

ON TRACKIFY:A PROJECT MONITORING WEBSITE

A MINIPROJECT REPORT

submitted by

ANDRIC ANTONY (PKD21IT013)
KRISHNAPRIYA P H (PKD21IT039)
NIYA PRAKASH (PKD21IT047)
RASEENA P A (PKD21IT049)

to

the APJ Abdul Kalam Technological University
in partial fulfilment of the requirements for the award of the Degree

of

Bachelor of Technology

in

Information Technology



Department of Information Technology
Government Engineering College Palakkad
Sreekrishnapuram, Palakkad-678633

May 2024

ON TRACKIFY:A PROJECT MONITORING WEBSITE

A MINIPROJECT REPORT

submitted by

ANDRIC ANTONY (PKD21IT013)
KRISHNAPRIYA P H (PKD21IT039)
NIYA PRAKASH (PKD21IT047)
RASEENA P A (PKD21IT049)

to

the APJ Abdul Kalam Technological University
in partial fulfilment of the requirements for the award of the Degree

of

Bachelor of Technology

in

Information Technology



Department of Information Technology
Government Engineering College Palakkad
Sreekrishnapuram, Palakkad-678633

May 2024

VISION AND MISSION OF THE COLLEGE

VISION

Excellence through the wings of science and technology.

MISSION

To transform youth to talented engineers with creativity and integrity who can meet the technological challenges for the service of society.

VISION AND MISSION OF THE DEPARTMENT

VISION

To achieve excellent standards in IT education and research by keeping abreast of innovations in Information Technology.

MISSION

- To nurture and develop students as competent IT professionals capable of undertaking challenging and innovative programmes.
- To foster self-disciplined and socially committed entrepreneurs.

PROGRAM EDUCATIONAL OBJECTIVES (PEOs)

- PEO1:** The graduates of the program will have strong foundation in mathematics, science and basic engineering concepts to solve engineering problems.
- PEO2:** They will possess in-depth knowledge in core subjects, enabling them to provide efficient, workable solutions for problems in various areas of IT.
- PEO3:** The graduates will have good interpersonal, leadership and communication skills, which will equip them to perform well in any environment.
- PEO4:** They will excel in IT professional careers and/or higher studies applying their technical knowledge and creative skills..
- PEO5:** They will be technically and ethically strong to relate engineering issues to the society, global economy and to emerging technologies.

PROGRAM SPECIFIC OUTCOMES (PSOs)

- PSO1:** Design and develop software and hardware systems in computing, IT, ITES and embedded systems.
- PSO2:** Apply mathematics, science, management and engineering concepts to solve emerging real world problems using suitable data structures and algorithms.

DECLARATION

We hereby declare that the project report entitled ” **ON TRACKIFY:A PROJECT MONITORING WEBSITE**” submitted by us to the APJ Abdul Kalam Technological University during the academic year 2023-24 in partial fulfilment of the requirements for the award of Degree of Bachelor of Technology in Information Technology is a record of bonafide project work carried out by us under the guidance and supervision of Dr.Rani M R.We further declare that the work reported in this project has not been submitted and will not be submitted, either in part or in full, for the award of any other degree or diploma in this institute or any other University.

ANDRIC ANTONY (PKD21IT013)

KRISHNAPRIYA P H (PKD21IT039)

NIYA PRAKASH (PKD21IT047)

RASEENA P A (PKD21IT049)

Place: Sreekrishnapuram

Date:19/06/2024

**DEPARTMENT OF INFORMATION TECHNOLOGY
GOVERNMENT ENGINEERING COLLEGE PALAKKAD
SREEKRISHNAPURAM, PALAKKAD – 678633**



CERTIFICATE

This is to certify that the report entitled “ **ON TRACKIFY:A PROJECT MONITORING WEBSITE**” submitted by **ANDRIC ANTONY (PKD21IT013)**, **KRISHNAPRIYA P H (PKD21IT039)**, **NIYA PRAKASH (PKD21IT047)**, and **RASEENA P A (PKD21IT049)**, to the APJ Abdul Kalam Technological University in partial fulfilment of the requirements for the award of the Degree of Bachelor of Technology in Information Technology is a bonafide record of the project work carried out by them under our guidance and supervision. This report in any form has not been submitted to any other University or Institute for any purpose.

GUIDE

Dr.Rani M.R

Assistant Professor

Dept. of Information Technology

HEAD OF THE DEPARTMENT

Dr.Sangeetha U

Associate Professor

Dept. of Information Technology

CONTENTS

ACKNOWLEDGEMENT	i
ABSTRACT	ii
LIST OF TABLES	iii
LIST OF FIGURES	iv
ABBREVIATIONS	v
Chapter 1 : INTRODUCTION	1
1.1 Scope	2
Chapter 2 : LITERATURE SURVEY	3
2.1 TRELLO-Web Based Project Management And Collabora- tion Tool	3
2.2 JIRA-Project Management And Issue Tracking Tool	3
2.3 Project Management System	3
2.4 Student Project Management System(SPMS)	4
2.5 Project Management System For Graduating Student Progress Monitoring	4
2.6 The Development Of a Final Year Project Management Sys- tem For Information Technology Programmes	5
Chapter 3 : PROPOSED SYSTEM	7
3.1 Problem Statement	7
3.2 Objectives	7
Chapter 4 : DESIGN	8
4.1 System Architecture	8
4.2 Uml Diagrams	9
4.2.1 Use Case Diagrams	9
4.3 Software Requirements	10
4.4 Hardware Requirements	10
Chapter 5 : WORK PLAN	11

5.1	Phase 1: Initial Stage.....	11
5.2	Phase 2: Intermediate Stage	11
5.3	Phase 3: Final Stage	12
5.4	Gantt Chart	12
5.5	Task Allocation.....	12
Chapter 6	: IMPLEMENTATION	14
6.1	Requirement Analysis	14
6.2	Planning	14
6.3	Design	14
6.4	Development	14
6.5	Integration and Testing.....	15
6.6	Deployment	16
Chapter 7	: RESULTS AND DISCUSSION	17
7.1	Results	17
Chapter 8	: CONCLUSION	26
REFERENCES		
APPENDICES		
LIST OF PUBLICATIONS		

ACKNOWLEDGEMENT

Many noble hearts contributed immense inspiration and support for the successful completion of the project. We are unable to express our gratitude in words to such individuals.

We would like to express our deep regard to Dr.Meenakshy K , Principal, Government Engineering College, Palakkad, for providing facilities throughout the works of our project.

We take this opportunity to express our profound gratitude to Dr.Sangeetha U, Head of the Department, Department of Information Technology, Government Engineering College, Palakkad, for providing permission and availing all required facilities for undertaking the project in a systematic way.We are extremely grateful to Dr.Rani M R, Assistant Professor, Department of Information Technology, Government Engineering College, Palakkad, who guided us with his kind, ordinal and valuable suggestions. We pay our deep sense of gratitude to Mr.Ebey S Raj and Mr.Vinayachandran K K , Project Coordinators, Department of Information Technology, Government Engineering College, Palakkad, for their valuable guidance, keen interest and encouragement at various stages of the project. We would also like to thank all the teaching and nonteaching staff of Department of Information Technology, Government Engineering College, Palakkad, for the sincere directions imparted and the cooperation in connection with the project.

We will be failing in duty if we do not acknowledge with grateful thanks to the authors of the references and other literatures referred in this project.

Last, but not the least, We take pleasant privilege in expressing our heartfelt thanks to our friends who were of precious help in completing this project.

ABSTRACT

OnTrackify is a comprehensive project monitoring website designed specifically for colleges to enhance the management of academic projects. This platform offers a centralized hub for coordinators, guides, and students to streamline coordination, communication, and oversight throughout the project lifecycle. Key features include progress tracking, document sharing, deadline reminders, calendar updates, chat functionalities, guidance and correction options, and individual profiles. The website "OnTrackify" helps colleges manage projects better, evaluate them more effectively, and foster academic success by providing real-time insights and facilitating collaborative work among stakeholders. It does this by giving real-time updates, making it easy for everyone involved to work together and stay on top of things.

LIST OF TABLES

2.1	Summary of Literature Survey	6
5.1	Task Allocation	13

LIST OF FIGURES

4.1	System Architecture.....	8
4.2	Use Case Diagram	9
5.1	Gantt Chart	12
7.1	Home Page.....	18
7.2	Login Page.....	18
7.3	Signup page	19
7.4	Student Dashboard.....	19
7.5	Previous Project Form	20
7.6	Previous Project View	20
7.7	Calendar.....	21
7.8	File Upload	21
7.9	Files View.....	22
7.10	Group Formation	22
7.11	Grouped Students View	23
7.12	Notification.....	23
7.13	Notification view	24
7.14	Evaluation of project for enrolled student.....	24
7.15	Evaluation form of each student	25
7.16	Evuated marks viewed by student in student panel	25
8.1	Enter Caption	

ABBREVIATIONS

CSS Cascading Style Sheets.

HTML Hypertext Markup Language.

PHP Hypertext Preprocessor.

XAMPP X-operating system, Apache, Mysql, Php, Perl.

CHAPTER 1

INTRODUCTION

OnTrackify is a comprehensive project tracking and monitoring website specifically designed to cater to the needs of college environments. It serves as a centralized platform where coordinators, guides, and students can collaborate seamlessly, ensuring that projects are executed with efficiency. In the realm of project management, coordinators play a pivotal role in coordinating various tasks. Their responsibilities include forming and managing teams effectively, overseeing project progress comprehensively, updating project schedules promptly, and driving change through thorough evaluations. These duties are integral to maintaining project momentum and ensuring successful outcomes within the organization.

Guides play a crucial role in mentoring and nurturing student innovation. OnTrackify provides guides with features to provide track student progress, and foster collaboration among project teams. Through interactive communication channels and assessment tools, guides can inspire creativity and guide students towards project excellence. This ensures that students receive the guidance and support they need to succeed in their projects.

Students are at the heart of every project, bringing creativity, dedication, and ambition to the table. OnTrackify empowers students by providing them with a user-friendly interface to manage tasks, share ideas, and collaborate with peers. From document sharing to progress tracking, students can showcase their skills and contribute meaningfully to project success. This fosters a culture of collaboration and teamwork, essential for achieving project objectives and academic success.

OnTrackify bridges the perspectives of coordinators, guides, and students, fostering collaboration and mutual responsibility. Seamless communication tools and project insights facilitate effective teamwork towards common project goals. With OnTrackify, collaboration becomes effortless, transparent, and productive, leading to improved project outcomes and academic success. OnTrackify empowers success by providing coordinators, guides, and students with the tools and resources they need to excel in project management. Join us on this journey as we unlock the full potential of project management with OnTrackify, driving excellence and innovation in college projects. OnTrackify is not just a project tracking tool; it's a platform for empowerment, collaboration, and success in college project management.

1.1 SCOPE

The scope of OnTrackify is comprehensive, encompassing a range of functionalities essential for efficient project management in college environments. The platform's primary objective is to enhance student-faculty communication through advanced communication tools like chat functionalities, notifications, and fostering collaboration . Additionally, OnTrackify facilitates efficient group formation by streamlining the process of creating and joining project teams, promoting teamwork and task allocation. It provides seamless facilities for uploading, sharing, and managing project documents for all team members.

Moreover, OnTrackify utilizes progress tracking tools to measure project performance in real-time, allowing students to monitor milestones, identify areas for improvement by communicating with guide. Lastly, OnTrackify serves as a repository of previous project ideas and references providing valuable insights and inspiration for current projects, fostering innovation, and promoting best practices in project management within college environments.

CHAPTER 2

LITERATURE SURVEY

Many research papers related to Project Monitoring Website were referred to and analyzed. A summary of them is given below.

2.1 TRELLO-WEB BASED PROJECT MANAGEMENT AND COLLABORATION TOOL

Trello is a user-friendly project management tool known for its visual interface and flexibility. It uses cards, boards, and lists to organize tasks and projects efficiently. With drag-and-drop functionality, users can easily move cards, assign tasks, set due dates, and collaborate with team members. Trello integrates seamlessly with other apps like Google Drive and Slack, enhancing collaboration and workflow management. It's highly customizable, allowing users to create personalized boards, lists, and cards. Trello promotes transparency and teamwork through features like comments, @mentions, and activity logs. Its mobile app enables on-the-go project management. Suitable for agile methodologies, Trello scales from small teams to large enterprises with adaptable pricing plans. Overall, Trello is valued for its simplicity, adaptability, and collaborative features, making it a popular choice for diverse teams.

2.2 JIRA-PROJECT MANAGEMENT AND ISSUE TRACKING TOOL

Jira, from Atlassian, is a versatile project management and issue tracking tool used globally. It supports agile methodologies like Scrum and Kanban, enabling teams to customize workflows and track tasks efficiently. With robust issue tracking features, seamless collaboration tools, and integration capabilities with other tools like Confluence and Slack, Jira streamlines project management, enhances team communication, and provides valuable insights through reporting and analytics. Overall, Jira's comprehensive features, adaptability to various methodologies, robust issue tracking capabilities, collaboration tools, reporting capabilities, and integration options make it a highly effective and popular choice for project management and issue tracking across industries.

2.3 PROJECT MANAGEMENT SYSTEM

Managing and controlling the final year projects of students using manual or traditional processes is a very tedious job. The main aim of the project is to create an automated system for managing all the activities of projects. The project management system is a

system for managing, controlling, and monitoring the final-year projects of students. It is a web-based portal or application that is useful for students, project coordinators, and project guides. Firstly, all the students need to register in the system using the registration form. Then registered students can log in to the system using their id and password to get authenticated. When the students log in to the system, then they will form the groups on their own. Similarly, the project guide needs to login into the system using their id and password. PMS also allows the group of students to provide at least three project domains and then the system will automatically assign the guides to the group of students. The project coordinator is the main aspect of the system which will assign different tasks to the students. The project coordinator and project guide coordinate with each other. Depending upon the different parameters, the work is assigned by the coordinator and the progress chart of the group is created. For creating a progress chart, we are using a work breakdown structure.

2.4 STUDENT PROJECT MANAGEMENT SYSTEM(SPMS)

It is very tedious to manage and control student's final year projects using manual or classical processes. The main goal of this project is to build an integrated framework to handle all project activities. Project management system is a system for the management, tracking and supervision of students' final year projects. It is a web-based platform or framework that is useful to students, project managers and project managers. Firstly, all students must register using the registration form. Then registered students can log in with their ID and password to authenticate. When the students log into the system, they form their own classes. In the same way, the project guide must log in through its id and password. PMS also provides the student group with at least three project domains, and the device automatically assigns the guides to the student group. The Project Leader is the key component of the system that assigns students various assignments. The project leader and project guide are coordinated. The job is delegated by the supervisor and the progress chart of the group is generated, depending on the various parameters. We use a job breakdown structure to build a progress map (WBS). The marks would automatically be assigned to a certain group of students, based on the progress charts

2.5 PROJECT MANAGEMENT SYSTEM FOR GRADUATING STUDENT PROGRESS MONITORING

Project management system is brought into development to overcome the existing manual transmission process of the project reports. The existing process is very hectic, time consuming and also can inculcate a numerous number of human errors. It will be providing a user-friendly platform for the users (students as well as teachers) to interact

with each other, share their progress and also maintain the deadlines accordingly. The special “PING” feature will be inculcated in the project that will help to notify the students if he/she is running behind schedule. also, the feature of automatic generation of the certificates i.e. the certificates will be provided automatically to the students once the entire process of the project has been completed is also an eye catcher. In all this will overcome the manual and time-consuming existing system and will help to make the process speedy and also error free mostly to be implicated at the institutional level.

