A

Project Report

on

**PROJECT MANAGER**

Submitted in partial fulfillment of the requirement of

Project – VI (BIT379C0)

of

Bachelor of Information Technology (BIT)

**Submitted To**



Purbanchal University

Biratnagar, Nepal

**Submitted By**

Rupak Rawal (364365)

Samir Shrestha (363574)

Utsav Maharjan (363580)

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

September 8, 2023

A

Project Report

on

**PROJECT MANAGER**

Submitted in partial fulfillment of the requirement of

Project – VI (BIT379C0)

of

Bachelor of Information Technology (BIT)

**Submitted To**

Purbanchal University

Biratnagar, Nepal

**Submitted By**

Rupak Rawal (364365)

Samir Shrestha (363574)

Utsav Maharjan (363580)

**Project Supervisor**

Saroj Pandey

HOD, Department of IT

**KANTIPUR CITY COLLEGE**

Putalisadak, Kathmandu

September 8, 2023

# TOPIC APPROVAL SHEET

This is to certify that this project report entitled **“Project Manager”** made by Rupak Rawal **(**364365**)**, Samir Shrestha(363574)and Utsav Maharjan(363580**)** of Bachelor of Information Technology**,** 6th semester has been found suitable as per the credit given by Kantipur City College and Purbanchal University, Biratnagar Nepal.

**Mr. Saroj Pandey Mr. Rubim Shrestha**

**Project Supervisor Project Coordinator**

# CERTIFICATE FROM SUPERVISOR

This is to certify that the major project report entitled **“Project Manager”** undertaken and demonstrated by Rupak Rawal **(**364365**)**, Samir Shrestha(363574)and Utsav Maharjan(363580**)** has been successfully completed under my supervision as a partial fulfillment of the requirements for the degree of Bachelor of Information Technology**,** 6th semester under Purbanchal University, Biratnagar Nepal. I, henceforth, approve this project to be awarded the certificate by the concerned authority.

During supervision, I found students hardworking, skilled and ready to undertake any professional work related to this field in future.

**Mr. Saroj Pandey**

HOD, Department of IT

Project Supervisor

Date: 8th September. 2023

# ACKNOWLEDGEMENTS

The project members would like to express our sincere gratitude to our project supervisor **Mr. Saroj Pandey** for his continuous support, motivation and enthusiasm. We are deeply grateful to the project supervisor for supervising, motivating and being co-operative, we would like to thank KCC for providing opportunities that help us to know more about PHP.

We are immensely obliged to our friends for their deviating inspiration, encouraging guidance and kind supervision in the completion of our project.

**Group Members**

Rupak Rawal (364365)

Samir Shrestha (363574)

Utsav Maharjan (363580)

# ABSTRACT

Project manager is a web-based platform that is useful to students and project managers for the management, tracking and supervision of projects.

Project manager is the concept of making the project successful through knowledge, processes, methods and experience. This website acts as an intermediate between students and super admin. The main objective of a project manager is to achieve project goals within the estimated time with quality.

**TABLE OF CONTENT**

[TOPIC APPROVAL SHEET i](#_Toc145010576)

[CERTIFICATE FROM SUPERVISOR ii](#_Toc145010577)

[ACKNOWLEDGEMENTS iii](#_Toc145010578)

[ABSTRACT iv](#_Toc145010579)

[LIST OF TABLES v](#_Toc145010580)

[LIST OF FIGURES vi](#_Toc145010581)

[Chapter 1: INTRODUCTION 1](#_Toc145010582)

[1.1 Overview 1](#_Toc145010583)

[1.2 Problem Statement 2](#_Toc145010584)

[1.3 Objectives 2](#_Toc145010585)

[1.4 Features 2](#_Toc145010586)

[1.5 Significances/Importance 2](#_Toc145010587)

[1.6 Scope and limitation 2](#_Toc145010588)

[1.7 Organization of document 3](#_Toc145010589)

[Chapter 2: LITERATURE REVIEW 4](#_Toc145010590)

[2.1 Redmine 4](#_Toc145010591)

[2.2 ProjectPier 5](#_Toc145010592)

[2.3 Collabtive 5](#_Toc145010593)

[2.4 DotProject 6](#_Toc145010594)

[2.5 Project HQ 7](#_Toc145010595)

[Chapter 3: METHODOLOGY 8](#_Toc145010596)

[3.1 Software Development Life Cycle 8](#_Toc145010597)

[3.1.1 Requirement Gathering and Analysis 8](#_Toc145010598)

[3.1.2 Quick Design 9](#_Toc145010599)

[3.1.3 Build a Prototype 9](#_Toc145010600)

[3.1.4 Initial User Evaluation 9](#_Toc145010601)

[3.1.5 Refining Prototype 9](#_Toc145010602)

[3.1.6 Implement Product and Maintain 9](#_Toc145010603)

[3.2 Technologies and Tools used 10](#_Toc145010604)

[3.2.1 PHP 10](#_Toc145010605)

[3.2.2 PHPMYADMIN 10](#_Toc145010606)

[3.2.3 VS CODE 11](#_Toc145010607)

[3.2.4 XDebug 11](#_Toc145010608)

[3.3 Assignment of roles and responsibilities 11](#_Toc145010609)

[Chapter 4: SYSTEM ANALYSIS 13](#_Toc145010610)

[4.1 Requirement Specification 13](#_Toc145010611)

[4.1.1 Functional Requirement 13](#_Toc145010612)

[4.1.2 Nonfunctional Requirement 14](#_Toc145010617)

[4.2 Feasibility study 14](#_Toc145010618)

[4.2.1 Technical Feasibility 14](#_Toc145010619)

[4.2.2 Schedule Feasibility 14](#_Toc145010620)

[4.2.3.1 Gantt chart 15](#_Toc145010621)

[Chapter 5: SYSTEM DESIGN 16](#_Toc145010622)

[5.1 System Architecture 16](#_Toc145010623)

[5.2 Object Oriented 16](#_Toc145010624)

[5.2.1 Class Diagram 16](#_Toc145010625)

[5.2.2 Activity Diagram 17](#_Toc145010626)

[5.2.2 Use Case Diagram 18](#_Toc145010627)

[5.2.3 Sequence Diagram 19](#_Toc145010628)

[5.3 Database Design 20](#_Toc145010629)

[5.3.1 ER Diagram 20](#_Toc145010630)

[5.3.2 Relational Data Structure 21](#_Toc145010631)

[5.3.3 Data Dictionary 22](#_Toc145010632)

[Chapter 6: TESTING AND DEBUGGING 27](#_Toc145010633)

[6.1 Tools Used in Testing 27](#_Toc145010634)

[6.2 Testing and Debugging 28](#_Toc145010635)

[Chapter 7: CONCLUSION 29](#_Toc145010636)

[7.1 Conclusion 29](#_Toc145010637)

[7.2 Limitations 29](#_Toc145010638)

[7.3 Future Enhancement 29](#_Toc145010639)

[Chapter 8: REFERENCES 30](#_Toc145010640)

[APPENDIX 31](#_Toc145010641)

[Appendix (Output Screenshots) 31](#_Toc145010642)

# LIST OF TABLES

|  |  |  |
| --- | --- | --- |
| Table Number | Table | Page No. |
| 3.3 | Assignment of Roles and Responsibilities | 11-12 |
| 4.3.2.1 | Gantt Chart | 15 |
| 6.1 | Tools Used in Testing | 27 |
| 6.2 | Testing and Debugging | 28 |

# LIST OF FIGURES

|  |  |  |
| --- | --- | --- |
| Figure Number | Figure | Page No. |
| 3.1 | Prototype Model | 8 |
| 5.2.1 | Class Diagram | 16 |
| 5.2.2 | Activity Diagram | 17 |
| 5.2.3 | Use Case Diagram | 18 |
| 5.2.4 | Sequence Diagram | 19 |
| 5.3.1 | ER Diagram | 20 |
| 5.3.2 | Relational Data Structure | 21 |
| 5.3.3 | Data Dictionary | 22-26 |

# Chapter 1: INTRODUCTION

We looked through a variety of websites and discovered the project management website to be quite intriguing. So, we made the decision to create the project manager website, which aids in team organization, monitoring, and project execution. Students can communicate with the super admin using this website.

