing and implementation of the proposed system. The chapter comprises of the following sections: system development requirements, system menu implementation, database implementation and system testing.

4.1 System Requirement for Development

To create a successful framework for the implementation of student project management system, a list of system requirements was developed. These requirements include functional and non-functional requirements as well as hardware and software. A server, an effective database system and some client devices like laptops and smartphones are required to access the web application. A dependable web server, a functional operating system and an appropriate programming language were needed as the software requirements. Core features such uploading of projects by admin, login capabilities, access to uploaded project by students and supervisors commenting on their student's project are example of functional needs. Non-functional requirements include maintaining system reliability, high security measures and scalability to accommodate varying user loads. Addressing these requirements thoroughly allowed for the development of the system to proceed effectively and efficiently.

SOFTWARE REQUIREMENTS

- Operating System: Windows
- Web Server: Nodejs to manage incoming requests from client devices.
- Database Management System: MySQL to store and manage user data.
- Programming Language: HTML, JavaScript, and CSS to design the system's functions.

4.2 System Menus Implementation

The system menus are what the users use to interact with the system. The construction of an interface that is user-friendly and easy-to-navigate is part of the system menu implementation. The main system menus include login page, student home page, supervisor home page and admin home page.

- Login page: The login page depicted in Figure 4.7 allows to access the system by entering their username and password. The user will be forwarded to the main page. The main page will differ depending on whether the user is a student, supervisor, or administrator. An error message will be displayed if the user enters a username or password that is not correct.

Login

Username:

Password:

Admin ▼

Login

Figure 4.6: Login Page

- **Admin Page:** This is the page that the user will be taken to if they login with the credentials of an admin. The admin uploads projects for students to see on their own end. Figure 4.7 below shows how the admin page looks.

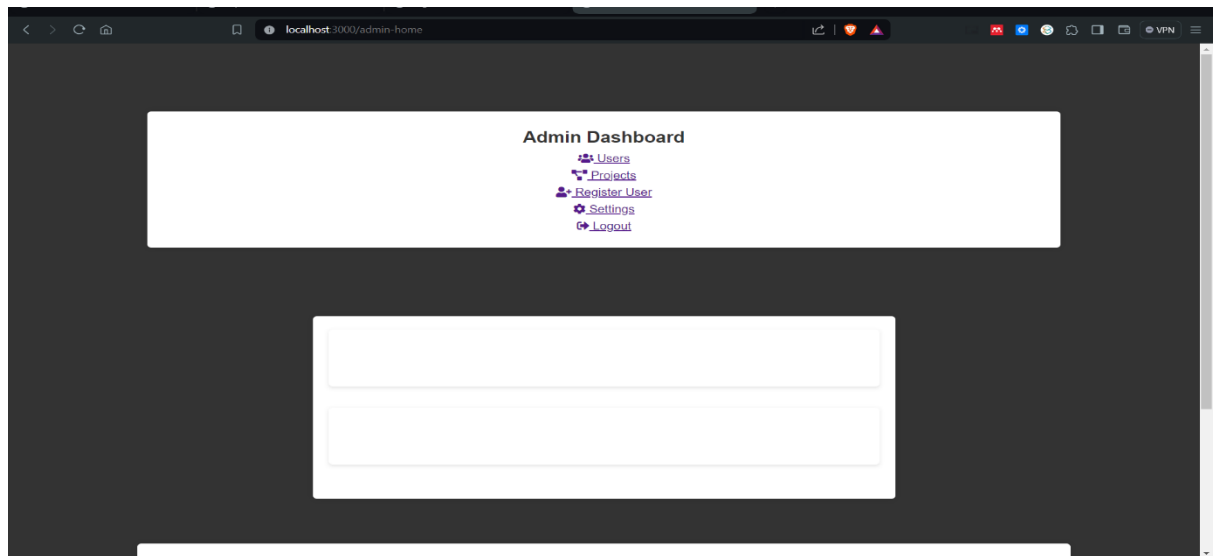


Figure 4.7: Admin Page

- **Student Page:** This page allows students to see projects uploaded by the admin. Users can access this page by login in with student credentials. Students can also see corrections done by their supervisor. Figure 4.8 shows how the student page should look.