2.6 THE DEVELOPMENT OF A FINAL YEAR PROJECT MANAGEMENT SYSTEM FOR INFORMATION TECHNOLOGY PROGRAMMES

This project tries to develop an online platform which facilitates the final year projects (FYP) process implemented by our information technology programme. The whole FYP is a year-long process involving groups of students and their supervisors to accomplish a theme based project. It is very necessary to employ the latest technologies in order to allow different parties to contribute and communicate more efficiently. Therefore, we have designed and developed a comprehensive web-based system to better support the three kinds of users; they are the FYP programme organizer (PO), project supervisors and the project group members. Before the academic year starts, PO will have to assign project topics according to the preferences and academic record of the formed project groups. This is a tedious and error prone task to manually complete the allocation of projects. The system is therefore useful to help PO to arrange project selection and allocation procedure, as well as helping the students to submit their preferences and group member information. During the academic year, the system would provide different functionalities for the PO to collect student’s assessments and markers’ scores via the submission and grading module. Project supervisors will also be able to employ the system for keep tracking the progress of the projects with the use of the project management tools, and online chat function. Both the students and supervisors will be benefited from these functionalities and allow the project to run more smoothly even face-to face meetings are not held very frequently. Project group members themselves are provided with similar the effective communication tools to allow easy of discussion on project issues among members. Moreover, they can share resources about the project including source code and data files using our online repository.

The summary of literature survey is given in table 2.1.

Author	Name	objective	Advantages	Disadvantages
Joel Spolsky Michael Pryor	Trello	-Provide a userfriendly platform for project management and task coordination. - Facilitate collaboration among team members. -Support agile methodologies for project development and task tracking.	- Easy-to-use interface with drag-and-drop functionality for task organization. -Real-time collaboration and updates for team members Integration with various thirdparty apps and services. -Customizable boards and lists for different project needs.	-Only for team collaboration. -May not be ideal for large-scale projects with complex workflows. -No student,guide, monitor interactions.
Atlassian	Jira	-Provide a comprehensive platform for project management, issue tracking, and agile development methodologies.	-Advanced issue tracking capabilities with customizable workflows. -Integration with development tools like Git, Jenkins, and Confluence. -Extensive reporting and analytics for project progress and performance monitoring.	-Require time and effort for learning. -Higher cost compared to some other project management tools. -Complex and overwhelming especially for college projects,
Sowmiya S, Yaswanthan S R	Project Management System(PMS).	-Managing, controlling and monitoring the final-year projects useful for students, project coordinators, and project guides.	-Simplified process of assigning supervisors. -Option for project submission and approval.	-No database for previous project. -No group formation. No communication platform
Dr.Nisha Soms	Student Project Management System(SPMS). Year:2021	-Eliminates the additional time and resources needed to plan.	-The average time spent in contrast with the current system would be lower. - The device shall be validated and checked.	-No database for previous database. -No track bar to show the progress.
K Thanuja , P Saktheram Reddy , P Sharuk Khann, Gunavardhan Reddy	Project Management System For Graduating Student Progress Monitoring. Year :2023	-Facilitate online processing and submission of project reports. -Generate accurate results by providing clear guidelines and evaluation criteria for project reports.	-It reduce miscommunication. -It make the process faster.	-No group formation - No previous year project ideas and references
Chun-Hang Leung, Chung-Lun Lai, Tsun-Kit Yuan, Wai-Man Pang, Jeff K.T. Tang, Wai-Shng Ho, Tak-Lam Wong	The Development Of A Final Year Project Management System For Information Technology Programmes.	-Project allocation, project management, file sharing & repository.	-Reduce the workload of program organizer. - Lots of arrangement, announcement and assessment collection tasks can be done automatically.	-Supervisors cannot provide grades on the system. -Chat and messaging tools in the system are absent.

Table 2.1: Summary of Literature Survey

CHAPTER 3

PROPOSED SYSTEM

The proposed system for the Project Monitoring System are as follows:

- Creating a system to keep track of college projects, meeting specific project needs.
- A project calender for important project updates.
- Including features to see how projects are progressing, like progress bar for completed tasks.
- Adding ways for students working on projects to chat with guide and share files easily within the system.
- Build a place where students can see the previous projects as references.

3.1 PROBLEM STATEMENT

Design a project monitoring website for college projects that facilitates team-guide communication, project submission, group formation, progress tracking, project evaluation processes to ensure academic success.

3.2 OBJECTIVES

The objective of this project are mentioned below.

1. To enhance student faculty communication.
2. To facilitate group formation.
3. To provide facilities for uploading documents.
4. To use progress tracking to measure performance.
5. To offers grading and evaluation.
6. To showcase previous project ideas.

CHAPTER 4

DESIGN

4.1 SYSTEM ARCHITECTURE

The system design of the Project Monitoring Website is given below. The project Monitoring website for colleges system architecture consists of dashboards and a shared database. The first dashboard is dedicated to coordinators and enables real-time monitoring and management of project in colleges. The second dashboard is designed for Guide, providing them with group and work information and tools for efficient academic project success. The third dashboard is designed for students, providing them with team work and academic information and tools for efficient coordination of team. The architecture ensures seamless communication between the dashboards, allowing coordinators and guide to track performance and make data-driven decisions, while guide access relevant information and tools. A shared database serves as a centralized repository, storing and managing critical data, including student information, project progress, and historical performance metrics. This architecture facilitates efficient data flow, enhances project monitoring operations, and boosts productivity for the academic sessions.

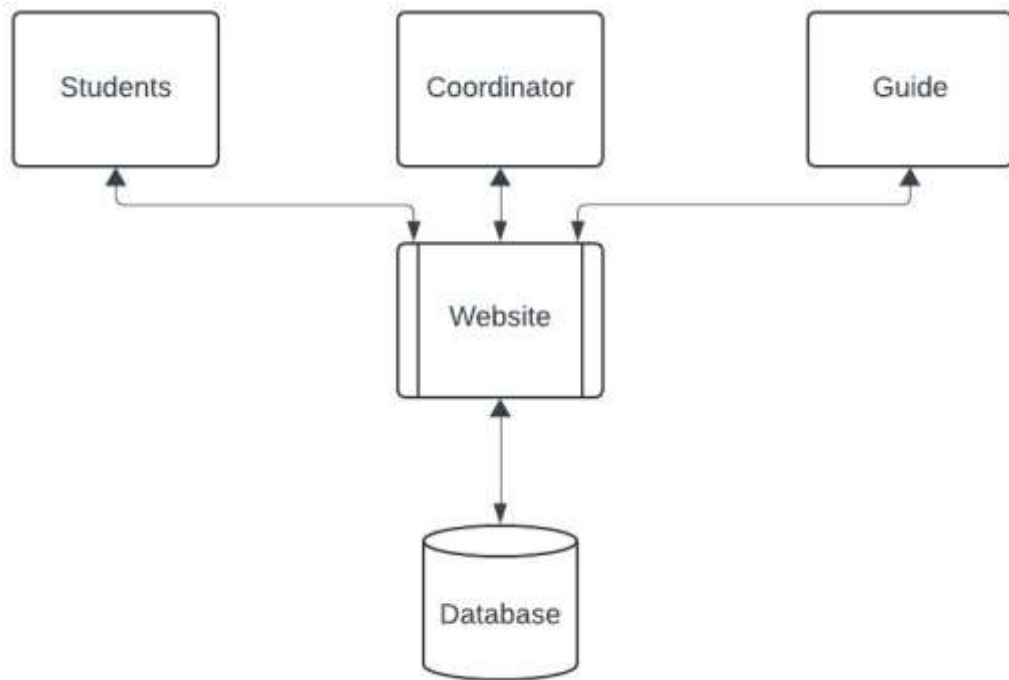


Figure 4.1: System Architecture

4.2 UML DIAGRAMS

A UML (Unified Modeling Language) diagram is a graphical representation used to visualize, specify, construct, and document the structure and behavior of a system.

4.2.1 Use Case Diagrams

A use case diagram is a visual representation of the functional requirements of a system from the user's perspective. It consists of actors, use cases, and their relationships. The following sections discuss UML diagrams used in the project.

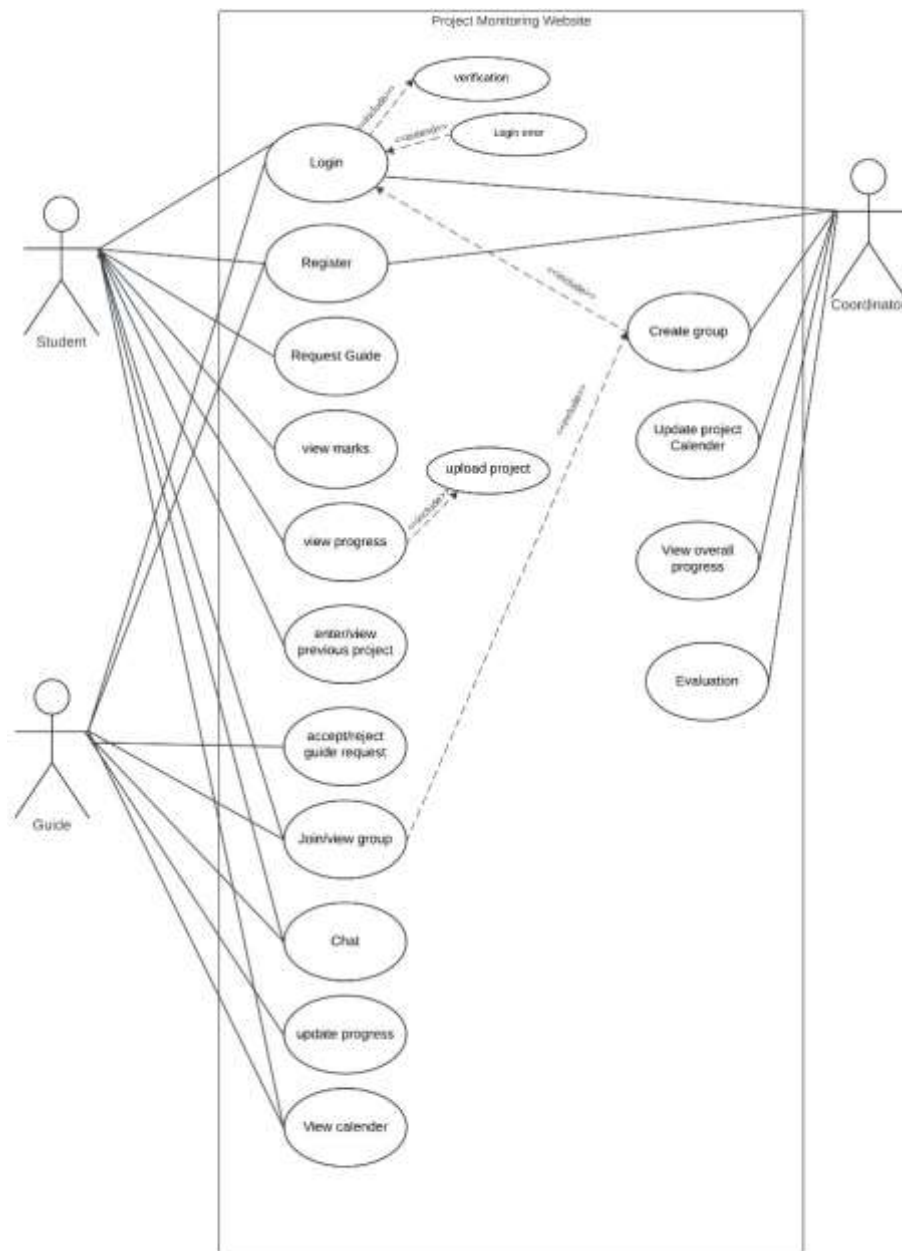


Figure 4.2: Use Case Diagram

4.3 SOFTWARE REQUIREMENTS

- Operating System: Microsoft Windows 11 Pro
- Development Environment: Visual Studio Code
- Frontend Technologies: HTML5, CSS3, JavaScript
- Backend Technology: PHP 8.2.12
- Database: MySQL(via XAMPP)

4.4 HARDWARE REQUIREMENTS

- System : Intel Core I3 Processor
- Hard Disk : 221 GB.
- Ram : 8 GB.

CHAPTER 5

WORK PLAN

5.1 PHASE 1: INITIAL STAGE

The project proposal was submitted to our guide. Our ideas include a project monitoring website for college, a blog app for colleges, and a leftover food collector website. Zeroth review: project proposals were submitted, and the panel approved the project monitoring website for colleges idea with suggestions. The panel instructed us to study more about the topic.

5.2 PHASE 2: INTERMEDIATE STAGE

We started working on the literature survey and referred to papers that were useful to our projects. Our guide instructed each one of us to do literature survey on 1 or more papers. The papers that have been referred to are listed below:

- Trello-web based project management and collaboration tool by Joel Spolsky and Michael Pryor.
- Jira: Project management and issue tracking tool by Atlassian.
- Project Management System by Sowmiya S and Yaswanthan S R.
- Student Project Management System(SPMS) by Dr .Nisha Soms.
- Project Management System For Graduating Student Progress Monitoring by K Thanuja , P Saktheram Reddy , P Sharuk Khann and Gunavardhan Reddy.
- The Development Of A Final Year Project Management System For Information Technology Programmes by Chun-Hang Leung, Chung-Lun Lai,Tsun-Kit Yuan, Wai-Man Pang, Jeff K.T. Tang, Wai-Shng Ho, Tak-Lam Wong.

We started working on ppt slides,design,UML diagrams. We met the guide, and cleared our doubts about the format to be followed . Incorporated the changes recommended by our guide and submitted UML diagram. First review: The panel suggested some changes in Objectives,Problem Statement,UML diagram. They also instructed us to modify system architecture and requirements. We met with our guide and discussed with her about the suggested changes and incorporated them to our project.We prepared an initial outline of system architecture and met our guide. She recommended some

changes and those were updated. The panel advised us to start implementation of the project.

5.3 PHASE 3: FINAL STAGE

We finalized the programming languages and technologies to be used, divided the project within our teammates. Our project was done in html,css,js with Php as backend and XAMPP server. We learnt XAMPP, html, css,js from scratch and built the Website. The Gantt chart for the work plan is given below.

5.4 GANTT CHART

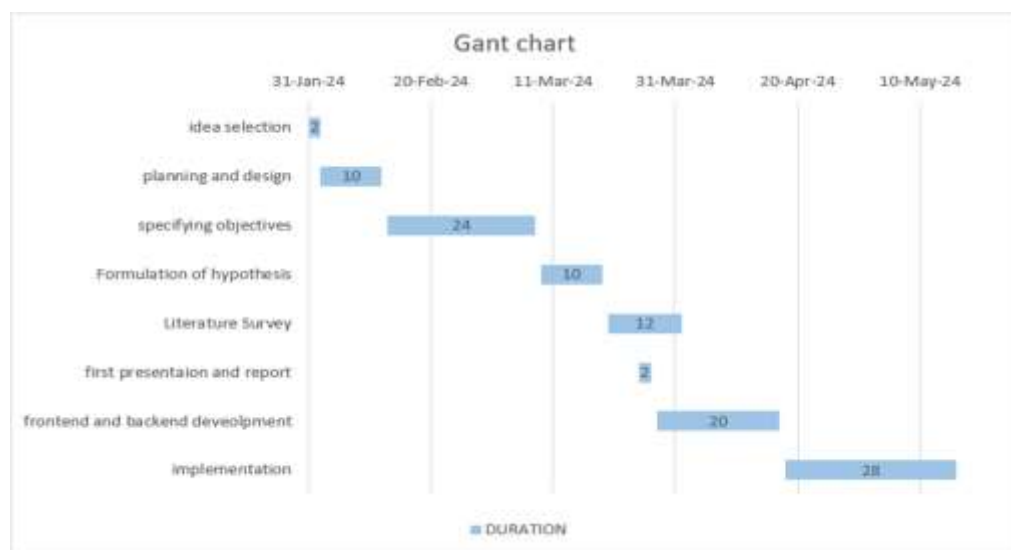


Figure 5.1: Gantt Chart

5.5 TASK ALLOCATION

Table 7.1 illustrates the distribution of tasks among the team members in the project. Each team member has been assigned specific responsibilities based on their expertise and skills. The allocation aims to optimize efficiency and collaboration within the team, ensuring the successful execution of the project's objective.