For the management, tracking, and supervision of projects, project managers and students can both benefit from the web-based platform known as "Project Manager." The idea of project management is to make a project successful through information, procedures, techniques, and experience. This website serves as a liaison between students and the top management. Planning, monitoring, and reporting on projects can be done in an organized manner with the help of a project manager. The primary goal of a project manager is to complete the project on schedule and to specification.

## 1.1 Overview

A project manager serves as a good example for controlling how a project is carried out. It can offer a structure for controlling expectations, assigning duties, and establishing procedures. A project manager may make use of standard business software programs or specialized project management software. A project manager is a collection of approaches and tools that help you plan, organize, and schedule all that is necessary for a project to be successful.

A project plan is a collection of official documents outlining the project's execution and control phases. In addition to addressing scope, cost, and schedule baselines, the plan takes risk management, resource management, and communications into account.

## 1.2 Problem Statement

We have discovered that managing a project often entails a number of responsibilities, such as planning, team cooperation, scheduling, etc. By researching various systems or applications about classical time. These duties can be time-consuming, challenging to coordinate, and labor-intensive. Hence, the need for effective time management develops. They might not be safe. Papers are prone to damage by water, fire, and other natural disasters by their sheer nature. This technology makes it possible to manage time through digital processes that take less time. When compared to traditional time, this is more secure.

## 1.3 Objectives

* To track activities of projects.
* To develop efficient communication and productive guidelines.

## 1.4 Features

* User can edit their profiles.
* User can manage events and tasks.
* User can monitor the project progress.

## 1.5 Significances/Importance

* It helps in efficient monitoring.
* It helps in efficient evaluation.

## 1.6 Scope and limitation

**Scope:** Project Manager can be used in most of the project management institutes.

**Limitation:** The website is not responsive.

## 1.7 Organization of document

Chapter 1 includes introduction, overview, problem statement, objectives, features, significance, scope and limitation and organization of the document. Chapter 2 includes the details of literature review. Chapter 3 includes methodology like software development life cycle, technologies and tools used and assignment of roles and responsibilities.

Chapter 4 includes System Analysis like Requirement Specification, Feasibility Study. Chapter 5 includes System Design like Context Level diagram, level 1 DFD, Use Case Diagram, ER Diagram, Data Dictionary and Relational Data Structure. Chapter 6 includes the details of System Development and Implementation i.e. programming platform and operating environment. Chapter 7 includes Testing and Debugging. Chapter 8 and 9 includes Conclusion and Reference.

# Chapter 2: LITERATURE REVIEW

Project management systems are software applications that help manage tasks, schedules, budgets, and resources associated with a project. In this literature review, we will explore some of the popular project management systems that use PHP as the primary programming language.

## 2.1 Redmine

Redmine is an open-source project management tool written in Ruby on Rails but can run on PHP. It provides features such as issue tracking, Gantt chart, calendar, wiki, and time tracking. It also supports multiple databases like MySQL, PostgreSQL, and SQLite.

**Pro****s**

* Redmine is very adaptable and may be adjusted to meet the unique requirements of your team or business.
* Redmine can be installed on various operating systems, including Windows, Linux, and macOS.
* Redmine can be integrated with many other tools, such as Git, Jenkins, and Slack.

**Cons**

* Redmine can be complex to set up and configure and it may take some time to learn how to use it effectively.
* Redmine has a significant user base, but the official support is few, so you might have to turn to outside sources for assistance.

## 2.2 ProjectPier

ProjectPier is a free, open-source, self-hosted project management software written in PHP. It offers features like task management, file sharing, wiki, and time tracking. It supports MySQL and PostgreSQL databases and has a simple user interface.

**Pros**

* ProjectPier has a simple and intuitive user interface.
* ProjectPier allows teams to collaborate and communicate effectively.

**Cons**

* ProjectPier is easy to use but still requires some technical knowledge to set up and configure properly.
* ProjectPier has limited customization options, and users cannot customize the software to meet their specific needs easily.

## 2.3 Collabtive

Collabtive is a free, open-source, web-based project management software written in PHP. It has features like task management, time tracking, file sharing, and calendar. It also supports multiple languages and is compatible with MySQL and PostgreSQL databases.

**Pros**

* Collabtive has a user-friendly interface, making it easy to use for all team members.
* Collabtive is highly customizable, allowing users to configure it to meet their specific needs.
* Collabtive is cloud-based, which means that it can be accessed from anywhere with an internet connection.

**Cons**

* Collabtive has limited integrations with other tools, which may be a drawback for teams that rely on other software for their work.
* Collabtive has limited mobile apps, which may be a challenge for the teams that need to manage their product on-the-go.

## 2.4 DotProject

DotProject is an open-source, web-based project management software written in PHP. It provides features like task management, file sharing, Gantt chart, and calendar. It supports multiple languages and databases like MySQL and PostgreSQL.

**Pros**

* DotProject is highly customizable, allowing users to create custom fields, forms, and reports according to their needs.
* DotProject is available in multiple languages.
* DotProject has multiple platform support.

**Cons**

* DotProject has an outdated user interface, which can make it difficult for users who are accustomed to modern UI designs.
* Setting up DotProject can be complicated especially for users who do not have technical knowledge.
* DotProject may be susceptible to security vulnerabilities that can compromise the data of its users.

## 2.5 Project HQ

Project HQ is a free, open-source, web-based project management software written in PHP. It provides features like task management, document management, and time tracking. It also supports multiple databases like MySQL, PostgreSQL, and SQLite.

**Pros**

* ProjectHQ is free to use and offers a high level of customization.
* ProjectHQ provides reporting tools.
* ProjectHQ offers collaboration.

**Cons**

* ProjectHQ doesn’t offer as much customization.
* ProjectHQ doesn’t offer as much integration.

In conclusion, the above-listed project management systems are just a few examples of the many project management tools available that use PHP. Each system has its unique features and functionalities, making them suitable for different types of projects and organizations. It is important to evaluate and select a system that meets your project requirements and aligns with your organization's goals.

# Chapter 3: METHODOLOGY

## 3.1 Software Development Life Cycle

We used a prototype model to develop our system. Prototype model is a software development methodology that involves creating a working model of the software application before building the full system. The prototype models to understand project feasibility and reduce cost.