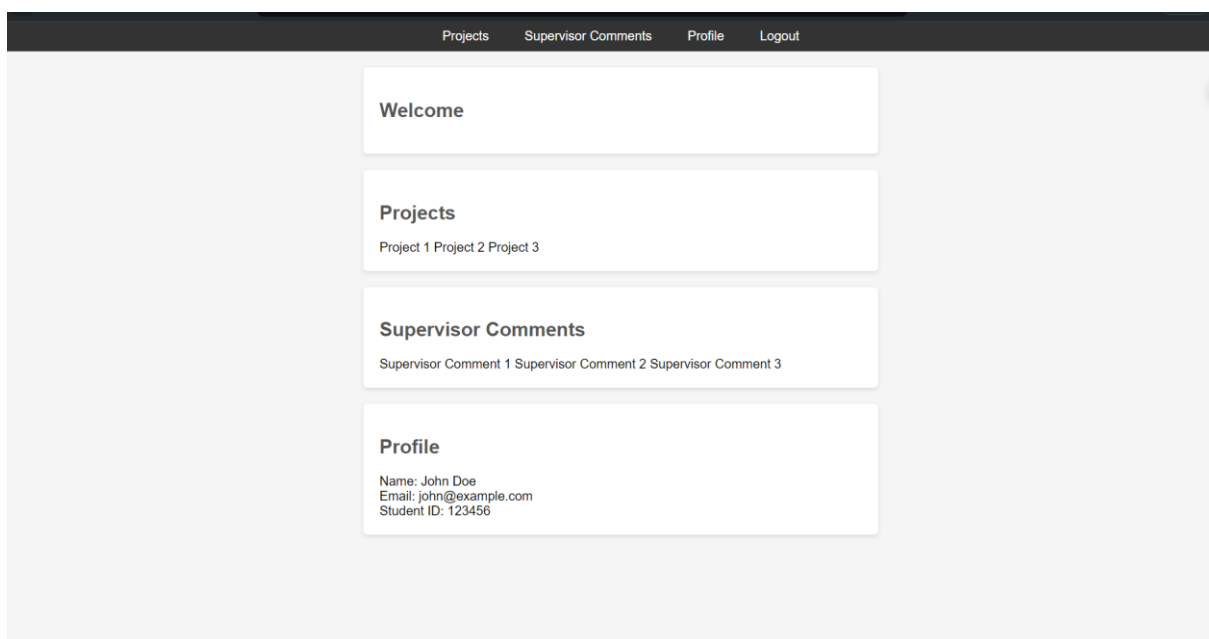


Figure 4.8: Student Page

- **Supervisor Page:** This page allows supervisors to keep track of their students and make corrections to their projects. This page is accessed by users who have the login credentials for supervisors. Figure 4.9 illustrates how the page looks.