Task	Krishnapriya P H	Niya Prakash	Andric Antony	Raseena P A
Literature Sur vey	Project Manageme nt System	Student project manageme nt system (SPMS)	The Developme nt of a Final Year Project Manageme nt System for Information Technology Programme	Project management system for graduating student progress
Formulation of Objectives				
Design	System Architecture	Interface of dashboard	Chat room	Use case Dia gram
Preliminary Analysis				
Phase-1 Re port				
Sample Project				
	Code for Previous project updates, Registratio n and login	Code for Dynamic calender,st udent list,Evaluati on.	Code for web design, chat room, file upload option, video call option	Develop code for forgot password, Request Guide,Guid e join
Phase-2 Re port				

Table 5.1: Task Allocation

CHAPTER 6

IMPLEMENTATION

The implementation of OnTrackify involved several steps to develop the web-based project monitoring platform for colleges. Here are the detailed implementation steps:

6.1 REQUIREMENT ANALYSIS

- Stakeholder Meetings: Conduct meetings with students, guides, and coordinators to gather detailed requirements.
- Feature Identification**: Identify key features needed: document sharing, progress tracking, communication tools, notifications, group management, and evaluation mechanisms.

6.2 PLANNING

- Project Plan: Create a detailed project plan with timelines and milestones.
- itemScope and Objectives: Define the scope and objectives of OnTrackify.
- Tools and Languages: Choose simple tools and languages: HTML, CSS, JavaScript for front-end, and PHP with MySQL for back-end.

6.3 DESIGN

- UI/UX Design: Develop wireframes and mockups for the user interface, including screens for login, project tracking, document sharing, and communication.
- Database Schema: Design the database schema to support the platform's functionalities.
- User Flow Diagrams: Create user flow diagrams to map out the interactions within the platform.

6.4 DEVELOPMENT

- Development Environment Setup: Set up the development environment using Visual Studio Code (VSCode) and XAMPP.
- Front-End Development: Implement the front-end components using HTML, CSS, and JavaScript.

- **Login and Registration:** Create pages for user registration and login.
- **Dashboard:** Develop a dashboard for students, guides, and coordinators to track project progress.
- **Document Sharing:** Implement functionality for uploading, sharing, and downloading documents.
- **Communication Tools:** Develop messaging and notification features to facilitate communication between users.
- **Progress Tracking:** Design interfaces for updating and viewing project progress, including milestones and deadlines.
- **Back-End Development:** Develop the back-end logic and database interactions using PHP and MySQL.
- **Database Connection:** Set up database connections and queries to interact with the MySQL database.
- **User Management:** Implement CRUD operations for managing user profiles, roles, and permissions.
- **Project Management:** Develop backend functionalities for creating, updating, and deleting projects and tasks.
- **Document Management:** Handle file uploads, storage, and retrieval operations.
- **Notification System:** Implement a backend system to manage notifications and alerts for project updates and deadlines.
- **Evaluation Mechanisms:** Develop features for project evaluation, including feedback and grading systems.

6.5 INTEGRATION AND TESTING

- **Feature Integration:** Ensure seamless integration of front-end and back-end components.
- **Unit Testing:** Conduct unit testing on individual components to ensure functionality.
- **Integration Testing:** Perform integration testing to verify that all components work together seamlessly.

6.6 DEPLOYMENT

- Deployment: Deploy OnTrackify on a web server using XAMPP.

By following these steps, OnTrackify was successfully developed and implemented, resulting in a robust and user-friendly project monitoring solution for our college. This structured approach ensured that all aspects of the platform were carefully considered and executed, leading to its successful deployment and adoption.

CHAPTER 7

RESULTS AND DISCUSSION

The OnTrackify project developed a web-based application using HTML, CSS, JavaScript for the frontend, PHP for the backend, and MySQL for the database. Secure login procedures ensured safe access for students, guides, and coordinators, each with distinct roles. The system effectively managed essential data, including user profiles, project details, and documentation.

Coordinators could manage projects, assign students, and monitor overall progress through performance analytics. Students could view project, upload documents, notifications and communicate with guides and coordinators. Guides could oversee student progress, provide feedback, and evaluate submissions.

The implementation details showcased the successful integration of HTML, CSS, JavaScript, PHP, and MySQL technologies, resulting in a user-friendly and interactive interface. The integration with the MySQL database ensured seamless storage and retrieval of data. The functionalities provided valuable insights for coordinators to track project progress and make informed decisions, while the student and guide dashboards facilitated efficient project management and communication. This successful implementation demonstrates the value of technology in enhancing academic project management and sets the foundation for future updates and improvements to meet evolving needs.

7.1 RESULTS

The home page is given in figure 7.1. OnTrackify prominently features the logo, login, and signup buttons at the top for easy access. Below this, there's a welcoming slogan "Your Project Companion," emphasizing the platform's purpose. The home page is designed to provide a user-friendly experience, helping users navigate and utilize the project management tools efficiently.



Figure 7.1: Home Page

The student login page is given in figure 7.2. OnTrackify features the logo at the top. Below, there are fields for entering email and password, along with a prominent login button. Additionally, there are links for signing up and resetting forgotten passwords.

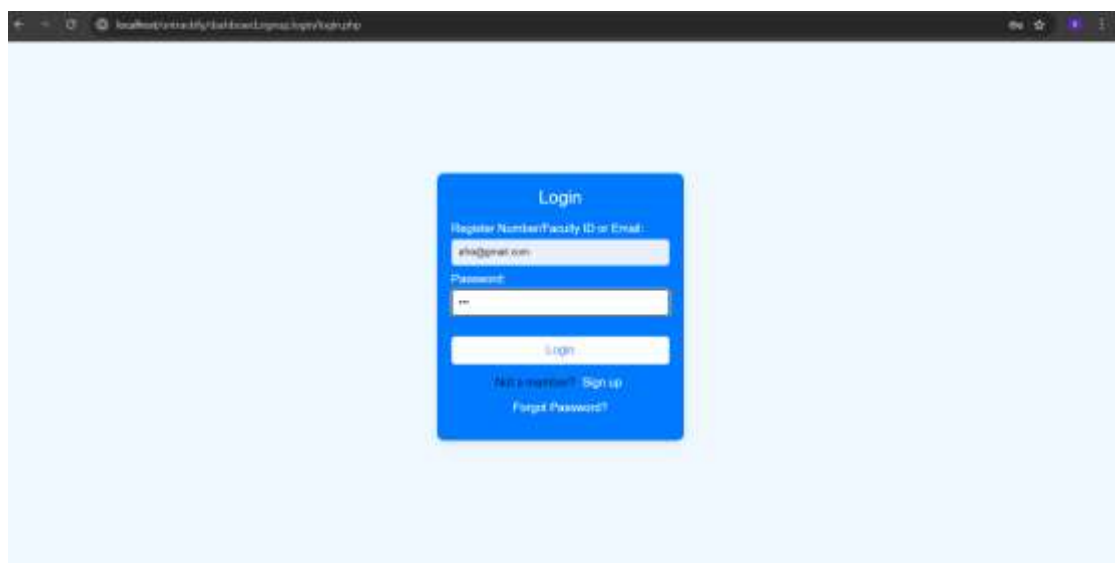


Figure 7.2: Login Page

The signup page given in fig 7.3 which provides roles for student,coordinator,guide to signup.

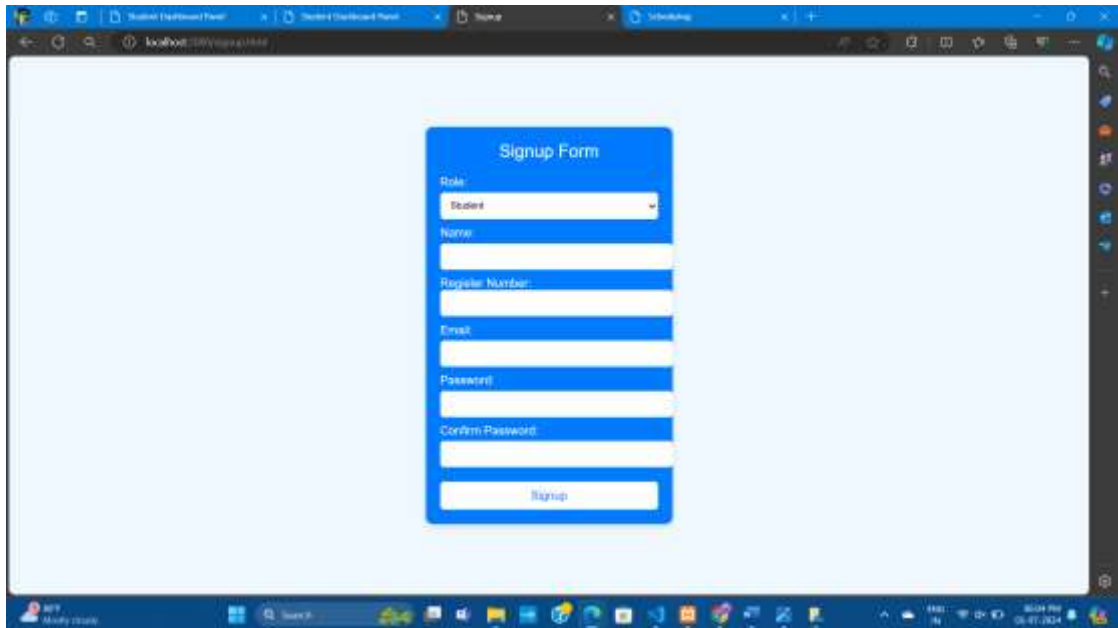
A screenshot of a web browser displaying the 'Signup Form' page. The form is centered on a light blue background. It has a title 'Signup Form' at the top. Below the title, there is a 'Role' dropdown menu with 'Student' selected. Following this are input fields for 'Name', 'Register Number', 'Email', 'Password', and 'Confirm Password'. At the bottom of the form is a 'Signup' button. The browser's address bar shows 'localhost:3000/signup'. The Windows taskbar is visible at the bottom of the screen.

Figure 7.3: Signup page

The OnTrackify student dashboard is given figure 7.4 which includes a personalized profile and notifications for important updates. It features a project calendar and allows easy access to previous projects and their marks. Additionally, students can upload assignments and other relevant documents seamlessly.

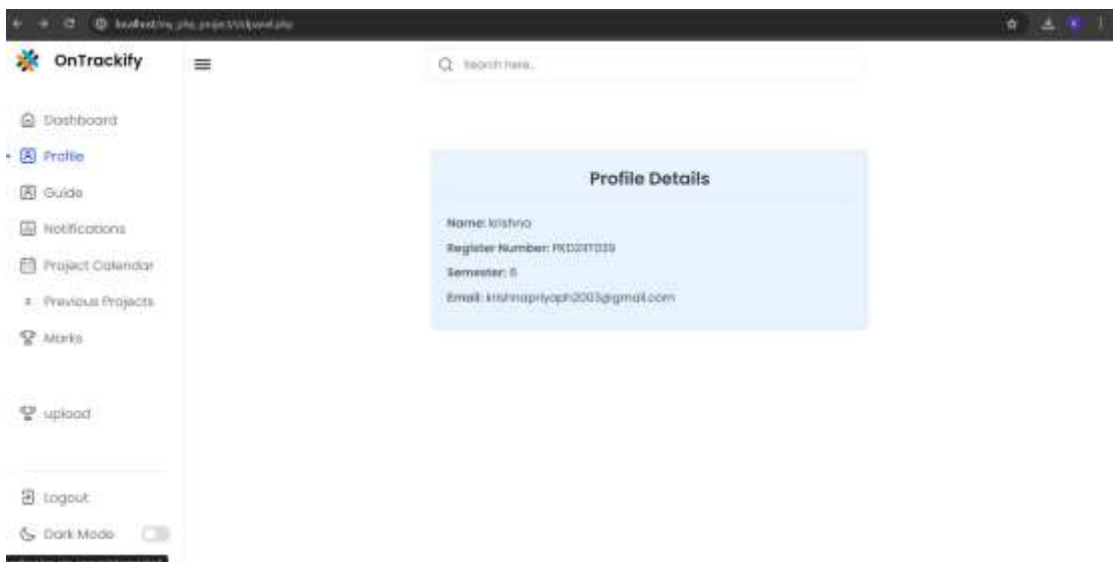
A screenshot of the OnTrackify student dashboard. The page has a dark header with the 'OnTrackify' logo and a search bar. A left sidebar contains a menu with items: Dashboard, Profile (highlighted), Guide, Notifications, Project Calendar, Previous Projects, Marks, upload, logout, and Dark Mode. The main content area features a 'Profile Details' card with the following information: Name: krishna, Register Number: RC0201039, Semester: 5, and Email: krishnapriyash2003@gmail.com. The Windows taskbar is visible at the bottom.

Figure 7.4: Student Dashboard

Previous Project form is given in figure 7.5 which features a project details form where users can enter project name, description, group members, and contact number and references.

Enter Project Details

Project Name:

Description:

Group Members:

Contact Number:

References:

Figure 7.5: Previous Project Form

Previous project view is given figure 7.6 provides a previous project view for referencing and reviewing different projects implemented in the past.

Previous Projects

Project Name	Description	Group Members	Contact Number	References
to_do_app	kuhwkpwel	arglha	4455455225	wdrpwlqjkgd
to_do_app	kuhwkpwel	arglha	4455455225	wdrpwlqjkgd
to_do_app	kuhwkpwel	arglha	4455455225	wdrpwlqjkgd
website1	ll	llll	555545	llll

Figure 7.6: Previous Project View

OnTrackify's calendar is given in figure 7.7 provides scheduling form allows coordinators to specify the title, description, start date, and end date for scheduling important project milestones and events.

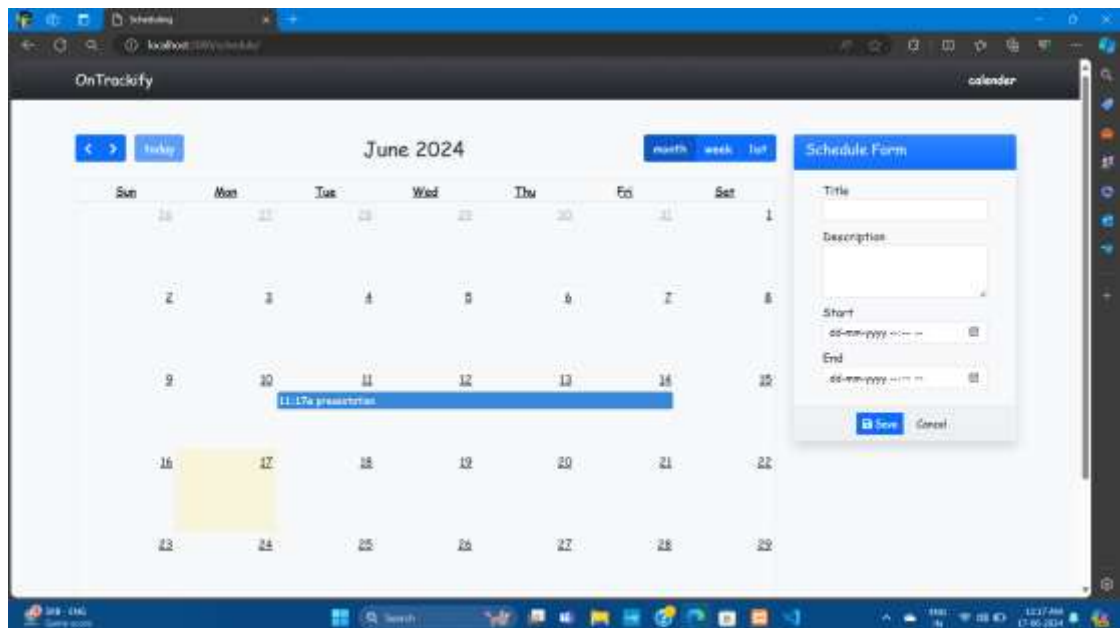


Figure 7.7: Calendar

File upload is given in figure 7.8 which provides students with a file upload option for submitting various file types including PPTs, code files, PDFs, images, and more.