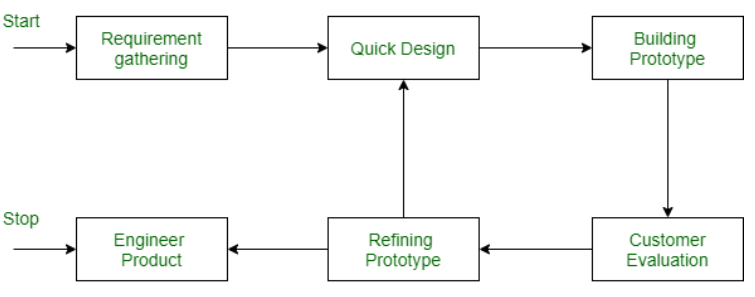


Fig 3.1: Prototype Model

### 3.1.1 Requirement Gathering and Analysis

A prototyping model starts with requirement gathering and analysis. In this phase we first gathered the various information about system in detail that are essential for our system. During this process users of the system are interviewed to know what is their expectation from the system.

### 3.1.2 Quick Design

In this stage a simple design of the web application is created after gathering the required requirements. It is not a complete design of the system but gives a brief idea of the system to the user.

### 3.1.3 Build a Prototype

In this stage an actual prototype of our web application is designed based on the information gathering from second phase quick design.

### 3.1.4 Initial User Evaluation

In this stage the system is presented to the user for an initial evaluation. It helps to find out the strength and weakness of the system and suggestions are collected from the user and provided to the developer.

### 3.1.5 Refining Prototype

If the users are not happy with the system, then it refines the system according to the user’s feedback and suggestion. This stage will not be over until all the requirements specified by the user are met. Once the user is satisfied with the software then the final system is developed.

### 3.1.6 Implement Product and Maintain

In this stage the final system is developed based on the final prototype. The system undergoes routine maintenance for minimizing downtime and preventing large scale failures.

## 3.2 Technologies and Tools used

### 3.2.1 PHP

Popular server-side scripting language for web development is PHP (Hypertext Preprocessor). Its main uses are web apps and dynamic web pages. PHP code runs on the server to produce client-side web technologies including HTML, CSS, and JavaScript, which are then transmitted to the user's browser to be displayed. In addition to handling form data, creating and manipulating files, and performing a number of other operations frequently required in web development, PHP can connect with databases.

Since PHP is an open-source language, a sizable developer community contributes to its growth and upkeep. Most web hosting services accept it, and it is frequently used in conjunction with well-known web frameworks like Laravel, and CodeIgniter.

### 3.2.2 PHPMYADMIN

A free and open-source web application called PHPMyAdmin is used to manage and oversee MySQL databases. Users can import and export data as well as run SQL queries in addition to creating, editing, and deleting databases, tables, and fields.

Web developers and administrators frequently use PHPMyAdmin as a tool to manage MySQL databases for websites and web applications. It offers a graphical user interface (GUI) that may be easier to use and more accessible than a command-line interface for managing databases.

### 3.2.3 VS CODE

Microsoft created the free and open-source code editor known as VS Code (Visual Studio Code). It supports a large number of programming languages and frameworks and is available for Windows, macOS, and Linux.

### 3.2.4 XDebug

Popular open-source PHP extension Xdebug gives PHP programs additional debugging and profiling features. By providing thorough details on the execution flow and variable values, it enables developers to hunt out defects and performance issues in their programs.

## 3.3 Assignment of roles and responsibilities

|  |  |  |
| --- | --- | --- |
| Member Name | Roles |  |
| Samir Shrestha  (363574) | * Designing * Documentation * Frontend * Backend | * DFD * ER Diagram * UI/ UX * Features * System architecture * Software system development |
| Rupak Rawal  (364365) | * Designing * Documentation * Frontend | * Use case Diagram * Literature review * Software system development * Feasibility Study * UI/ UX * Log in and sign-up page (Frontend) |
| Utsav Maharjan  (363580) | * Designing * Documentation * Frontend | * UI/ UX * Feasibility Study * Problem Statement * Objectives * Feasibility Study |

Table 3.3: Assignment of roles and responsibilities

## Chapter 4: SYSTEM ANALYSIS

## 4.1 Requirement Specification

After the selection of the development process of the system. The first thing we did was to specify the requirement, which has been divided into two parts according to the requirement of the system.

### 4.1.1 Functional Requirement

#### Super Admin Requirements:

* The admin can approve/reject projects.
* The admin can monitor all the project progress.
* The admin can view, edit and delete user details.
* The admin can view and take actions on the report.

#### Teacher Requirements:

* The teacher can assign tasks to the projects.
* The teacher can monitor the progress of projects assigned to him/her.
* The teacher can assign/accept/reject the meetings.

#### Student Requirements:

* The student can submit a abstract.
* The students can request a meeting schedule.
* The student can view their project progress.

#### System Requirements:

* Project Manager offers logout functionality to end user’s sessions.
* Project Manager will only accept valid login details to access their respective projects.
* Project Manager will provide a password recovery facility.

### 4.1.2 Nonfunctional Requirement

* **Performance**: By using Ajax, we’ve received the efficiency of the website more fluent.
* **Reliability**: The system should maintain accuracy as it updates the data from the database.
* **Compatibility**: The website should be compatible with a range of devices and browsers, ensuring that all users can access the site regardless of their preferred platform.

## 4.2 Feasibility study

### 4.2.1 Technical Feasibility

During the study of this process, we studied the requirements of the technical equipment for the development of the system and found out all the equipment (php, mysql, javascript, xampp sever) is free to use which is convenient for us to develop our project.

### 4.2.2 Schedule Feasibility

During the study we studied about the time required to complete the development of the system and found out that we can complete the basic requirements of project management.

## 4.2.3.1 Gantt chart

|  |  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | June 2023 | | | | July 2023 | | | | August 2023 | | | | September 2023 |
|  | Week | | | | | | | | | | | | |
|  | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 | 2 | 3 | 4 | 1 |
| Requirements Analysis |  |  |  |  |  |  |  |  |  |  |  |  |  |
| System Design |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Database Design |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Implementation and coding |  |  |  |  |  |  |  |  |  |  |  |  |  |
| Documentation |  |  |  |  |  |  |  |  |  |  |  |  |  |

Table 4.2.3.1: Gantt Chart

# Chapter 5: SYSTEM DESIGN

## 5.1 System Architecture

A system that hosts, provides, and manages the majority of the resources and services that the client requests is known as a client-server architecture. This approach, also known as the networking computing model or client server network, involves the delivery of all requests and services across a network.

The client first transmits their request using a network-capable device. The network server then acknowledges and handles the user request. The server then sends the response to the client.