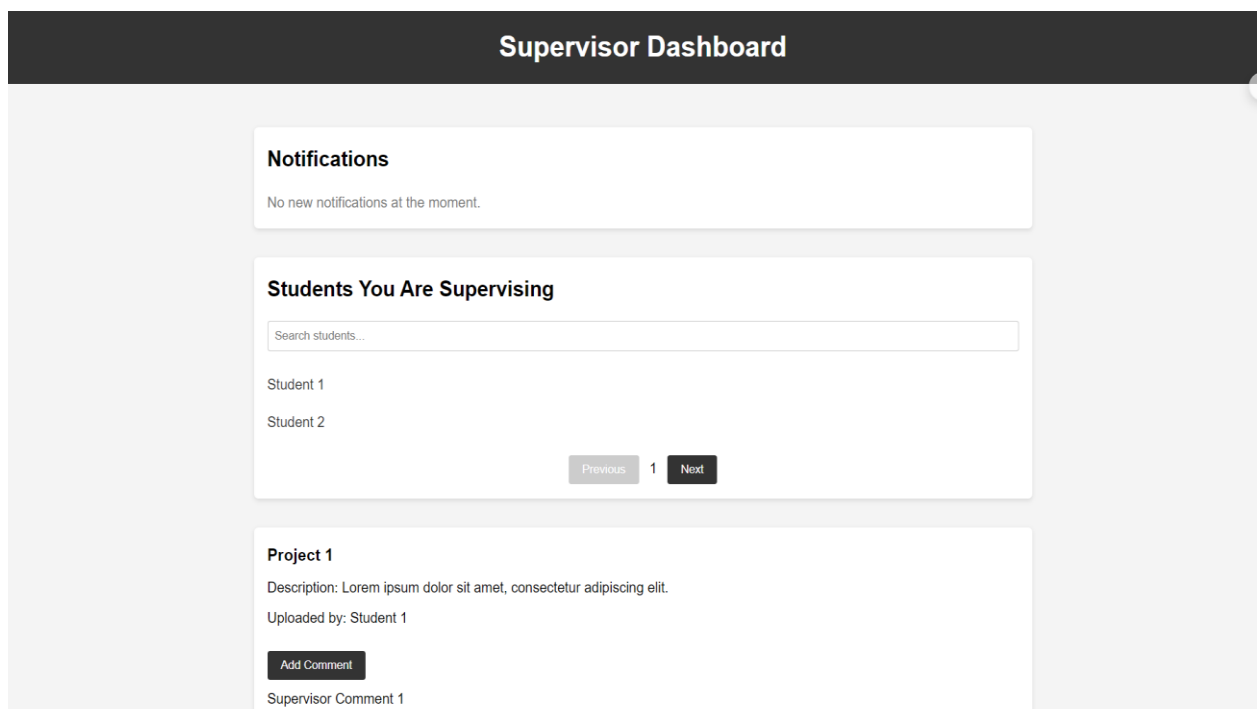


Figure 4.9: Supervisor Page

4.3 Database Implementation

Database implementation is important when developing a web application. It serves as a central repository for storing and managing essential data associated with the system. This includes admin credentials, student credentials and supervisor credentials. Database

implementation involves creating tables, defining the appropriate data types for each attribute and establishing the correct relationship between tables.

- Admin data model: This model shown in Figure 4.10 represents an admin of the system. It has attributes like admin_id, user_name, password, first_name, last_name and email.

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	admin_ID	user_name	password	first_name	last_name	email
▶	1	admin1	adminpassword1	Admin	One	admin1@example.com
*	NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.10: Admin Data Model

- Student data model: This model shown in Figure 4.11 represents the students that will login into the system. It consists of attributes like student_id, user_name, password, first_name, last_name, supervisor_id and email.

student_ID	user_name	password	first_name	last_name	supervisor_ID	email
1	student1	studentpassword1	Student	One	1	student1@email.com
6	abc	123	John	Doe	1	123@email.com
NULL	NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.11: Student Data Model

- Supervisor data model: This model shown in Figure 4.12 represents the supervisors that will use the system. It consists of attributes like supervisor_id, user_name, password, first_name, last_name and email.

Result Grid

Filter Rows:

Edit:

Export/Import:

Wrap Cell Content:

	supervisor_ID	user_name	password	first_name	last_name	email
▶	1	supervisor1	supervisorpassword1	Supervisor	One	supervisor1@example.com
	NULL	NULL	NULL	NULL	NULL	NULL

Figure 4.12: Supervisor Data Model

4.4 System Testing

Testing the software is an important phase in the development process since it helps detect and resolve any errors in the system. System testing involves both functional and non-functional aspects to assess the system's performance based on predetermined standards. Functional testing involves tasks like logging in, uploading projects, viewing projects etc. Functional testing ensures that different parts of the system work correctly. While non-functional involves testing reliability, usability, efficiency, and security. Table 4.2 below summarizes the functional testing for the project management system, including their descriptions, expected outcome, actual outcome, and results.