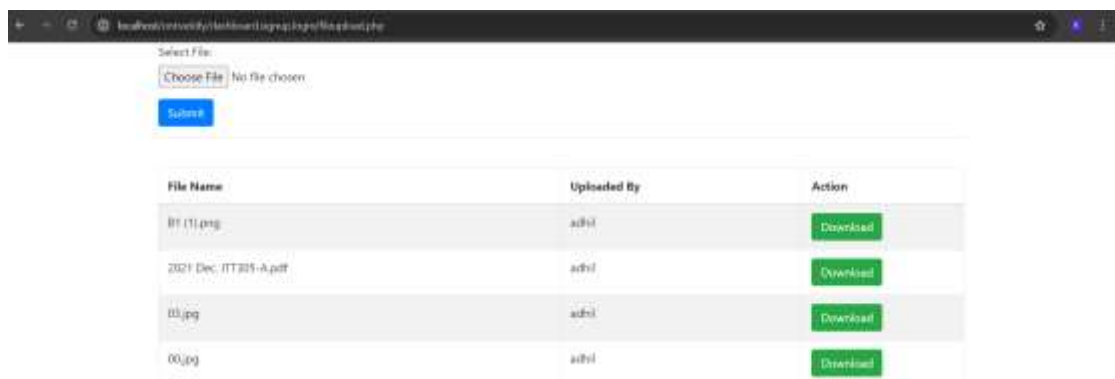


Figure 7.8: File Upload

OnTrackify's File view is given in figure 7.9 which offers guides a file upload view to manage and review submitted documents from student



Figure 7.9: Files View

The group formation is given in figure 7.10 feature for coordinators in OnTrackify allows them to assign students to groups by selecting members using checkboxes and finalizing the group formation with a "Form Group" button. This streamlined process ensures efficient management of student teams for collaborative projects.

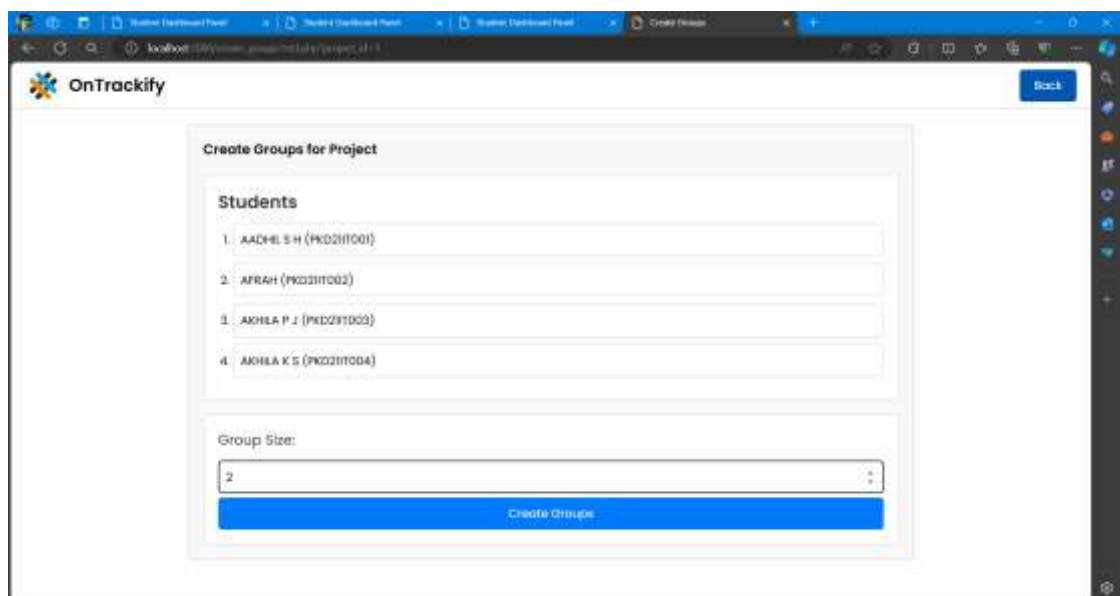


Figure 7.10: Group Formation

Grouped Students view is given in figure 7.11 that allows coordinators to view grouped students and send group join requests to individual students.

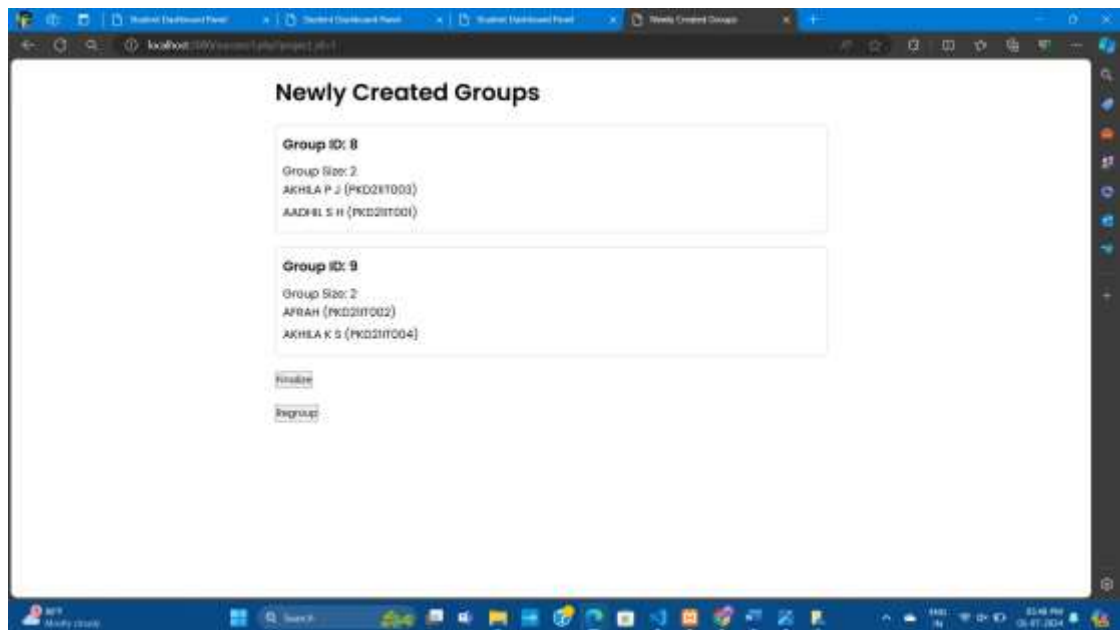


Figure 7.11: Grouped Students View

A notification system is given 7.12 where coordinators can create, edit, delete, and send important notifications to students.

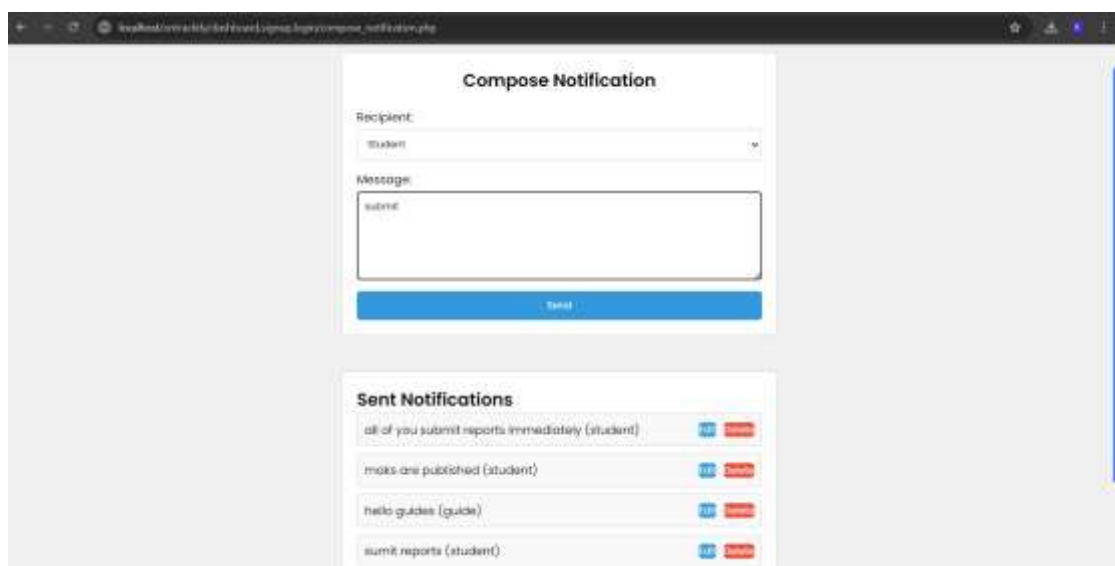


Figure 7.12: Notification

A notification view is given in 7.13 where students can view the notifications sent by coordinator



Figure 7.13: Notification view

The OnTrackify Evaluation system by coordinator is given fig 7.14 where coordinator can make evaluation for each students on assigned project.

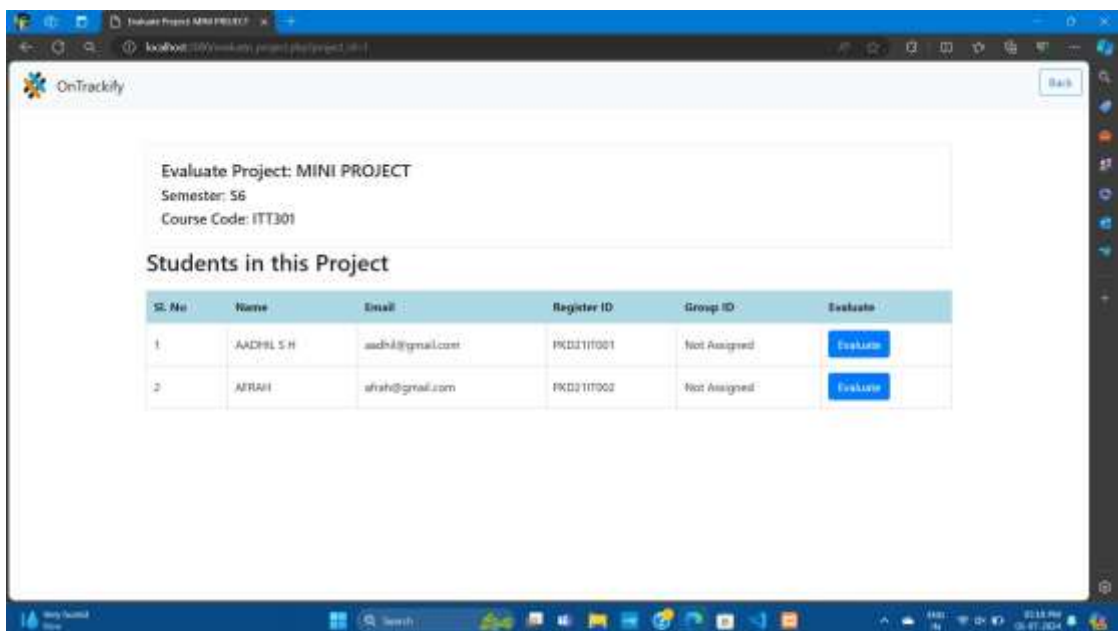


Figure 7.14: Evaluation of project for enrolled student

Evaluation for each student with a form is given in Fig 7.15 .Guide's marks are fetched directly from guide

Evaluate Student
 Project: MPH PROJECT
 Student: AADHI, S H
 Email: aadhil@gmail.com
 Register ID: PK0210001
 Group ID: Not Assigned

Attendance (10 Marks)	10
Guide Marks (15 Marks)	0
Project Report (10 Marks)	10
Review 1 Total Marks	33
Review 2 Total Marks	33
Average Review Marks	33
Final CIE Mark (Total: 75)	53.00

Buttons: Save, Cancel

Figure 7.15: Evaluation form of each student

The evaluated marks for each student is displayed on their student panel is given in Fig 7.16

My Marks

Student Details
 Name: AADHI, S H
 Register ID: PK0210001
 Group ID: Not Assigned

Marks Category	Marks Obtained
Attendance (out of 10)	10
Guide Marks (out of 15)	
Project Report (out of 10)	10
Review 1 Total Marks (out of 40)	33
Review 2 Total Marks (out of 40)	33
Total CIE Marks (out of 75)	53.00

Figure 7.16: Evuated marks viewed by student in student panel

CHAPTER 8

CONCLUSION

Finally, we have successfully developed and implemented OnTrackify, our project monitoring website for colleges. OnTrackify is now fully operational, providing a range of features that make managing student projects easier and more efficient.

The platform includes tools for document sharing, progress tracking, communication, notifications, group management, and evaluation. These features help students, guides, and coordinators stay organized and on track with their projects.

The user-friendly interface and easy-to-navigate design have made it easy for everyone to use OnTrackify effectively. The dynamic calendar has been especially helpful in keeping projects on schedule and ensuring timely guidance.

Overall, OnTrackify has improved project management and communication within our college. It has made the process more streamlined and effective, supporting both students and faculty in achieving their project goals. This successful implementation demonstrates the value of technology in enhancing academic project management.

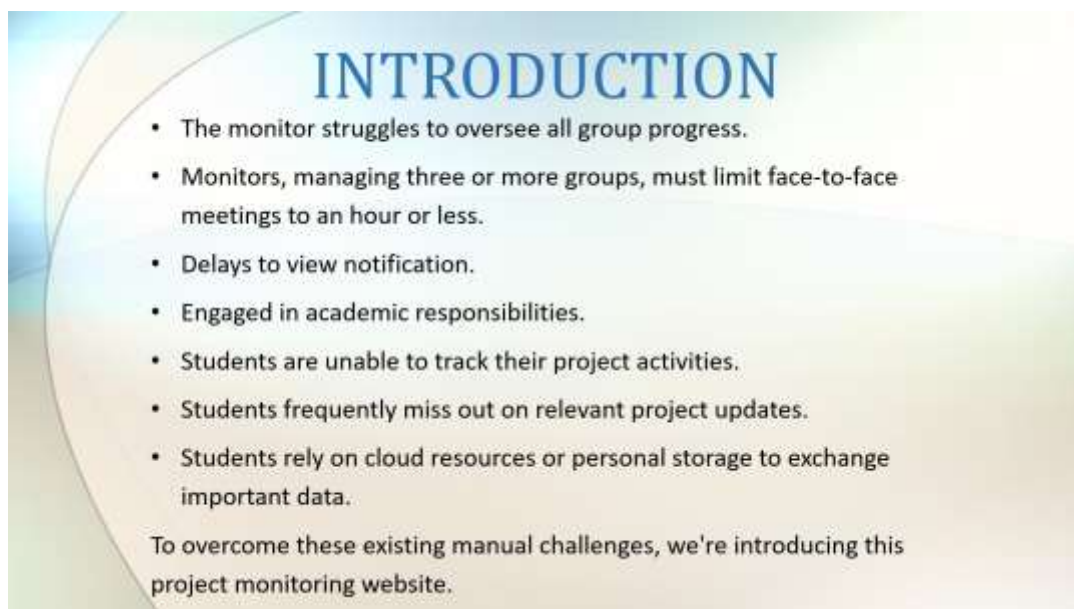
However, it is important to note that this is just a sample of what OnTrackify can achieve. Further updates and improvements will be required to keep the platform aligned with evolving needs and technological advancements. Continuous enhancements will ensure that OnTrackify remains a robust and valuable tool for project management in the academic environment.