## 5.2 Object Oriented

### 5.2.1 Class Diagram

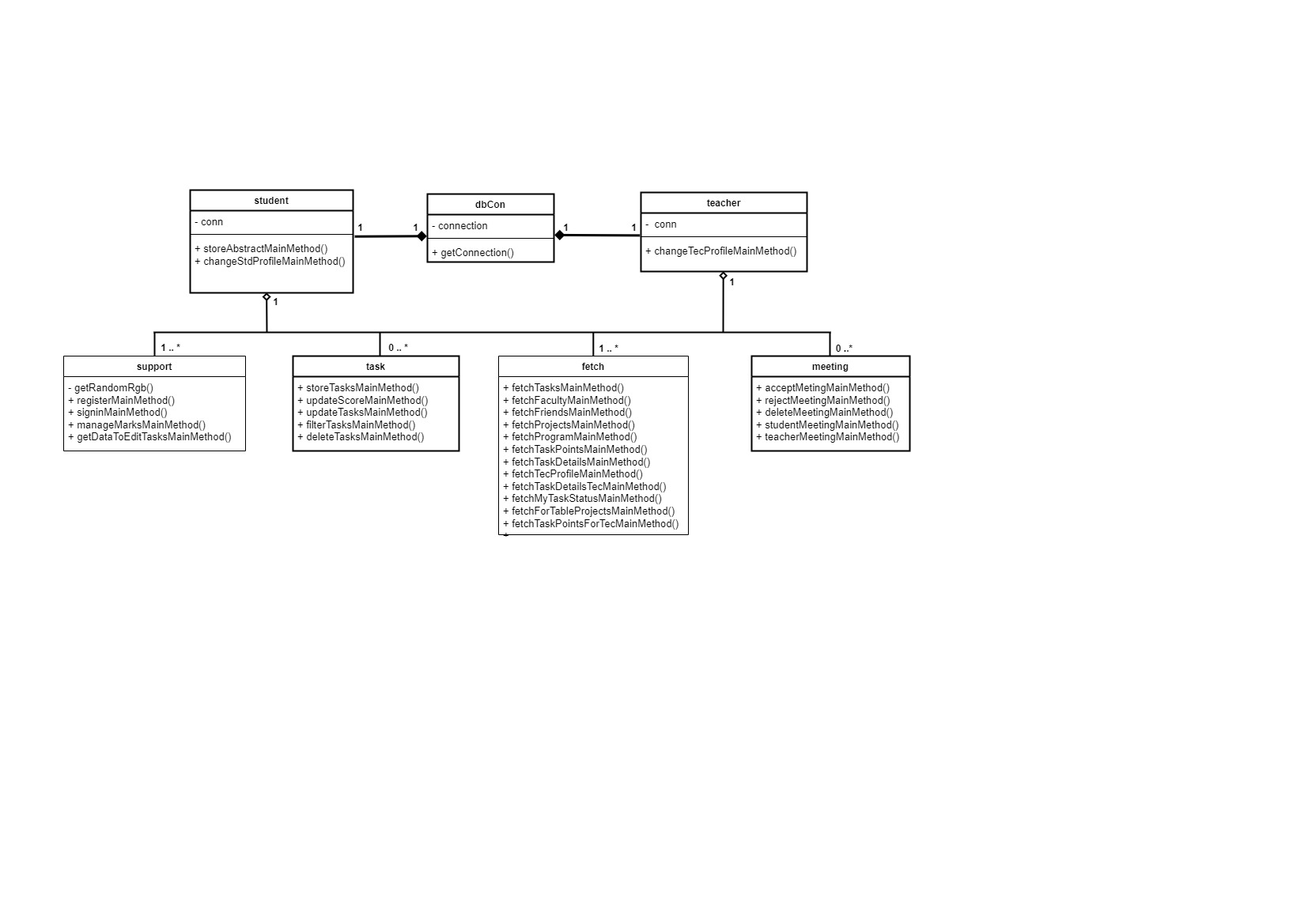
****

Fig 5.2.1: Class Diagram

### 5.2.2 Activity Diagram

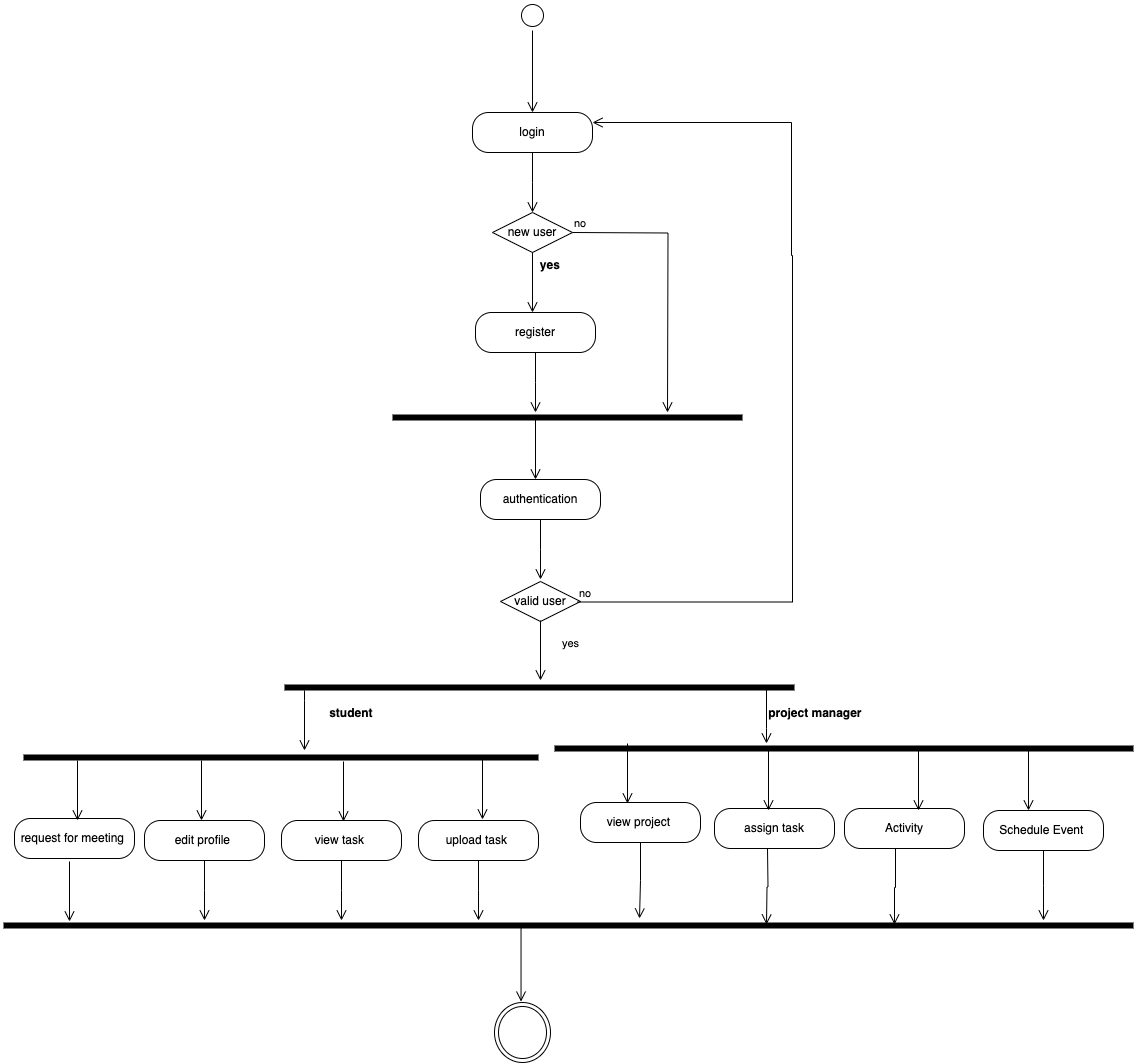