Table 4.2 Functional Testing

Test Case	Description	Expected Outcome	Actual Outcome	Pass/Fail
Admin Login	Login using valid credentials	User is successfully logged in	User is successfully logged in	Pass
Student Login	Login using valid credentials	User is successfully logged in	User is successfully logged in	Pass
Supervisor Login	Login using valid credentials	User is successfully logged in	User is successfully logged in	Pass

Upload Project	Upload of project to the repository	The project is uploaded successfully	The project is uploaded successfully	Pass
Admin Login	Login using invalid credentials	Error message is displayed	Error message is displayed	Pass
Student Login	Login using invalid credentials	Error message is displayed	Error message is displayed	Pass
Supervisor Login	Login using invalid credentials	Error message is displayed	Error message is displayed	Pass

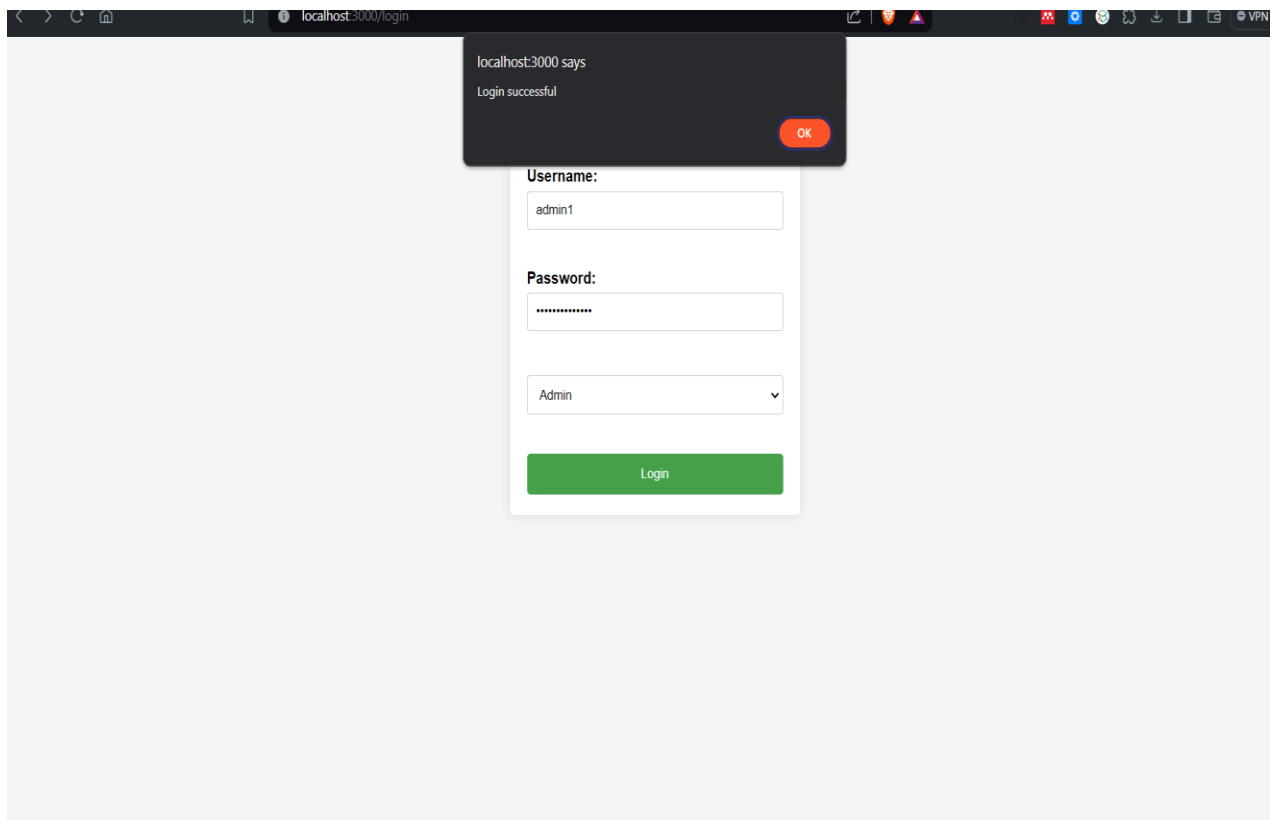