REFERENCES

- [1] Atlassian. Jira project management and issue tracking tool. *software development and project management contexts*, 2002.
 - [2] Atlassian. Trello web based project management and collaboration tool. *software application for project management and collaboration*, 2021.
 - [3] N Gunavardhan Reddy P sharukhan K Thanuja, P Saktheram Reddy. Project management system for graduating student progress. *International journal of Computational Learning and intelligence*, 2023.
 - [4] Dr.Nisha Soms. Student project management system(spms). *International Journal of All Research Education and Scientific Methods(IJARESM)*, 2021.
 - [5] Yaswanathan S R Sowmya S. Project management system. *Department Of Information Technology School Of Computing*, 2023.
 - [6] Chun-Hang Leung; Chung-Lun Lai; Tsun-Kit Yuan Wai-Man Pang; Jeff K.T. Tang;Wai-Shng Ho;Tak-Lam Wong. The development of a final year project management system for information technology programmes. *Department of Computer Science, Caritas Institute of Higher Education, Hong Kong SAR, China*, 2015.
- [2] [1] [5] [4] [6] [3]

APPENDIX-A

A.1 PRESENTATION



LITERATURE SURVEY

1:TRELLO-WEB BASED PROJECT MANAGEMENT AND COLLABORATION TOOL

- Trello provides a user-friendly Kanban board interface with features like task cards, checklists, and file attachments.
- It is designed for simpler task management and collaboration, making it easy for teams to organize and track their projects visually.

<https://trello.com/>

2.JIRA:PROJECT MANAGEMENT AND ISSUE TRACKING TOOL

- Jira offers features such as customizable workflows, extensive reporting and analytics, and integration with development tools.
- It is designed to handle complex project management needs and is commonly used in professional software development environments.

www.atlassian.com/jira

3.PROJECT MANAGEMENT SYSTEM

- A simple web portal anyone can carry out their project-related work which is the main aim of this Project Management System (PMS).
- The system has three main modules ,project coordinator module , project guide module ,student module.
- The paper aims to simplify the process of assigning supervisors and submitting projects of students which is fully automated with document submissions and approvals.
- Using a work breakdown structure, progress chart is created.

SOWMIPA S ,TASWANTHAN S R.2023 INSTITUTE OF SCIENCE AND TECHNOLOGY
(RESEARCH GATE)year:2023

4.STUDENT PROJECT MANAGEMENT SYSTEM(SPMS)

- It is very tedious to manage and control student's year projects using manual or classical processes.
- Managing tasks manually for the final year is a very stressful task. But anyone with a simple web platform can do project-related work.
- The main goal of this project is to build an integrated framework to handle all project activities.
- It offers a web portal to students, project coordinator and project manager for the management and monitoring of project activities.

Dr. Nisha Senu,2021 International Journal of All Research Education and Scientific
Methods(IJARESME)
(RESEARCH GATE)year:2023

5. PROJECT MANAGEMENT SYSTEM FOR GRADUATING STUDENT PROGRESS MONITORING

- It provides a user-friendly platform for the users to interact with each other share their progress and also maintains deadlines.
- It use agile software development methodology.
- Ping feature for notify the students if he/she running behind the schedule.
- Automatic certificate generation after completion of project.

K. Thirugaj; P. Sakethram Reddy, P. Shanuk Shan, N. Gunavardhan reddy 2023 School of Computer Science and Engineering,
 REVA University
 @Microsoft Research Publications, Part of CLOCESS-archiving
 | RESEARCH GATE | year: 2023

6.The Development of a Final Year Project Management System for Information Technology Programmes

- This project tries to develop an online platform which facilitates the final year projects (FYP) process implemented by information technology programmes.
- Project supervisors will also be able to employ the system for tracking the progress of the projects with the use of the project management tools.
- Moreover, they can share resources about the project including source code and data files using online repository.

Chun-Hang Leung; Chung-Sun Lai; Tsun-Kit Yuen; Wei-Min Pang; Jeff S.T. Tang; Wei-Sheng Ho; Tai-Lam Wong
 2015 Department of Computer Science, Caritas Institute of Higher Education, Hong Kong SAR, China
 | RESEARCH GATE | year: 2015

SUMMARY OF LITERATURE SURVEY

Author	Name	objective	Advantages	Disadvantages
Joel Spolsky Michael Pryor	Trello	<ul style="list-style-type: none"> • Provide a user-friendly platform for project management and task coordination. • Facilitate collaboration among team members. • Support agile methodologies for project development and task tracking. 	<ul style="list-style-type: none"> • Easy-to-use interface with drag-and-drop functionality for task organization. • Real-time collaboration and updates for team members. • Integration with various third-party apps and services. • Customizable boards and lists for different project needs. 	<ul style="list-style-type: none"> • Only for team collaboration. • May not be ideal for large-scale projects with complex workflows. • No student, guide, monitor interactions.
Atlassian	Jira	<ul style="list-style-type: none"> • Provide a comprehensive platform for project management, issue tracking, and agile development methodologies. 	<ul style="list-style-type: none"> • Advanced issue tracking capabilities with customizable workflows. • Integration with development tools like Git, Jenkins, and Confluence. • Extensive reporting and analytics for project progress and performance monitoring. 	<ul style="list-style-type: none"> • Require time and effort for learning. • Higher cost compared to some other project management tools. • Complex and overwhelming especially for college projects.

Sowmiya S, Yashwanth S R	Project Management System(PMS).	<ul style="list-style-type: none"> • Managing, controlling and monitoring the final-year projects useful for students, project coordinators, and project guides. 	<ul style="list-style-type: none"> • Simplified process of assigning supervisors. • Option for project submission and approval. 	<ul style="list-style-type: none"> • No database for previous project. • No group formation. • No communication platform.
Dr.Nisha Soms	Student Project Management System(SPMS). Year:2021	<ul style="list-style-type: none"> • Eliminates the additional time and resources needed to plan. 	<ul style="list-style-type: none"> • The average time spent in contrast with the current system would be lower. • The device shall be validated and checked. 	<ul style="list-style-type: none"> • No database for previous database. • No track bar to show the progress.
K Thanuja , P Saktharan Reddy , P Sharuk Khan, Gunavardhan Reddy.	Project Management System For Graduating Student Progress Monitoring. Year :2023	<ul style="list-style-type: none"> • Facilitate online processing and submission of project reports • Generate accurate results by providing clear guidelines and evaluation criteria for project reports 	<ul style="list-style-type: none"> • It reduce miscommunication. • It make the process faster. 	<ul style="list-style-type: none"> • No group formation • No previous year project ideas and references
Chun-Hang Leung, Chung-Lun Lai, Tsun-Kit Yuan, Wai-Man Fung, Jeff K.C. Tang, Wai-Shing Ho, Tak-Lam Wong	The Development Of A Final Year Project Management System For Information Technology Programmes.	<ul style="list-style-type: none"> • Project allocation, project management, file sharing & repository 	<ul style="list-style-type: none"> • Reduce the workload of program organizer. • Lots of arrangement, announcement and assessment collection tasks can be done automatically. 	<ul style="list-style-type: none"> • Supervisors cannot provide grades on the system. • Chat and messaging tools in the system are absent.

PROBLEM STATEMENT

To design a project monitoring website for college projects that facilitates team-guide communication, project submission, group formation, progress tracking, project evaluation processes to ensure academic success.

OBJECTIVES

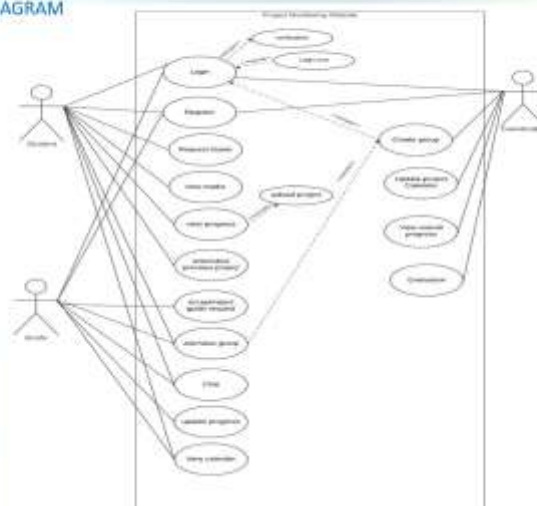
- Enhance student faculty communication.
- Facilitate group formation.
- Provide facilities for uploading documents.
- Use progress tracking to measure performance.
- Offers grading and evaluation.
- Showcase previous project ideas.

PROPOSED SYSTEM

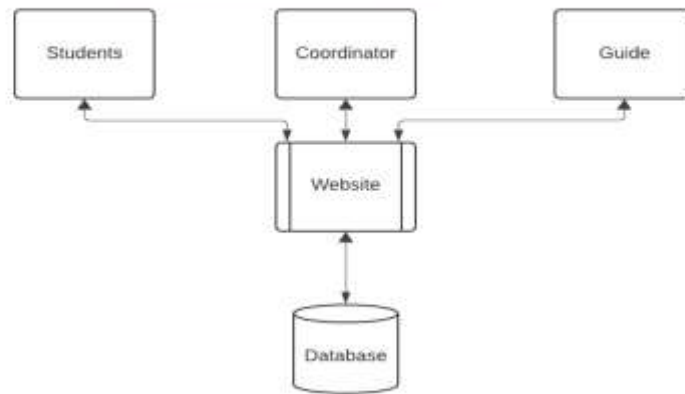
- Creating a system to keep track of college projects, meeting specific project needs.
- A project calendar for important project updates.
- Including features to see how projects are progressing, like progress bar for completed tasks.
- Adding ways for students working on projects to chat with guide and share files easily within the system.
- Build a place where students can see the previous projects as references.

DESIGN

USECASE DIAGRAM



SYSTEM ARCHITECTURE



DESIGN

A screenshot of a web browser displaying a login form. The form is titled 'Login' and is contained within a blue rectangular box. It includes three input fields: 'Username' (with a placeholder 'Please enter Username'), 'Password' (with a placeholder 'Please enter Password'), and a 'Login' button. Below the input fields, there are two links: 'Forgot Password?' and 'Register?'. The browser's address bar shows a URL starting with 'localhost'.

SYSTEM REQUIREMENTS

Hardware requirements:

- System: Intel Core i3 Processor
- Hard Disk: 221 GB
- RAM: 8GB

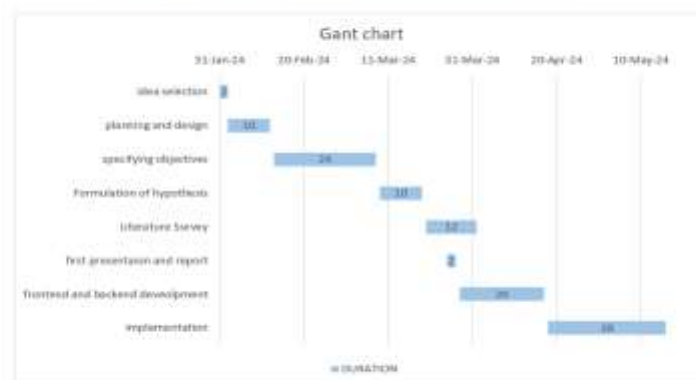
Front End (Designing):

- HTML5
- CSS3
- JavaScript

Back End:

- Programming Language - PHP
- Database - MySQL

WORK SCHEDULE



IMPLEMENTATION

Development:

- Set up development environment using VSCode and XAMPP.
- Implement front-end:
 - User registration/login.
 - Dashboard for Student, Guide and Coordinator.
 - Document sharing ,progress tracking and communication tools.
- Develop back-end:
 - Database setup and queries (PHP, MySQL).
 - User and project management.
 - Document and notification systems.

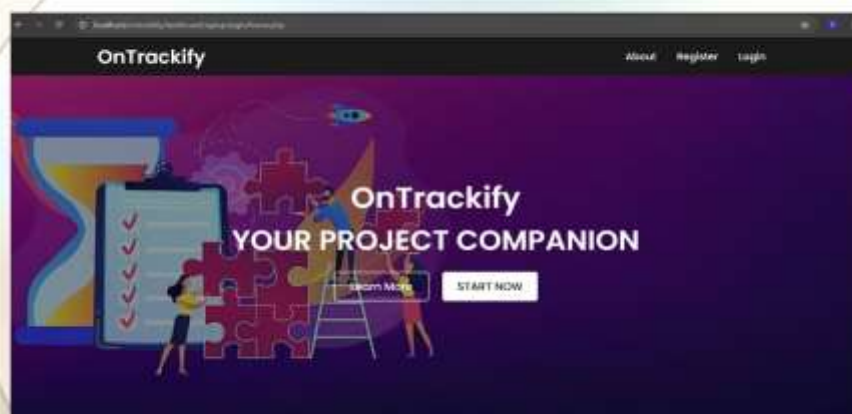
Testing:

- Ensure integration and conduct unit testing.
- Perform integration testing for functionality and performance.

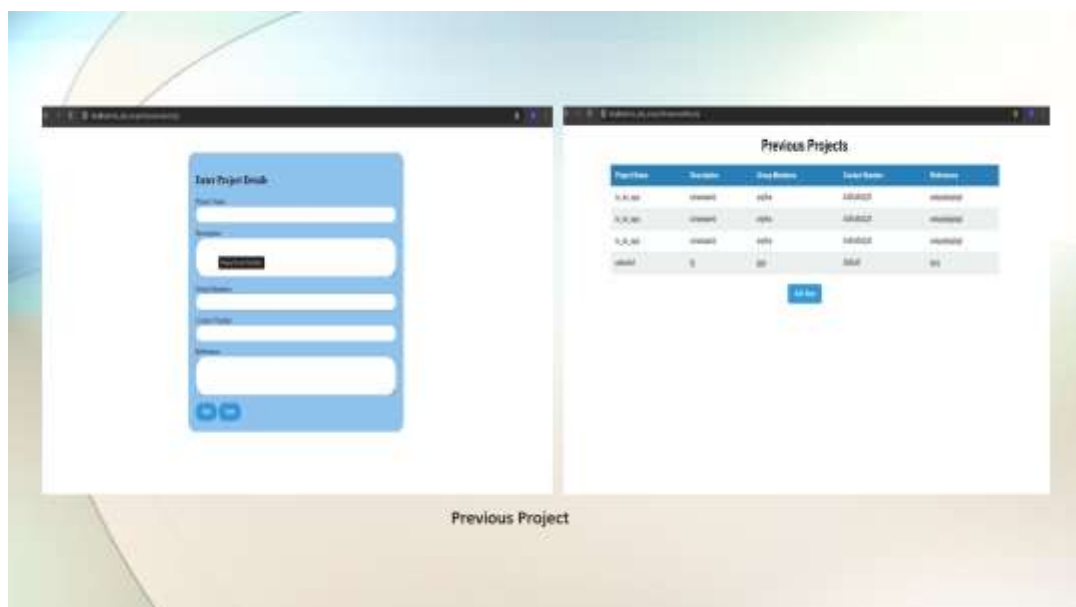
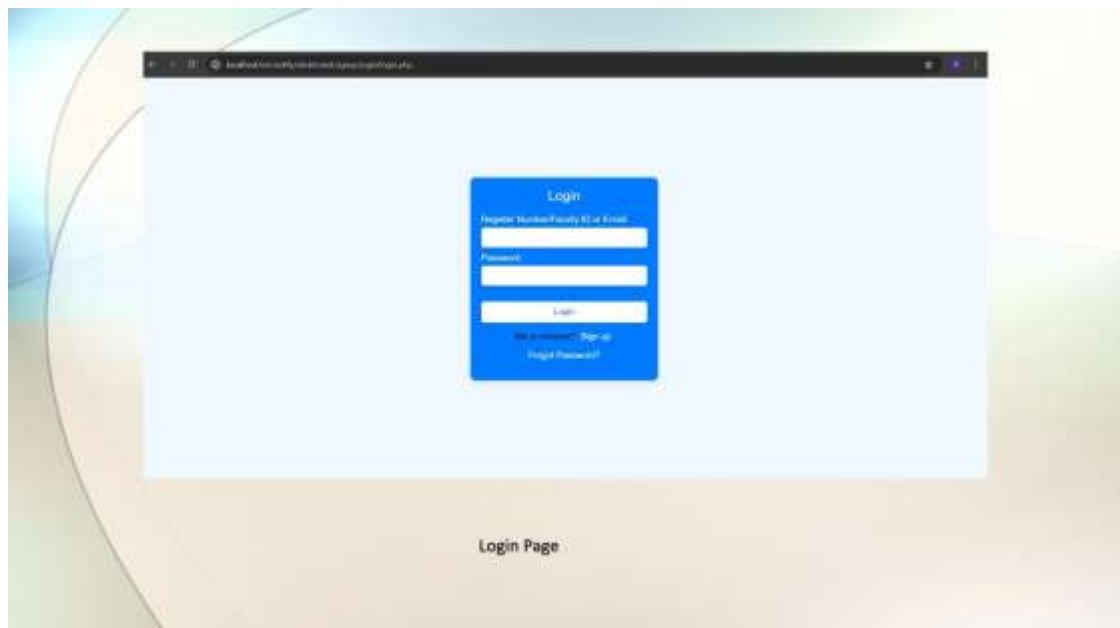
Deployment:

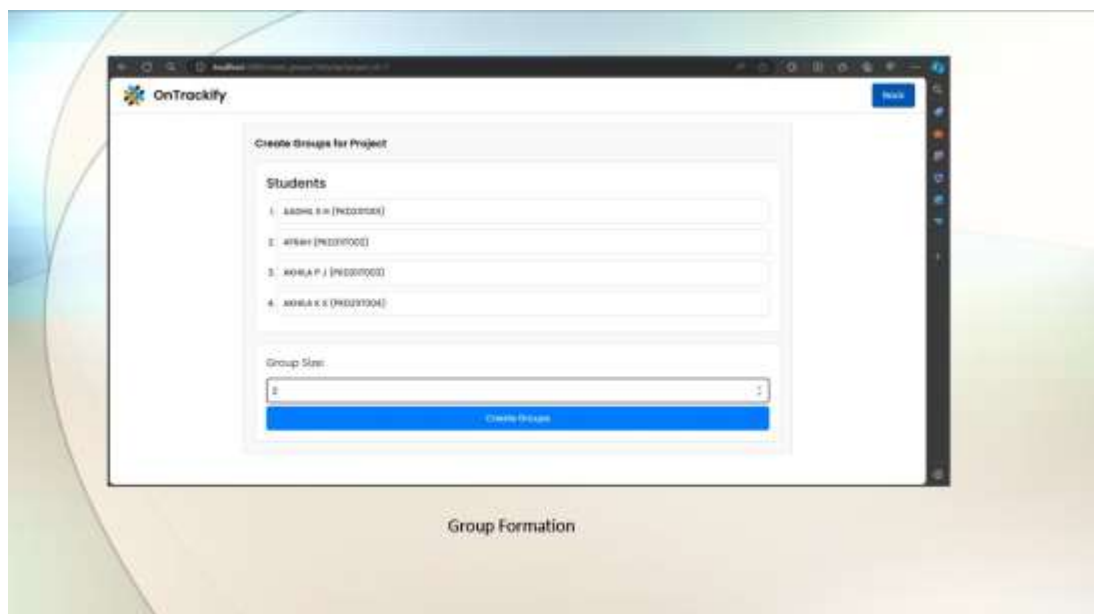
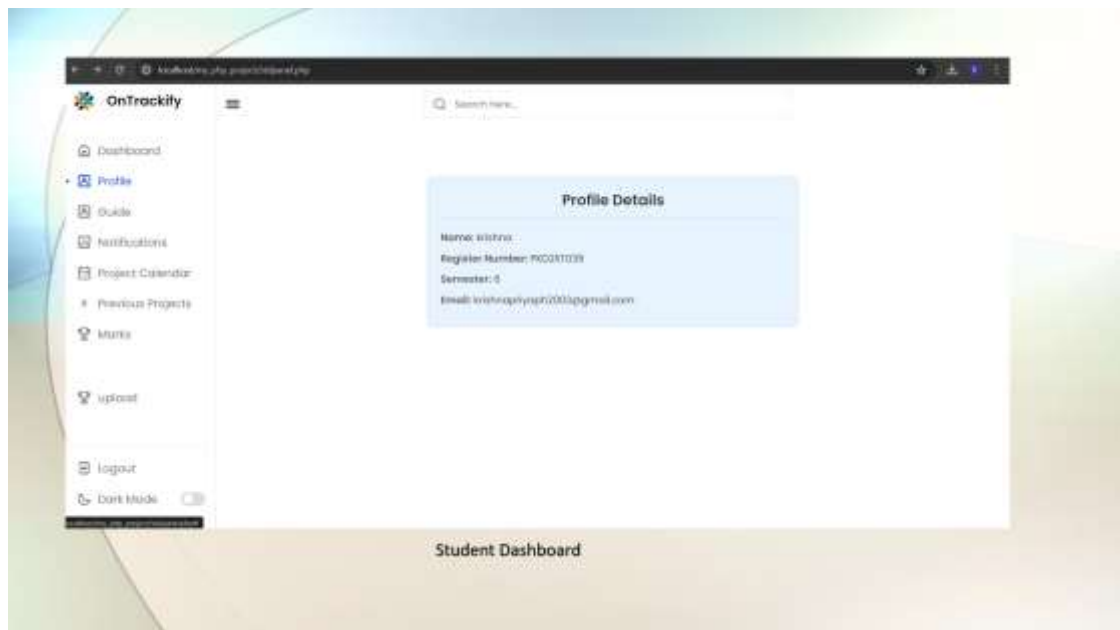
- Deploy OnTrackify on a web server using XAMPP.

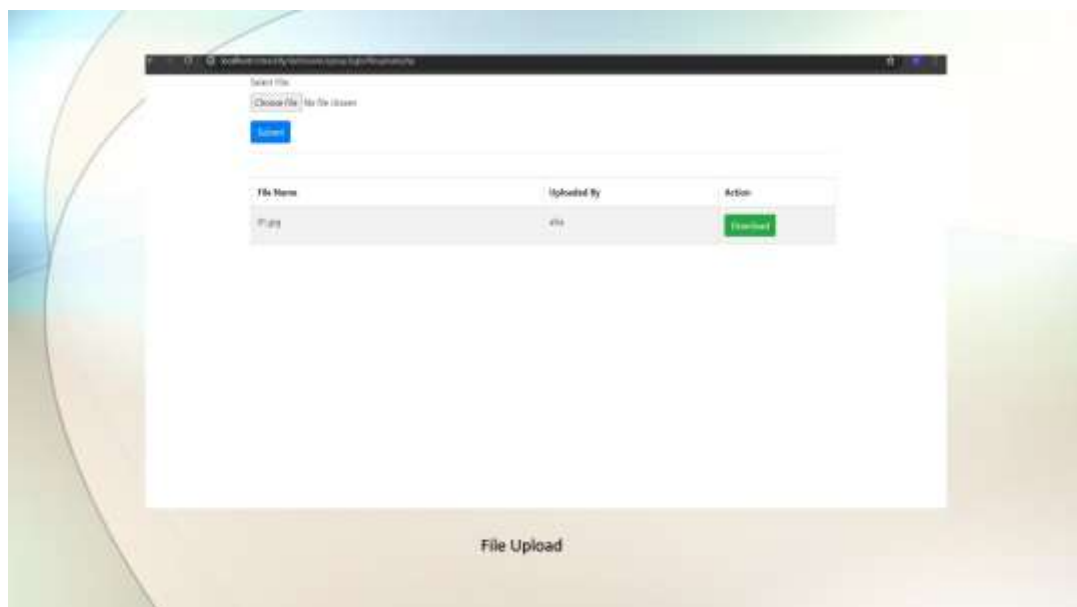
RESULT

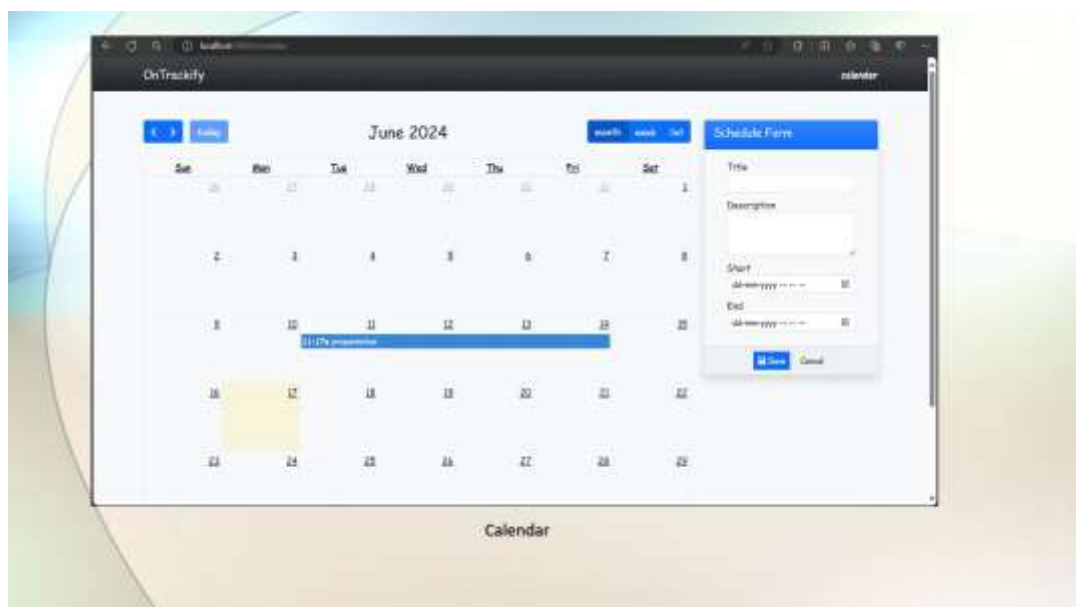


Home Page









OnTrackify

Evaluate Student

Project Name: PROJECT
 Student: RAJESH K R
 Email: aashith@gmail.com
 Register ID: PG2010100
 Group: IT-1001 Assigned

Attendance (10 Marks)	10
Quiz Marks (10 Marks)	5
Project Report (10 Marks)	10
Reviewer 1 Total Marks	15
Reviewer 2 Total Marks	15
Average Reviewer Marks	15
Final CE Mark (Total: 75)	55.00

Save Cancel

Evaluation form

Figure 8.1: Enter Caption

CONCLUSION

- Successfully developed and implemented OnTrackify, a project monitoring website for colleges.
- Provides features for:
 - Document sharing
 - Progress tracking
 - Communication
 - Notifications
 - Group management
 - Evaluation
- User-friendly interface and easy navigation.
- Dynamic calendar and real-time feedback ensure projects stay on schedule.
- Improved project management and communication within the college.
- Supports students and faculty in achieving project goals.
- Demonstrates the value of technology in enhancing academic project management.

REFERENCES

1. **TRELLO** : <https://trello.com/>
2. **JIRA** : www.atlassian.com/jira
3. **PROJECT MANAGEMENT SYSTEM**
SOWMIYA S ,YASWANTHAN S R 2023 INSTITUTE OF SCIENCE AND TECHNOLOGY
| RESEARCH GATE |year:2023
4. **STUDENT PROJECT MANAGEMENT SYSTEM(SPMS)**
Dr. Nisha Soms;2021 International Journal of All Research Education and Scientific Methods(UARES M)
|RESEARCH GATE |year:2023
5. **PROJECT MANAGEMENT SYSTEM FOR GRADUATING STUDENT PROGRESS MONITORING**
K Thanuja , P Sakethram Reddy ,P Sharuk Khan ,N Gunavardhan Reddy 2023 School of Computer Science and Engineering,
REVA University
| RESEARCH GATE |year:2023
6. **The Development of a Final Year Project Management System for Information Technology Programmes**
Chun-Hang Leung; Chung-Lun Lai-Tsun-Kit Yuan Wai-Man Pang, Jeff K.T. Tang-Wai-Shing Ho, Tak-Lam Wong
2015 Department of Computer Science, Caritas Institute of Higher Education, Hong Kong SAR, China
|RESEARCH GATE |year:2015

THANK YOU

APPENDIX-B

B.1 CODE SNIPPETS

Create_group.php

```
<?php
session_start();

// Redirect if user is not logged in as coordinator
if (!isset($_SESSION['user_id']) || $_SESSION['role'] != 'coordinator') {
    header("Location: login.php");
    exit();
}

include 'config.php';

$project_id = $_GET['project_id'];

// Fetch students who have joined the project
$query = "
SELECT u.id, u.name, u.register_or_faculty_id
FROM users u
JOIN project_members pm ON u.id = pm.student_id
WHERE pm.project_id = ? AND u.role = 'student'";
$stmt = $conn->prepare($query);
$stmt->bind_param("i", $project_id);
$stmt->execute();
$result = $stmt->get_result();
$students = $result->fetch_all(MYSQLI_ASSOC);
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Create Groups</title>
    <link rel="stylesheet" href="stdstyle.css">
    <style>
        body {
            background-color: white;
            font-family: Arial, sans-serif;
            margin: 0;
            padding: 0;
        }
        .header {
            background-color: white;
            color: black;
            padding: 10px 20px;
            border-bottom: 1px solid #ccc;
            display: flex;
            justify-content: space-between;
```

```
        align-items: center;
    }

    .logo {
        display: flex;
        align-items: center;
    }

    .logo img {
        width: 50px; /* Adjust as per your logo size */
        height: auto;
        margin-right: 10px;
    }

    .logo span {
        font-size: 1.5rem; /* Adjust as needed */
        font-weight: bold;
    }

    .back-button {
        background-color: #0056b3;
        color: white;
        border: none;
        padding: 10px 20px;
        border-radius: 5px;
        cursor: pointer;
        text-decoration: none;
    }

    .back-button:hover {
        background-color: #00408a;
    }

    .content {
        max-width: 1000px;
        margin: 20px auto;
        padding: 20px;
        border: 1px solid #ccc;
        border-radius: 3px;
        background-color: #f9f9f9;
    }

    .student-list {
        margin-top: 20px;
        padding: 20px;
        border: 1px solid #ccc;
        border-radius: 3px;
        background-color: white;
    }

    .student-list h3 {
        font-size: 1.5rem;
        margin-bottom: 10px;
    }
```

```

        color: #333;
    }

    .student-item {
        margin-bottom: 10px;
        padding: 10px;
        border: 1px solid #ccc;
        border-radius: 5px;
        background-color: white;
    }

    .group-form {
        margin-top: 20px;
        padding: 20px;
        border: 1px solid #ccc;
        border-radius: 5px;
        background-color: white;
    }

    .group-form label {
        display: block;
        font-size: 1.2rem;
        margin-bottom: 10px;
        color: #333;
    }

    .group-form input[type="number"],
    .group-form button {
        width: 100%;
        padding: 10px;
        margin-top: 5px;
        border: 1px solid #ccc;
        border-radius: 5px;
        font-size: 1rem;
        box-sizing: border-box;
    }

    .group-form button {
        background-color: #007bff;
        color: white;
        border: none;
        cursor: pointer;
        transition: background-color 0.3s ease;
    }

    .group-form button:hover {
        background-color: #0056b3;
    }

    ol {
        padding-left: 20px;
    }
</style>