Fig 5.2.2: Activity Diagram

### 5.2.2 Use Case Diagram

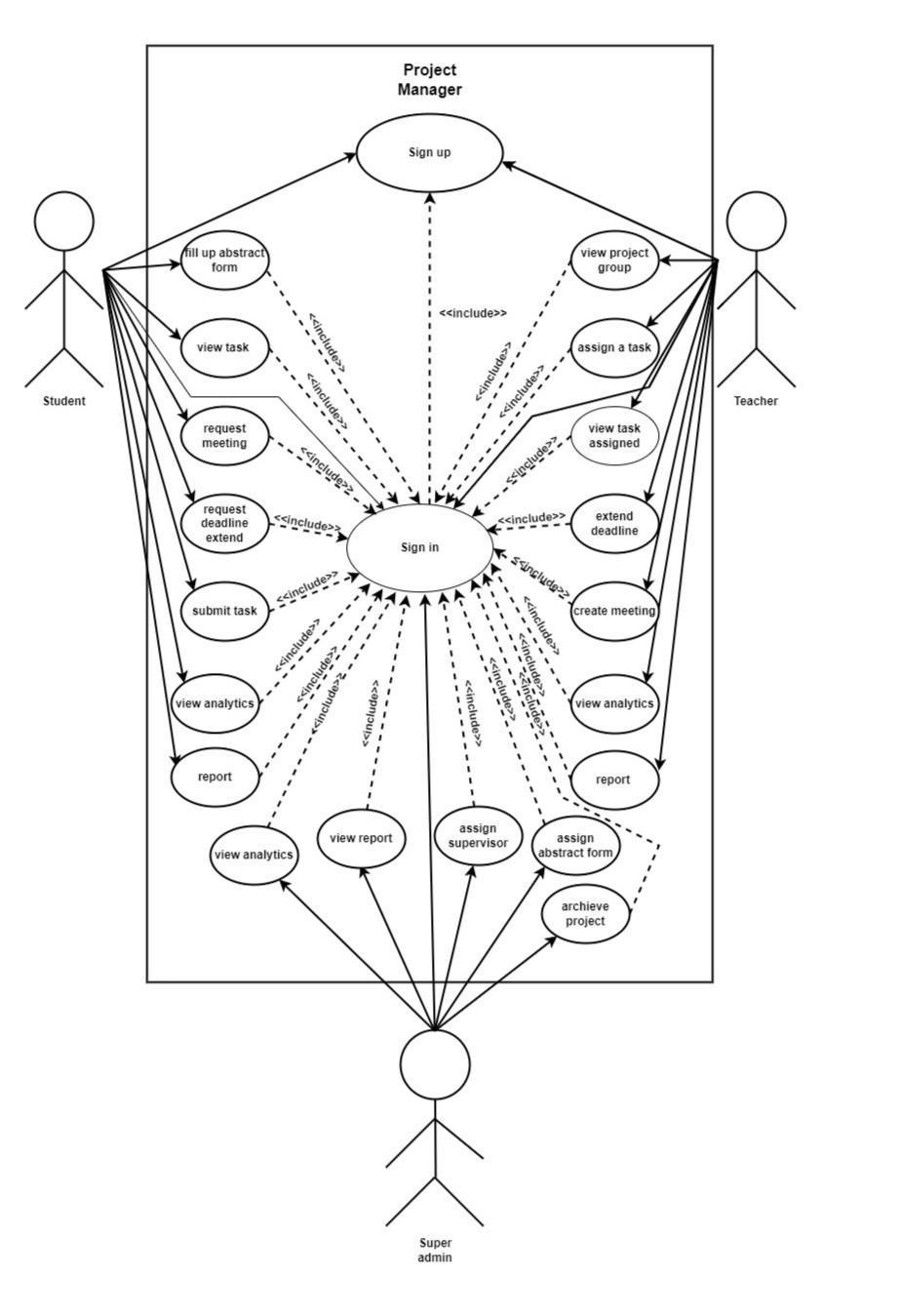
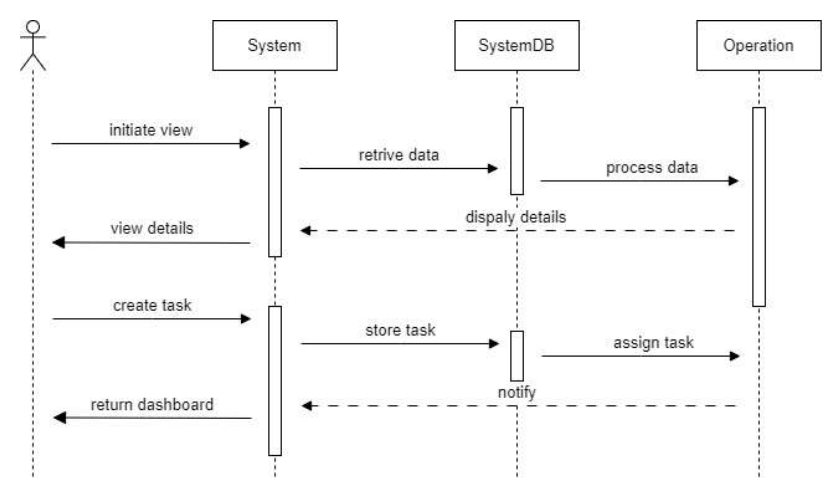
****