Figure 4.13: Admin Login Successful

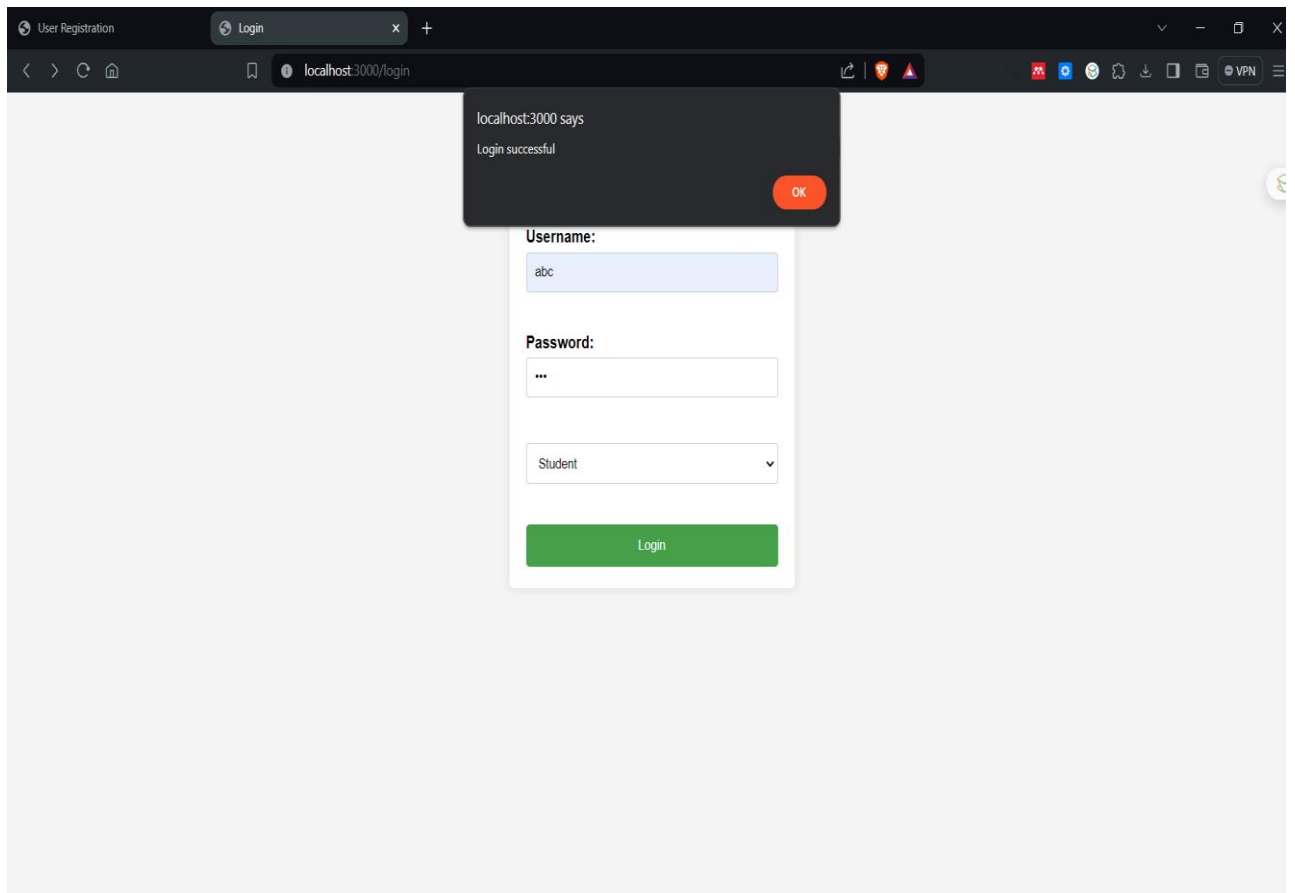


Figure 4.14: Student Login Successful

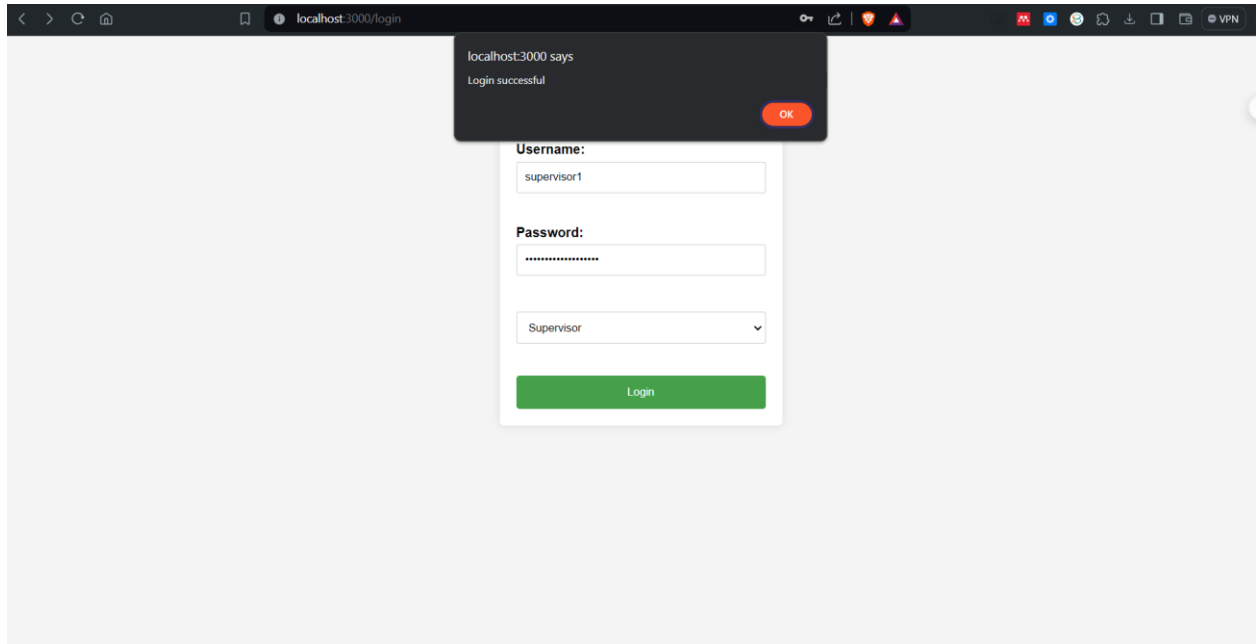


Figure 4.15: Supervisor Login Successful

CHAPTER FIVE

CONCLUSION AND RECOMMENDATION

An overview of the research on the student project management systems in Nile University, Faculty of Computing. This chapter contains inferences made from the data as well as advice on how to successfully implement these systems.

5.1 Summary of the study

The proposed student project management system for the Faculty of Computing at Nile University aims to provide an efficient platform for archiving student projects, enabling supervisors to effectively monitor student progress, and giving students easy access to past project work. By conducting a thorough literature review, this work explored existing systems, their features, development approaches, and overall assessments to gain insights into best practices.

5.2 Conclusion and Recommendations

The implementation of a student project management system could greatly benefit the Faculty of Computing at Nile University. By moving from traditional manual processes to a centralized online system, students would gain convenient access to a repository of past projects, this can foster knowledge-sharing and serve as reference. Furthermore, supervisors would be able to streamline the process of reviewing project submissions, providing feedback, tracking progress, and communicating seamlessly with their students.

The web-based systems eliminate the need to physically store project documents, optimizing office space management and mitigating loss or misplacement risks. Additionally, the

system's accessibility from various devices with internet connections enhances its practicality and convenience for both students and supervisors.