```

```

</head>
<body>
    <div class="header">
        <div class="logo">
            
            <span>OnTrackify</span>
        </div>
        <a href="create_groups1.php" class="back-button">Back</a>
    </div>

    <div class="content">
        <h3>Create Groups for Project</h3>
        <div class="student-list">
            <h3>Students</h3>
            <ol>
                <?php foreach ($students as $student): ?>
                    <li class="student-item">
                        <?= htmlspecialchars($student['name'], ENT_QUOTES,
'UTF-8') ?> (<?= htmlspecialchars($student['register_or_faculty_id'],
ENT_QUOTES, 'UTF-8') ?>)
                    </li>
                <?php endforeach; ?>
            </ol>
        </div>
        <div class="group-form">
            <form action="process_create_groups.php" method="POST">
                <input type="hidden" name="project_id" value="<? =
htmlspecialchars($project_id, ENT_QUOTES, 'UTF-8') ?>">
                <label for="group_size">Group Size:</label>
                <input type="number" name="group_size" min="1" required>
                <button type="submit">Create Groups</button>
            </form>
        </div>
    </div>
</body>
</html>

```

Process create group.php

```

<?php
session_start();

// Redirect if user is not logged in as coordinator
if (!isset($_SESSION['user_id']) || $_SESSION['role'] != 'coordinator') {
    header("Location: login.php");
    exit();
}

include 'config.php';

// Function to delete existing groups and their members for a project
function deleteExistingGroups($conn, $project_id) {
    // Begin transaction for atomicity
    $conn->begin_transaction();

```

```

        // Delete existing group members
        $query_delete_members = "DELETE gm FROM group_members gm JOIN groups g ON
gm.group_id = g.group_id WHERE g.project_id = ?";
        $stmt_delete_members = $conn->prepare($query_delete_members);
        $stmt_delete_members->bind_param("i", $project_id);
        $stmt_delete_members->execute();

        // Delete existing groups
        $query_delete_groups = "DELETE FROM groups WHERE project_id = ?";
        $stmt_delete_groups = $conn->prepare($query_delete_groups);
        $stmt_delete_groups->bind_param("i", $project_id);
        $stmt_delete_groups->execute();

        // Commit transaction
        $conn->commit();
    }

    // Function to create groups
    function createGroups($conn, $project_id, $group_size) {
        // Initialize an array to store group information
        $group_info = [];

        // Calculate number of groups needed
        $query = "SELECT COUNT(*) AS total_students FROM project_members WHERE
project_id = ?";
        $stmt = $conn->prepare($query);
        $stmt->bind_param("i", $project_id);
        $stmt->execute();
        $result = $stmt->get_result();
        $row = $result->fetch_assoc();
        $total_students = $row['total_students'];
        $num_groups = ceil($total_students / $group_size);

        // Fetch students for the project
        $query_students = "
            SELECT u.id, u.name
            FROM users u
            JOIN project_members pm ON u.id = pm.student_id
            WHERE pm.project_id = ? AND u.role = 'student'
        ";
        $stmt_students = $conn->prepare($query_students);
        $stmt_students->bind_param("i", $project_id);
        $stmt_students->execute();
        $result_students = $stmt_students->get_result();
        $students = $result_students->fetch_all(MYSQLI_ASSOC);

        // Shuffle students randomly
        shuffle($students);

        // Initialize group index and group members array
        $group_index = 0;
        $group_members = [];
    }

```

```

// Begin transaction for atomicity
$conn->begin_transaction();

// Loop through shuffled students to create groups
foreach ($students as $student) {
    // Add student to current group members array
    $group_members[] = $student;

    // Check if current group members array size equals group size
    if (count($group_members) == $group_size || count($group_members) +
$group_index * $group_size >= $total_students) {
        // Insert group into database
        $query_insert_group = "
            INSERT INTO groups (coordinator_id, project_id, group_size)
            VALUES (?, ?, ?)
        ";
        $stmt_insert_group = $conn->prepare($query_insert_group);
        $stmt_insert_group->bind_param("iii", $_SESSION['user_id'],
$project_id, $group_size);
        $stmt_insert_group->execute();
        $group_id = $stmt_insert_group->insert_id;

        // Insert group members into database
        foreach ($group_members as $member) {
            $student_id = $member['id'];
            $query_insert_member = "
                INSERT INTO group_members (group_id, student_id)
                VALUES (?, ?)
            ";
            $stmt_insert_member = $conn->prepare($query_insert_member);
            $stmt_insert_member->bind_param("ii", $group_id,
$student_id);
            $stmt_insert_member->execute();
        }

        // Reset group members array and move to next group
        $group_members = [];
        $group_index++;
    }
}

// Commit transaction
$conn->commit();

// Retrieve created group information
$query_fetch_groups = "
    SELECT g.group_id, g.group_size, GROUP_CONCAT(u.name SEPARATOR ', ')
    AS group_members
    FROM groups g
    JOIN group_members gm ON g.group_id = gm.group_id
    JOIN users u ON gm.student_id = u.id
    WHERE g.project_id = ? AND g.created_at >= NOW() - INTERVAL 1 HOUR

```

```

        GROUP BY g.group_id
    ";
    $stmt_fetch_groups = $conn->prepare($query_fetch_groups);
    $stmt_fetch_groups->bind_param("i", $project_id);
    $stmt_fetch_groups->execute();
    $result_groups = $stmt_fetch_groups->get_result();
    $group_info = $result_groups->fetch_all(MYSQLI_ASSOC);

    return $group_info;
}

// Get project_id and group_size from POST data
$project_id = $_POST['project_id'];
$group_size = $_POST['group_size'];

// Delete existing groups and members for the project
try {
    deleteExistingGroups($conn, $project_id);
} catch (Exception $e) {
    // Handle any exceptions or errors in deletion
    error_log("Error deleting existing groups: " . $e->getMessage());
    echo "Error deleting existing groups: " . $e->getMessage();
    exit();
}

// Attempt to create groups
$group_info = [];
try {
    $group_info = createGroups($conn, $project_id, $group_size);
} catch (Exception $e) {
    // Handle any exceptions or errors in group creation
    error_log("Error creating groups: " . $e->getMessage());
    echo "Error creating groups: " . $e->getMessage();
    exit();
}

// Redirect to success.php to show newly created groups
header("Location: success1.php?project_id=" . $project_id);
exit();
?>

```

Success group.php

```

<?php
session_start();

// Redirect if user is not logged in as coordinator
if (!isset($_SESSION['user_id']) || $_SESSION['role'] != 'coordinator') {
    header("Location: login.php");
    exit();
}

include 'config.php';

```

```

// Get project_id from GET data
$project_id = $_GET['project_id'];
?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Success</title>
    <link rel="stylesheet" href="stdstyle.css">
    <style>
        body {
            background-color: #ffffff;
            font-family: Arial, sans-serif;
            line-height: 1.6;
            text-align: center;
        }

        .container {
            max-width: 800px;
            margin: 0 auto;
            padding: 20px;
        }

        .success-message {
            margin-top: 50px;
            background-color: #d4edda;
            color: #155724;
            border: 1px solid #c3e6cb;
            padding: 15px;
            border-radius: 5px;
        }

        .back-to-dashboard,
        .view-groups {
            margin-top: 20px;
            display: inline-block;
            padding: 10px 20px;
            background-color: #007bff;
            color: #fff;
            text-decoration: none;
            border-radius: 5px;
        }

        .back-to-dashboard:hover,
        .view-groups:hover {
            background-color: #0056b3;
        }
    </style>
</head>
<body>
    <div class="container">
        <h1>Groups Successfully Finalized</h1>

```



```

        <div class="success-message">
            <p>Your groups have been successfully finalized.</p>
        </div>
        <a href="coordinator_panel.php" class="back-to-dashboard">Back to
Dashboard</a>
        <a href="view_groups.php?project_id=<?= htmlspecialchars($project_id,
ENT_QUOTES, 'UTF-8') ?>" class="view-groups">View Groups</a>
    </div>
</body>
</html>

```

View group.php

```

<?php
session_start();

// Redirect if user is not logged in as coordinator
if (!isset($_SESSION['user_id']) || $_SESSION['role'] != 'coordinator') {
    header("Location: login.php");
    exit();
}

include 'config.php';

// Function to fetch group information created by the current coordinator
function fetchGroups($conn) {
    $query = "
        SELECT g.group_id, GROUP_CONCAT(u.name SEPARATOR ', ') AS
group_members
        FROM groups g
        JOIN group_members gm ON g.group_id = gm.group_id
        JOIN users u ON gm.student_id = u.id
        WHERE g.coordinator_id = ?
        GROUP BY g.group_id
    ";
    $stmt = $conn->prepare($query);
    $stmt->bind_param("i", $_SESSION['user_id']);
    $stmt->execute();
    $result = $stmt->get_result();
    return $result->fetch_all(MYSQLI_ASSOC);
}

// Handle notify all action
if (isset($_POST['notify_all'])) {
    $query = "
        SELECT gm.student_id
        FROM groups g
        JOIN group_members gm ON g.group_id = gm.group_id
        WHERE g.coordinator_id = ?
    ";
    $stmt = $conn->prepare($query);
    $stmt->bind_param("i", $_SESSION['user_id']);
    $stmt->execute();
    $result = $stmt->get_result();
}

```

```

$students = $result->fetch_all(MYSQLI_ASSOC);

$conn->begin_transaction();

foreach ($students as $student) {
    $student_id = $student['student_id'];
    $message = 'You have a new notification from your coordinator.';

    $query_insert_notification = "
        INSERT INTO notificationsgroup (student_id, message, is_read,
created_at)
        VALUES (?, ?, 0, NOW())
    ";
    $stmt_insert_notification = $conn-
>prepare($query_insert_notification);
    $stmt_insert_notification->bind_param("is", $student_id, $message);
    $stmt_insert_notification->execute();
}

$conn->commit();
$notification_success = true;
}

// Fetch group information for display
$group_info = fetchGroups($conn);

?>
<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta name="viewport" content="width=device-width, initial-scale=1.0">
    <title>Coordinator View Groups</title>
    <link rel="stylesheet" href="stdstyle.css">
    <style>
        /* Add your custom styles here */
        body {
            background-color: #ffffff; /* White background */
            font-family: Arial, sans-serif;
            line-height: 1.6;
        }

        .container {
            max-width: 800px;
            margin: 0 auto;
            padding: 20px;
            text-align: center;
        }

        .group-info {
            margin-top: 20px;
        }
    </style>

```

```

.group {
    margin-bottom: 20px;
    padding: 10px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

.group h3 {
    margin-bottom: 10px;
}

.group-members {
    list-style-type: none;
    padding: 0;
}

.group-members li {
    margin-bottom: 5px;
}

.action-btn {
    margin-top: 20px;
}

.notify-success {
    margin-top: 20px;
    color: green;
}

.back-to-dashboard {
    margin-top: 20px;
    display: inline-block;
    padding: 10px 20px;
    background-color: #007bff; /* Bootstrap primary button color */
    color: #fff;
    text-decoration: none;
    border-radius: 5px;
}

.back-to-dashboard:hover {
    background-color: #0056b3; /* Darker shade of primary color on
hover */
}
</style>
</head>
<body>
    <div class="container">
        <?php if (isset($notification_success) && $notification_success): ?>
            <h1>Notifications Sent Successfully</h1>
            <p class="notify-success">Notifications sent to all group members
successfully!</p>
            <a href="coordinator_panel.php" class="back-to-dashboard">Back to
Dashboard</a>

```

```

        <?php else: ?>
            <h1>Groups Created by Coordinator</h1>
            <div class="group-info">
                <?php foreach ($group_info as $group): ?>
                    <div class="group">
                        <h3>Group ID: <?=
htmlspecialchars($group['group_id'], ENT_QUOTES, 'UTF-8') ?></h3>
                        <ul class="group-members">
                            <?php $members = explode(',',
$group['group_members']); ?>
                            <?php foreach ($members as $member): ?>
                                <li><?= htmlspecialchars($member, ENT_QUOTES,
'UTF-8') ?></li>
                            <?php endforeach; ?>
                        </ul>
                    </div>
                <?php endforeach; ?>
            </div>

            <!-- Notify All Button -->
            <form action="<?= $_SERVER['PHP_SELF'] ?>" method="POST"
class="action-btn">
                <button type="submit" name="notify_all">Notify All</button>
            </form>
        <?php endif; ?>
    </div>
</body>
</html>

```

Fileupload.php

```

<?php
session_start(); // Start session for user authentication

// Database connection
$conn = new PDO('mysql:host=localhost;dbname=role_management', 'root', '');

if(isset($_POST['submit'])) {
    $name = $_FILES['file']['name'];
    $fname = date("YmdHis") . '_' . $name;
    $temp = $_FILES['file']['tmp_name'];

    // Check if user is logged in and has a valid session
    if(isset($_SESSION['user_id']) && !empty($_SESSION['user_id'])) {
        $user_id = $_SESSION['user_id'];
    } else {
        die("User session not found. Please log in.");
    }

    // Fetch the group_id for the logged-in user
    $stmt = $conn->prepare("SELECT group_id FROM group_members WHERE
student_id = :student_id");
    $stmt->bindParam(':student_id', $user_id);
    $stmt->execute();
}

```

```

$result = $stmt->fetch(PDO::FETCH_ASSOC);

if ($result) {
    $group_id = $result['group_id'];
} else {
    die("Group ID not found for the user. Please check the group_members
table.");
}

// Move uploaded file to upload directory
$upload_directory = "uploads/"; // Directory where files will be uploaded
$move = move_uploaded_file($temp, $upload_directory . $fname);

if ($move) {
    // Insert uploaded file details into database with group ID
    $query = $conn->prepare("INSERT INTO upload (name, fname, student_id,
group_id) VALUES (:name, :fname, :student_id, :group_id)");
    $query->bindParam(':name', $name);
    $query->bindParam(':fname', $fname);
    $query->bindParam(':student_id', $user_id);
    $query->bindParam(':group_id', $group_id);

    if ($query->execute()) {
        // Redirect after successful upload
        header("Location: fileupload.php");
        exit;
    } else {
        die("An error occurred: Unable to execute query. Please try again
later.");
    }
} else {
    die("File upload failed. Please try again.");
}
}
?>

```

```

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>File Upload</title>
    <!-- Include Bootstrap CSS -->
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.min.cs
s">
</head>
<body>

    <div class="container">
        <!-- Upload form -->
        <form enctype="multipart/form-data" action="" method="post">
            <div class="form-group">
                <label for="file">Select File:</label >

```

```

        <input type="file" class="form-control-file" id="file"
name="file">
    </div>
    <button type="submit" name="submit" class="btn btn-
primary">Submit</button>
</form>

<hr>
<br>

<!-- Display uploaded files for the logged-in user/group -->
<table class="table table-striped table-bordered">
    <thead>
        <tr>
            <th width="50%" align="center">File Name</th>
            <th width="30%" align="center">Uploaded By</th>
            <th align="center">Action</th>
        </tr>
    </thead>
    <tbody>
        <?php
        // Fetch files uploaded by the logged-in user or their group
        $query_str = "
        AS student_name
        FROM upload
        JOIN users ON upload.student_id = users.id
        WHERE upload.student_id = :student_id OR upload.group_id
        = :group_id
        ORDER BY upload.id DESC";
        $query = $conn->prepare($query_str);
        $query->bindParam(':student_id', $_SESSION['user_id']);
        $query->bindParam(':group_id', $group_id);
        $query->execute();

        while ($row = $query->fetch(PDO::FETCH_ASSOC)) {
            $file_name = $row['file_name'];
            $fname = $row['fname'];
            $student_name = $row['student_name'];
            ?>
            <tr>
                <td><?php echo htmlspecialchars($file_name); ?></td>
                <td><?php echo htmlspecialchars($student_name); ?></td>
                <td>
                    <a href="download.php?filename=<?php echo
urlencode($file_name); ?>&f=<?php echo urlencode($fname); ?>" class="btn btn-
success">Download</a>
                </td>
            </tr>
        <?php } ?>
    </tbody>
</table>
</div>