Fig 5.2.3: Use Case Diagram

### 5.2.3 Sequence Diagram

****

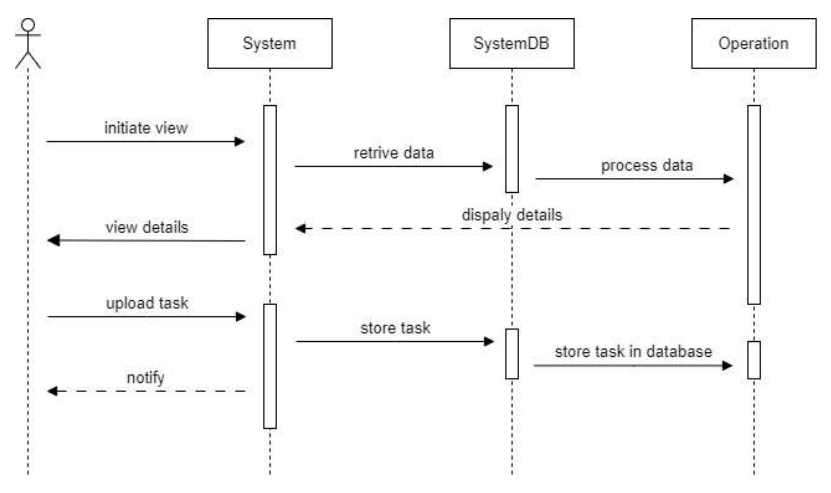
****

Fig 5.2.4: Sequence Diagram

## 5.3 Database Design

### 5.3.1 ER