Based on the findings of this study, the following recommendations are proposed for the successful implementation of the student project management system:

1. Conduct Comprehensive user training: Ensure that students, supervisors, and administrators receive thorough training on the system's features and functionalities to maximize its effective utilization.
2. Emphasize data security and privacy: Implement robust security measures, such as encryption and access controls, to safeguard the confidentiality and integrity of student projects and personal information.
3. Foster a culture of collaboration: Encourage open communication and collaboration among students, supervisors, and administrators to continuously improve the system and address any emerging challenges or requirements.
4. Regularly maintain and update the system: Allocate resources for periodic software updates, database maintenance, and hardware upgrades to ensure the system remains efficient, secure, and compatible with evolving technologies.
5. Integrate with existing university systems: Explore the possibility of integrating the student project management system with other relevant university systems, such as student information systems or learning management platforms, to enhance data consistency and streamline processes. If integrated other faculty can be added to the platform for seamless and easy management of student's project.
6. Continuously monitor and evaluate system performance: Implement mechanisms to regularly monitor system performance, collect user feedback, and analyze usage data to identify areas for improvement and make informed decisions about future enhancements.

By implementing these recommendations, Nile University can maximize the benefits of the student project management system, fostering a more efficient and collaborative environment for research, project development, and academic excellence within the Faculty of Computing.

REFERENCES

- [1] A. V. Afanasev, N. N. Gorlushkina, and D. E. Shutov, "Development of Information Support for the Student Project Management System (In Russ.)," *Economics. Law. Innovaion*, no. 2, pp. 92–101, Jun. 2023, doi: 10.17586/2713-1874-2023-2-92-101.
- [2] L. Yahaya, "Final Year Students' Projects Allocation and Management System," *Arid Zone Journal of Basic and Applied Research*, vol. 2, no. 2, pp. 83–98, Apr. 2023, doi: 10.55639/607.131211.
- [3] J. Soyemi and F. O. Isinkaye, "A WEB-BASED FINAL YEAR STUDENT PROJECT DUPLICATION DETECTION SYSTEM."
- [4] E. C. Nwangwu and G. E. Asuquo, "Development of Computerized Students' Project Management System (CSPMS) for Enhancing Students' Research in Public Universities in Akwa Ibom State," *Journal of CUDIMAC (J-CUDIMAC)*, vol. 10, no. 1, 2022, [Online]. Available: <http://cudimac.unn.edu.ng/volume-10/http://cudimac.unn.edu.ng/volume-10/>
- [5] S. Malik, S. Abdulla Al-youbi, S. Bandar Al-youbi, R. Hussein, G. Dweik, and S. Arif, "Web-based Archive Management and Student Guidance for Final Year Projects," *Journal of e-Learning and Higher Education*, pp. 1–11, May 2018, doi: 10.5171/2018.871144.
- [6] V. Vijayakumar, "Protrack: A Student-Teacher Project Management Tool," vol. 12, no. 10, pp. 3173–3186, Apr. 2021, doi: 10.17762/TURCOMAT.V12I10.4972.
- [7] Imed Romdhani, Safa Habibullah, and Martin Tawse, "(PDF) Student project performance management system for effective final year and dissertation projects supervision." Accessed: Dec. 21, 2023. [Online]. Available: https://www.researchgate.net/publication/266210409_Student_project_performance_management_system_for_effective_final_year_and_dissertation_projects_supervision
- [8] U. U. Chikwendu and E. M. Amaechi, "International Journal of Current Science Research and Review Web-Based Student Project Management System: A Tetfund Institution-Based Research Report", doi: 10.47191/ijcsrr/V4-i12-05.
- [9] R. Bhatt, D. Bhorade, A. Kinare, and D. Patel, "Student Project Management System (SPMS)," *International Research Journal of Engineering and Technology*, vol. 9001, 2008, [Online]. Available: www.irjet.net
- [10] M. A. Bakar, N. Jailani, Z. Shukur, and N. F. Mohd Yatim, "Final year supervision management system as a tool for monitoring Computer Science projects," in *Procedia - Social and Behavioral Sciences*, 2011, pp. 273–281. doi: 10.1016/j.sbspro.2011.05.039.
- [11] International Communication Sciences Association., *2010 International Conference on Multimedia Communications: Mediacom 2010, 7-8 August 2010, Hong Kong, China, proceedings*. IEEE Computer Society, 2010.
- [12] E. C. Nwangwu and G. E. Asuquo, "Development of Computerized Students' Project Management System (CSPMS) for Enhancing Students' Research in Public Universities in Akwa Ibom State," *Journal of CUDIMAC (J-CUDIMAC)*, vol. 10, no. 1, 2022, [Online]. Available: <http://cudimac.unn.edu.ng/volume-10/http://cudimac.unn.edu.ng/volume-10/>

- [13] A. Adamu, "FINAL YEAR STUDENT PROJECT ALLOCATION ARCHIVING AND MANAGEMENT SYSTEM," 2020. [Online]. Available: www.ftstjournal.com
- [14] N. Soms, "Student Project Management System," 2021. [Online]. Available: www.pmi.org.
- [15] J. T. Fakoya, M. A. Ibiyomi, and A. Akeem, "STUDENTS' FINAL YEAR PROJECTS RECORD MANAGEMENT SYSTEM," 2021. [Online]. Available: www.repcomseet.com
- [16] L. Yahaya, "Final Year Students' Projects Allocation and Management System," *Arid Zone Journal of Basic and Applied Research*, vol. 2, no. 2, pp. 83–98, Apr. 2023, doi: 10.55639/607.131211.

APPENDIX I

```
// MySQL Connection
const connection = mysql.createConnection({
  host: 'localhost',
  user: 'root',
  password: 'nazeer3664',
  database: 'projectdb',

  authSwitchHandler: function ({ pluginName, pluginData }, cb) {
    if (pluginName === 'mysql_native_password') {
      const password = 'nazeer3664';
      const buffer = Buffer.from(password + '\0');
      cb(null, buffer);
    } else {
      cb(new Error('Unsupported authentication plugin: ' + pluginName));
    }
  }
});

connection.connect(err => {
  if (err) {
    console.error('Database connection failed: ' + err.stack);
    return;
  }
  console.log('Connected to database.');
```

```
});

app.get('/get_username', (req, res) => {
  const userType = req.session.userType;

  // Define the table name based on the user's role
  let tableName;
  switch (userType) {
    case 'student':
      tableName = 'student';
```

APPENDIX II

```
login.html > html > body > script > addEventListener('submit') callback
2  <html lang="en">
3  <head>
4
5  </head>
6  <body>
7
8  <title>Login</title>
9  </head>
10 <body>
11 <h1>Login</h1>
12 <form id="login-form">
13   <label for="username">Username:</label>
14   <input type="text" id="username" name="username" required><br><br>
15   <label for="password">Password:</label>
16   <input type="password" id="password" name="password" required><br><br>
17   <select id="user-type" name="user-type">
18     <option value="admin">Admin</option>
19     <option value="student">Student</option>
20     <option value="supervisor">Supervisor</option>
21   </select><br><br>
22   <button type="submit">Login</button>
23 </form>
24
25 <script>
26 document.getElementById('login-form').addEventListener('submit', function(event) {
27   event.preventDefault();
28   if (validateForm()) {
29     var formData = new FormData(this);
30     var userType = formData.get('user-type');
31     var userData = {
32       username: formData.get('username'),
33       password: formData.get('password')
34     };
35
36     var xhr = new XMLHttpRequest();
37     xhr.onreadystatechange = function() {
38       if (this.readyState == 4) {
39         if (this.status == 200) {
40           var response = JSON.parse(this.responseText);
41           alert(response.message);
42           // Redirect to appropriate page based on user type
43           window.location.href = response.redirectUrl;
44         } else {
```