Diagram

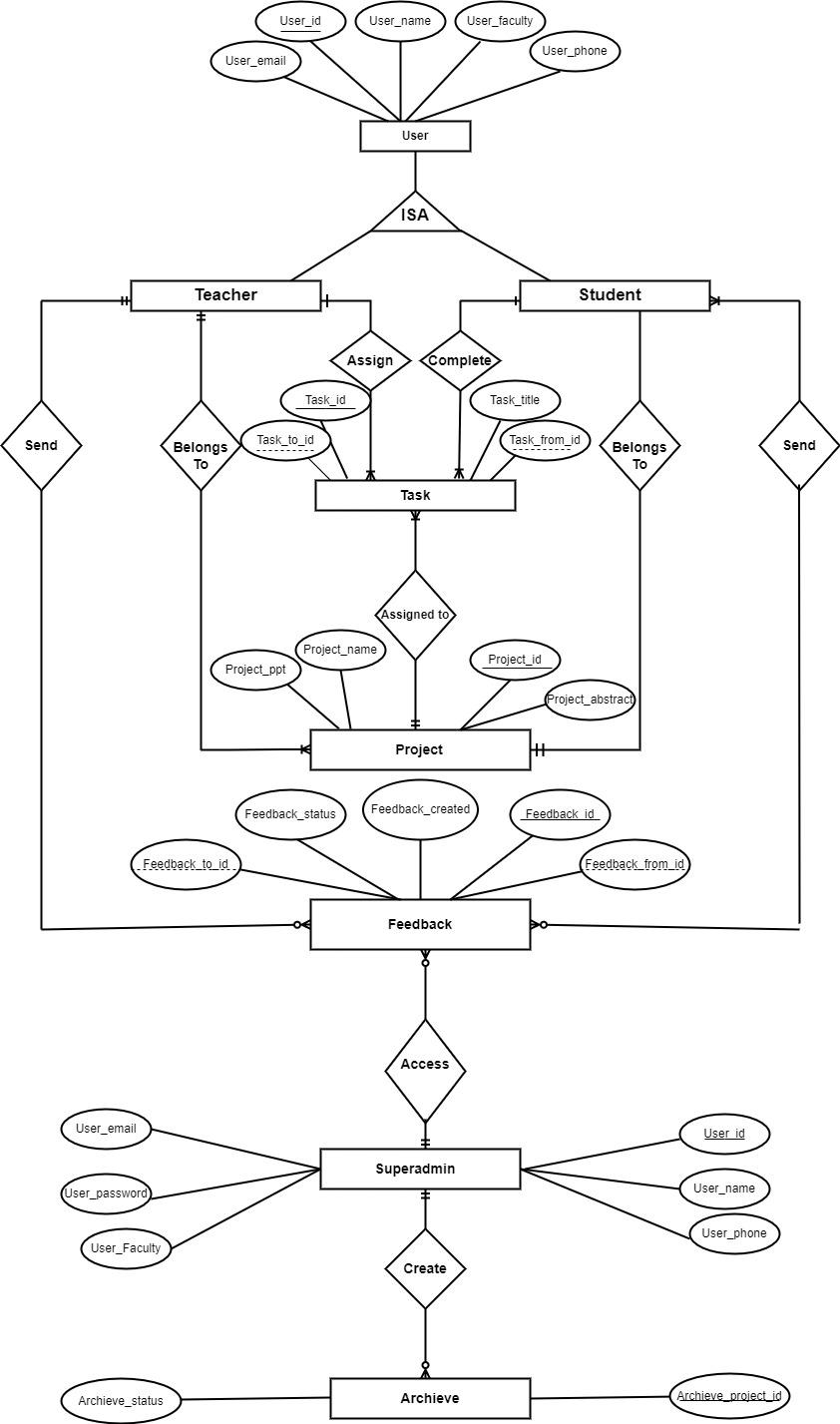
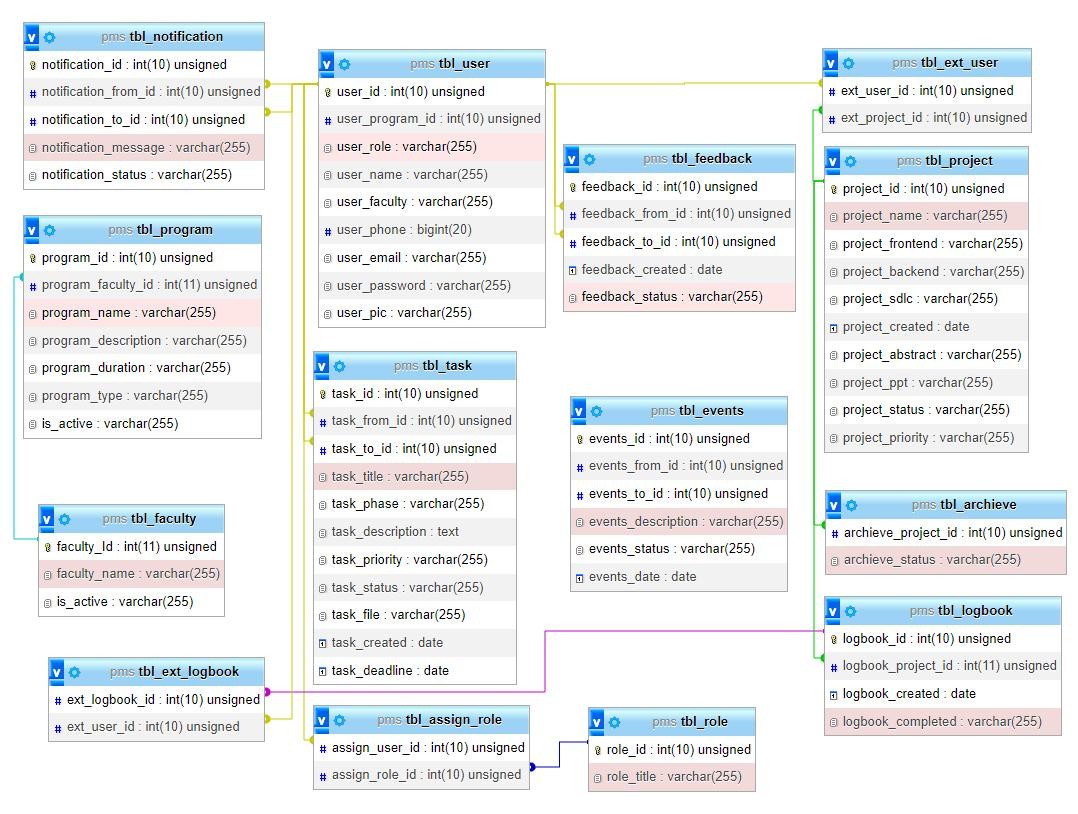
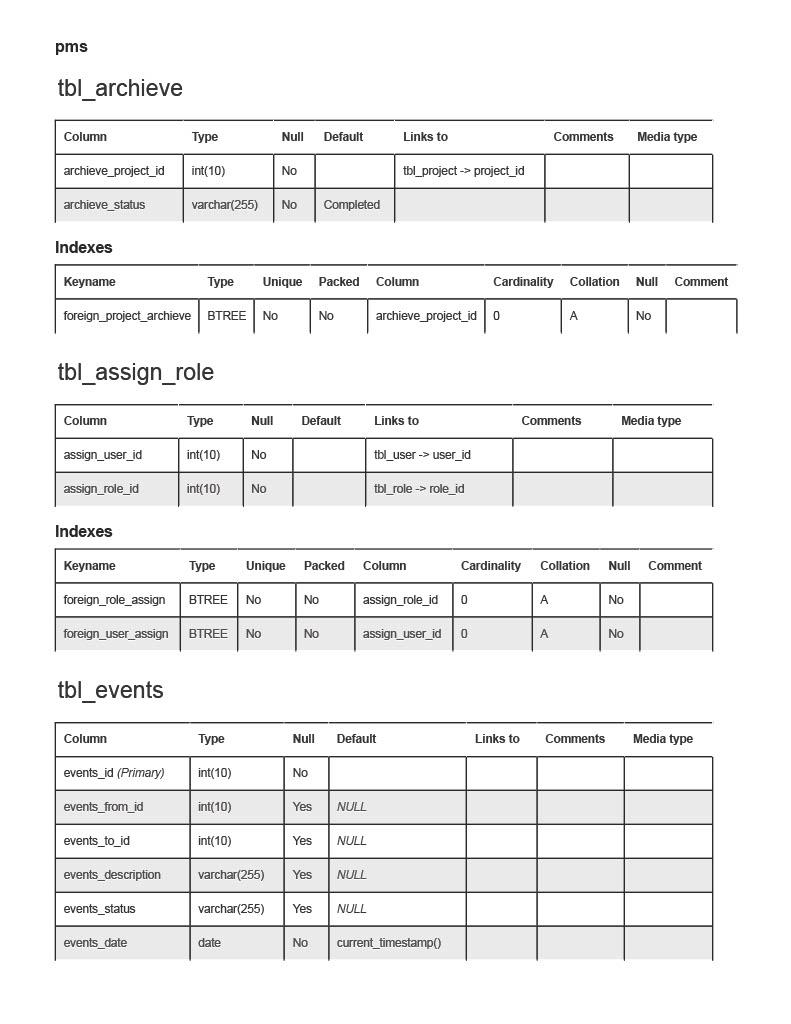
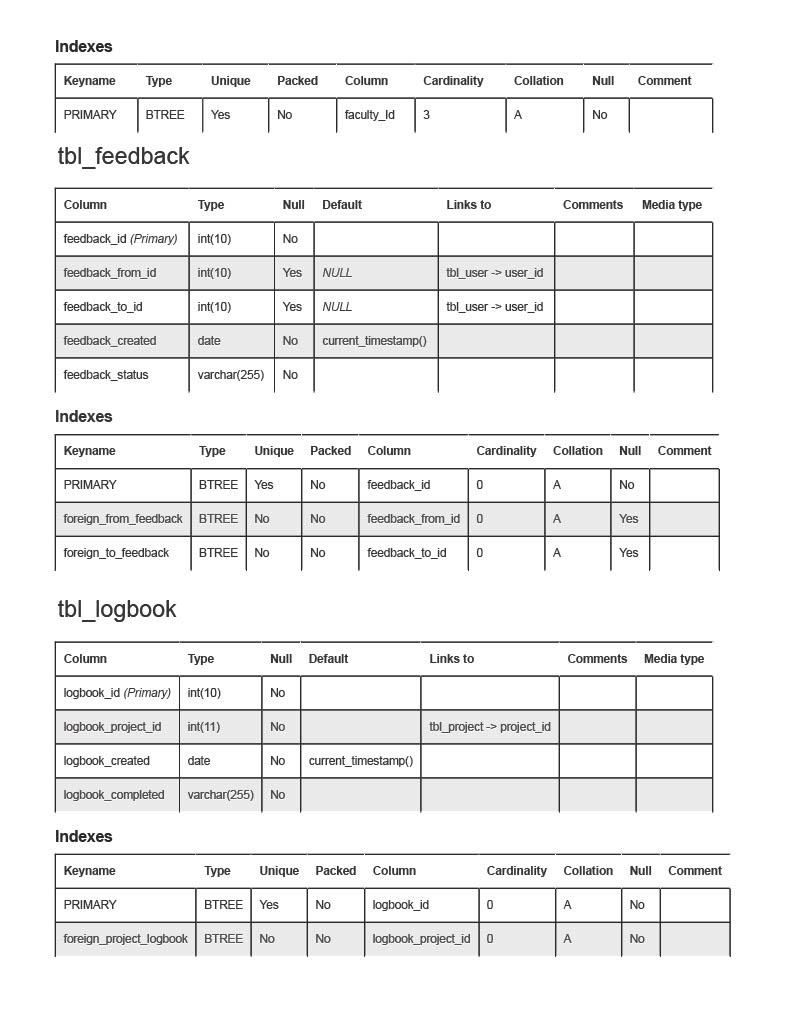
****

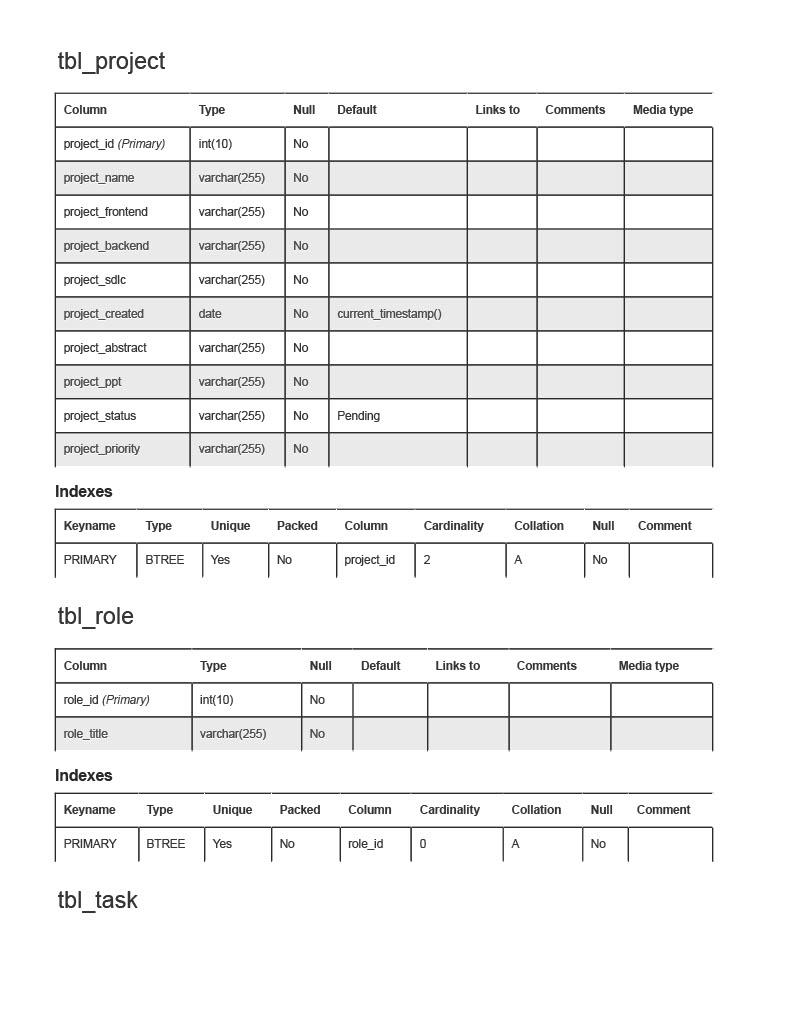
Fig 5.3.1: ER Diagram

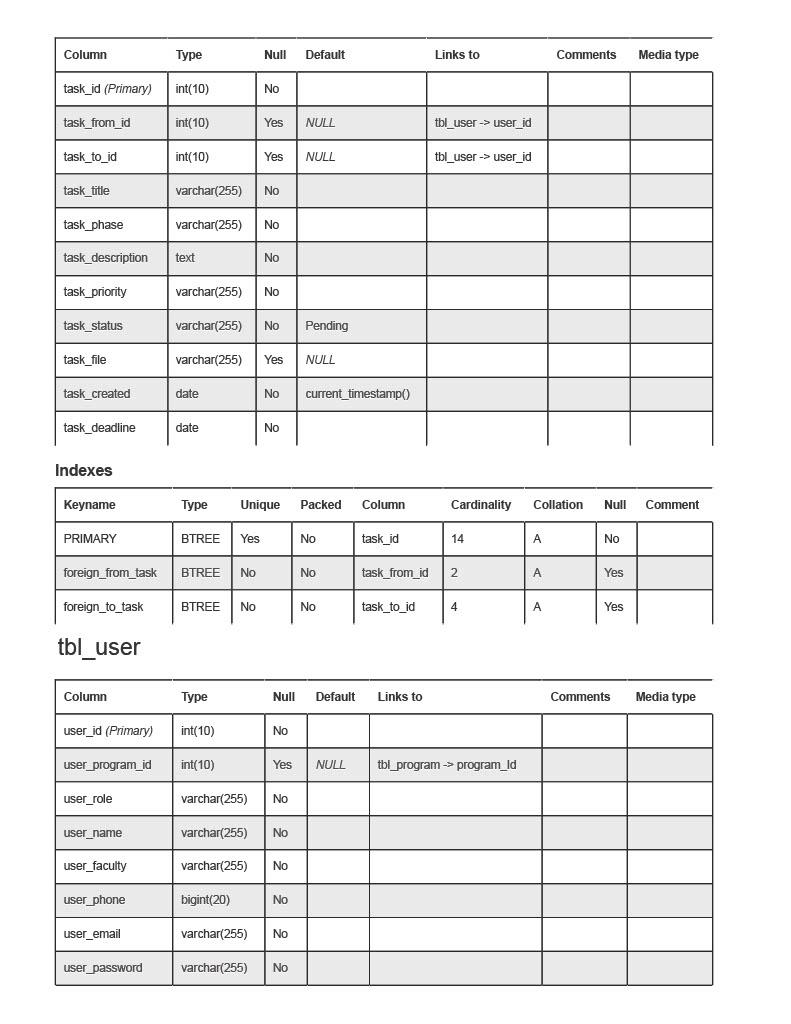
### 5.3.2 Relational Data Structure

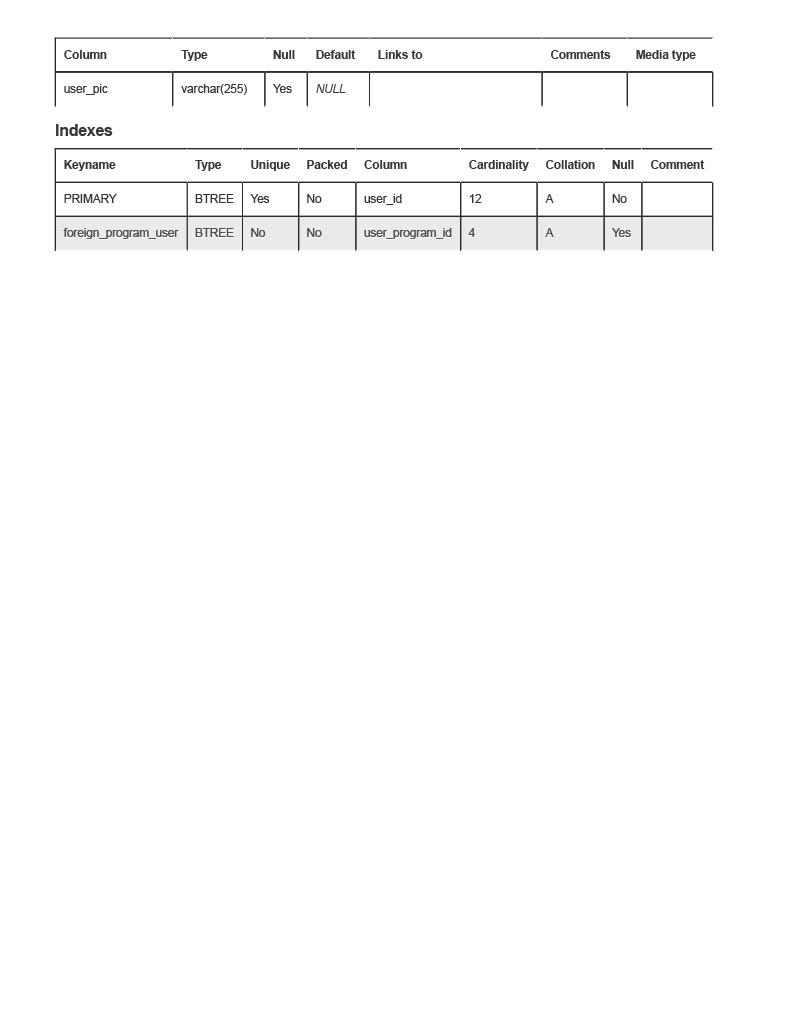


5.3.3 Data Dictionary









Chapter 6: TESTING AND DEBUGGING

## 6.1 Tools Used in Testing

|  |  |  |
| --- | --- | --- |
| **S.NO** | **Tool** | **Specification** |
| 1 | Hardware (Computer/ Laptop/Mobile) | Core: i5 or more  Generation: 8th or more |
| 2 | Vs Code | Version: 1.75.1 |
| 3 | Postman | Version: 10.0 |
| 4 | Xdebug | Version: 3.2.0 |

Table 6.1: Tools Used in Testing

## 6.2 Testing and Debugging

|  |  |  |  |
| --- | --- | --- | --- |
| **S.NO** | **Test** | **Expected Result** | **Actual Result** |
| 1 | Check whether teacher is registered or not | Teacher should be registered | Teacher is registered |
| 2 | Check whether student is registered or not | Student should be registered | Student is registered |
| 3 | Check whether user session is created after login or not | User session should be created | User session not created |
| 4 | Check whether user session is created after login or not | User session should be created | User session created |
| 5 | Check whether user is redirect  to respective dashboard | User should be redirected to their respective dashboard | No Response |
| 6 | Check whether user is redirect  to respective dashboard | User should be redirected to their respective dashboard | User is redirected to their respective dashboard |
| 7 | Check whether the user can request/assign meeting | User should be able to request/assign meeting | User was able to request/assign meeting |
| 8 | Check whether the interaction with database is efficient | User should interact with database effectively | Error Data Retrived |
| 9 | Check whether the interaction with database is efficient | User should interact with database effectively | User can interact with database effectively |
| 10 | Check whether user can manage tasks | User should be able to manage tasks | User can manage tasks |
| 11 | Check whether user can edit their profile | User should able to edit their profile | User can edit their profile |

Table 6.2: Testing and Debugging

# Chapter 7: CONCLUSION

## 7.1 Conclusion

For organizing projects, tasks, and resources, Project manager can be a very useful tool. They make it possible for teams to work together more productively, monitor deadlines, and track progress.

Project manager solutions can assist teams in staying organized and guaranteeing projects are finished on time and within budget using features like Gantt charts, work assignment, time tracking, and resource management.

## 7.2 Limitations

As we know that no any system can be 100% reliable and efficient. So, there are some drawbacks from our system which are as follows:

* Admin is not integrated
* Supervisor cannot respond to the submitted task

## 7.3 Future Enhancement

* Admin will be included
* Notification system will be implemented

# Chapter 8: REFERENCES

1. Aaron A. Izang. (2016), “A web- based system project management system”, International Journal of Advanced Research in Computer Science and Software Engineering, Vol. 6, pp. 39-45.
2. Ralf Gutkowski, (2016-10-09), Redmine, https://www.redmine.org/
3. Project-Pier. (2007-10-24), “A free open-source collaboration tool developed using PHP as a programming language”, http://www.projectpier.org/

# APPENDIX

## Appendix (Output Screenshots)