```

```
</body>
</html>
```

Composenotifications.php

```
<?php
session_start();

// Check if user is not logged in, redirect to login page
if (!isset($_SESSION['user_id'])) {
    header("Location: login.php");
    exit();
}

include 'config.php';

$user_id = $_SESSION['user_id'];
$role = $_SESSION['role'];

// Handle form submission
if ($_SERVER["REQUEST_METHOD"] == "POST" && isset($_POST['send'])) {
    $message = $_POST['message'];
    $recipient = $_POST['recipient'];
    $coordinator_id = $user_id;
    $date_time = date('Y-m-d H:i:s');

    $sql = "INSERT INTO notifications (message, recipient, coordinator_id,
date_time) VALUES (?, ?, ?, ?)";
    $stmt = $conn->prepare($sql);
    $stmt->bind_param("ssis", $message, $recipient, $coordinator_id,
$date_time);
    $stmt->execute();
    $stmt->close();
    echo "<script>alert('Notification sent');
window.location.href='compose_notification.php';</script>";
}

// Fetch notifications
$sql = "SELECT * FROM notifications WHERE coordinator_id = ? ORDER BY
date_time DESC";
$stmt = $conn->prepare($sql);
$stmt->bind_param("i", $user_id);
$stmt->execute();
$result = $stmt->get_result();
$notifications = $result->fetch_all(MYSQLI_ASSOC);
$stmt->close();
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <meta http-equiv="X-UA-Compatible" content="IE=edge">
```

```
<meta name="viewport" content="width=device-width, initial-scale=1.0">
<title>Compose Notification</title>
<link rel="stylesheet" href="stdstyle.css">
<style>
  /* Ensure high specificity to override existing styles */
  body {
    font-family: Arial, sans-serif;
    background-color: #f0f0f0;
    margin: 0;
    padding: 0;
  }

  .compose-container {
    max-width: 600px;
    margin: 50px auto;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
    background-color: #fff;
  }

  .compose-container h2 {
    text-align: center;
    margin-bottom: 20px;
  }

  .compose-container form {
    display: flex;
    flex-direction: column;
  }

  .compose-container label {
    margin-bottom: 5px;
  }

  .compose-container input,
  .compose-container select,
  .compose-container textarea {
    padding: 10px;
    margin-bottom: 15px;
    border: 1px solid #ccc;
    border-radius: 5px;
  }

  .compose-container button {
    padding: 10px;
    background-color: #3498db;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
  }
}
```



```
.compose-container button:hover {
    background-color: #2980b9;
}

.notification-list {
    max-width: 600px;
    margin: 20px auto;
    background-color: #fff;
    padding: 20px;
    border: 1px solid #ccc;
    border-radius: 5px;
}

.notification-item {
    border: 1px solid #ccc;
    padding: 10px;
    margin-bottom: 10px;
    border-radius: 5px;
    background-color: #f9f9f9; /* Light gray background color */
    display: flex;
    justify-content: space-between;
    align-items: center;
}

.notification-item span {
    flex-grow: 1;
}

.notification-item button {
    margin-left: 10px;
    background-color: #e74c3c;
    color: #fff;
    border: none;
    border-radius: 5px;
    cursor: pointer;
    transition: background-color 0.3s ease;
}

.notification-item button.edit {
    background-color: #3498db;
}

.notification-item button:hover {
    background-color: #c0392b;
}

.notification-item button.edit:hover {
    background-color: #2980b9;
}

.back-button {
    display: inline-block;
```

```

        padding: 10px 20px;
        background-color: #007bff;
        color: #fff;
        text-decoration: none;
        border-radius: 5px;
        margin-top: 20px;
        transition: background-color 0.3s ease;
    }

    .back-button:hover {
        background-color: #0056b3;
    }
</style>
</head>
<body>
    <div class="compose-container">
        <h2>Compose Notification</h2>
        <form action="compose_notification.php" method="POST">
            <label for="recipient">Recipient:</label>
            <select name="recipient" id="recipient" required>
                <option value="student">Student</option>
                <option value="guide">Guide</option>
            </select>

            <label for="message">Message:</label>
            <textarea name="message" id="message" rows="5"
required></textarea>

            <button type="submit" name="send">Send</button>
        </form>
    </div>

    <div class="notification-list">
        <h2>Sent Notifications</h2>
        <?php foreach ($notifications as $notification): ?>
            <div class="notification-item" data-id="<?= $notification['id']
?>">
                <span><?= htmlspecialchars($notification['message']) ?> (<?=
htmlspecialchars($notification['recipient']) ?>)</span>
                <button class="edit">Edit</button>
                <button class="delete">Delete</button>
            </div>
        <?php endforeach; ?>
    </div>

    <a href="coordinator_panel.php" class="back-button">Back to Coordinator
Panel</a>

    <script>
        document.addEventListener("DOMContentLoaded", function() {
            const notificationItems =
document.querySelectorAll('.notification-item');
            notificationItems.forEach(item => {

```

```

const editButton = item.querySelector('.edit');
const deleteButton = item.querySelector('.delete');

editButton.addEventListener('click', () => {
    const id = item.dataset.id;
    window.location.href = edit_notification.php?id=${id};
});

deleteButton.addEventListener('click', () => {
    const id = item.dataset.id;
    if (confirm('Are you sure you want to delete this
notification?')) {
        fetch(delete_notification.php?id=${id}, {
            method: 'GET'
        }).then(response => response.text()).then(data => {
            if (data === 'success') {
                item.remove();
            } else {
                alert('Error deleting notification');
            }
        });
    }
});
});
});
</script>
</body>
</html>

```

evaluate_student.php

```

<?php
session_start();
include 'config.php';

// Ensure only coordinators can access this page
if ($_SESSION['role'] != 'coordinator') {
    header("Location: login.php");
    exit();
}

// Retrieve student_id and project_id from GET parameters
if (isset($_GET['student_id']) && isset($_GET['project_id'])) {
    $student_id = $_GET['student_id'];
    $project_id = $_GET['project_id'];
} else {
    // Redirect to an appropriate page or handle the error scenario
    header("Location: display_project.php");
    exit();
}

// Fetch student details
$student_query = "SELECT u.id, u.name, u.email, u.register_or_faculty_id,
sg.group_id

```

```

        FROM users u
        LEFT JOIN student_groups sg ON u.id = sg.student_id
        WHERE u.id = ?";
$stmt_student = $conn->prepare($student_query);
$stmt_student->bind_param("i", $student_id);
$stmt_student->execute();
$student_result = $stmt_student->get_result();
$student = $student_result->fetch_assoc();

// Fetch project details
$project_query = "SELECT * FROM projects WHERE id = ?";
$stmt_project = $conn->prepare($project_query);
$stmt_project->bind_param("i", $project_id);
$stmt_project->execute();
$project_result = $stmt_project->get_result();
$project = $project_result->fetch_assoc();

// Initialize variables for marks
$attendance = $guide_marks = $project_report = $review1_total_marks =
$review2_total_marks = $average_review_marks = $final_cie_mark = 0;
$success_message = $error_message = '';

// Check if form is submitted
if ($_SERVER["REQUEST_METHOD"] == "POST") {
    $attendance = $_POST['attendance'];
    $project_report = $_POST['project_report'];
    $review1_total_marks = $_POST['review1_total_marks'];
    $review2_total_marks = $_POST['review2_total_marks'];

    // Calculate average review marks
    $average_review_marks = ($review1_total_marks + $review2_total_marks) /
2;

    // Calculate final_cie_mark as the total of average review marks, guide
marks, attendance, and project report
    $final_cie_mark = $average_review_marks + $guide_marks + $attendance +
$project_report;
    $final_cie_mark = min($final_cie_mark, 75); // Ensure it totals out of 75

    // Insert or update student marks into student_marks table
    $sql = "INSERT INTO student_marks (student_id, project_id, attendance,
project_report, review1_total_marks, review2_total_marks,
average_review_marks, final_cie_mark)
VALUES (?, ?, ?, ?, ?, ?, ?, ?)
ON DUPLICATE KEY UPDATE
attendance = VALUES(attendance),
project_report = VALUES(project_report),
review1_total_marks = VALUES(review1_total_marks),
review2_total_marks = VALUES(review2_total_marks),
average_review_marks = VALUES(average_review_marks),
final_cie_mark = VALUES(final_cie_mark)";

    $stmt = $conn->prepare($sql);

```

```

$stmt->bind_param("iiiiidd",
                  $student_id,
                  $project_id,
                  $attendance,
                  $project_report,
                  $review1_total_marks,
                  $review2_total_marks,
                  $average_review_marks,
                  $final_cie_mark);

if ($stmt->execute()) {
    // Success message
    $success_message = "Student evaluation updated successfully.";
} else {
    // Error message
    $error_message = "Error updating student evaluation: " . $conn-
>error;
}

$stmt->close();
} else {
    // Fetch existing marks if available
    $marks_query = "SELECT * FROM student_marks WHERE student_id = ? AND
project_id = ?";
    $stmt_marks = $conn->prepare($marks_query);
    $stmt_marks->bind_param("ii", $student_id, $project_id);
    $stmt_marks->execute();
    $marks_result = $stmt_marks->get_result();

    if ($marks_result->num_rows > 0) {
        // Marks already exist, fetch and display them
        $marks_row = $marks_result->fetch_assoc();
        $attendance = $marks_row['attendance'];
        $project_report = $marks_row['project_report'];
        $review1_total_marks = $marks_row['review1_total_marks'];
        $review2_total_marks = $marks_row['review2_total_marks'];

        // Calculate average review marks
        $average_review_marks = ($review1_total_marks + $review2_total_marks)
/ 2;

        // Calculate final_cie_mark
        $final_cie_mark = $marks_row['final_cie_mark'];
    }

    // Fetch guide marks separately
    $guide_marks_query = "SELECT guide_marks FROM guide_marks WHERE
student_id = ? AND project_id = ?";
    $stmt_guide_marks = $conn->prepare($guide_marks_query);
    $stmt_guide_marks->bind_param("ii", $student_id, $project_id);
    $stmt_guide_marks->execute();
    $guide_marks_result = $stmt_guide_marks->get_result();

```

```

if ($guide_marks_result->num_rows > 0) {
    $guide_marks_row = $guide_marks_result->fetch_assoc();
    $guide_marks = $guide_marks_row['guide_marks'];
}

$stmt_marks->close();
$stmt_guide_marks->close();
}
?>

<!DOCTYPE html>
<html lang="en">
<head>
    <meta charset="UTF-8">
    <title>Evaluate Student</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.5.2/css/bootstrap.min.cs
s">
    <style>
        .navbar-brand {
            display: flex;
            align-items: center;
        }
        .navbar-brand img {
            margin-right: 10px;
        }
        .back-button {
            position: absolute;
            top: 10px;
            right: 10px;
        }
        .card {
            margin-top: 20px;
            background-color: lightblue; /* Blue background */
            color: black; /* White text */
        }
        .form-group {
            margin-bottom: 15px;
        }
        .form-group label {
            font-weight: bold;
        }
        .form-control {
            border-radius: 0;
        }
        .btn-primary {
            border-radius: 0;
        }
        .btn-secondary {
            border-radius: 0;
        }
    </style>
</script>

```



```

        <label for="attendance" class="col-sm-4 col-form-
label">Attendance (10 Marks)</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="attendance" name="attendance" max="10" value="<?php echo $attendance;
?>">

        </div>
    </div>
    <div class="form-group row">
        <label for="guide_marks" class="col-sm-4 col-form-
label">Guide Marks (15 Marks)</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="guide_marks" name="guide_marks" max="15" value="<?php echo $guide_marks;
?>" disabled>

        </div>
    </div>
    <div class="form-group row">
        <label for="project_report" class="col-sm-4 col-form-
label">Project Report (10 Marks)</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="project_report" name="project_report" max="10" value="<?php echo
$project_report; ?>">

        </div>
    </div>
    <div class="form-group row">
        <label for="review1_total_marks" class="col-sm-4 col-
form-label">Review 1 Total Marks</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="review1_total_marks" name="review1_total_marks" max="40" value="<?php
echo $review1_total_marks; ?>">

        </div>
    </div>
    <div class="form-group row">
        <label for="review2_total_marks" class="col-sm-4 col-
form-label">Review 2 Total Marks</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="review2_total_marks" name="review2_total_marks" max="40" value="<?php
echo $review2_total_marks; ?>">

        </div>
    </div>
    <div class="form-group row">
        <label for="average_review_marks" class="col-sm-4
col-form-label">Average Review Marks</label>
        <div class="col-sm-8">
            <input type="number" class="form-control"
id="average_review_marks" name="average_review_marks" value="<?php echo
$average_review_marks; ?>" readonly>

        </div>
    </div>

```



```

        <div class="form-group row">
            <label for="final_cie_mark" class="col-sm-4 col-form-
label">Final CIE Mark (Total: 75)</label>
            <div class="col-sm-8">
                <input type="number" class="form-control"
id="final_cie_mark" name="final_cie_mark" value="<?php echo $final_cie_mark;
?>" readonly>
            </div>
        </div>
        <div class="form-group row">
            <div class="col-sm-12 text-right">
                <button type="submit" class="btn btn-
primary">Save</button>
                <a href="display_project.php" class="btn btn-
secondary">Cancel</a>
            </div>
        </div>
    </form>
    <?php if ($success_message): ?>
        <div class="alert alert-success mt-4"><?php echo
$success_message; ?></div>
    <?php elseif ($error_message): ?>
        <div class="alert alert-danger mt-4"><?php echo
$error_message; ?></div>
    <?php endif; ?>
</div>
</div>
</body>
</html>

```

SQL DDL COMMANDS

```

-- Create the role_management database if it doesn't exist
CREATE DATABASE IF NOT EXISTS role_management;

-- Switch to the role_management database
USE role_management;

-- Create the users table
CREATE TABLE IF NOT EXISTS users (
    id INT AUTO_INCREMENT PRIMARY KEY,
    name VARCHAR(100) NOT NULL,
    email VARCHAR(100) NOT NULL UNIQUE,
    register_or_faculty_id VARCHAR(50) NOT NULL UNIQUE,
    password VARCHAR(255) NOT NULL,
    role ENUM('student', 'coordinator', 'guide') NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP
);

-- Create the students table
CREATE TABLE IF NOT EXISTS students (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    department VARCHAR(100),
    semester INT,

```

```

        college_name VARCHAR(100),
        batch INT,
        FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
    );

```

```

-- Create the coordinators table
CREATE TABLE IF NOT EXISTS coordinators (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    department VARCHAR(100),
    college_name VARCHAR(100),
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);

```

```

-- Create the guides table
CREATE TABLE IF NOT EXISTS guides (
    id INT AUTO_INCREMENT PRIMARY KEY,
    user_id INT NOT NULL,
    department VARCHAR(100),
    college_name VARCHAR(100),
    FOREIGN KEY (user_id) REFERENCES users(id) ON DELETE CASCADE
);

```

```

CREATE TABLE groups (
    group_id INT(11) AUTO_INCREMENT PRIMARY KEY,
    coordinator_id INT(11) DEFAULT NULL,
    project_id INT(11) DEFAULT NULL,
    group_size INT(11) DEFAULT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    INDEX (coordinator_id),
    INDEX (project_id)
);

```

```

CREATE TABLE group_members (
    id INT(11) AUTO_INCREMENT PRIMARY KEY,
    group_id INT(11) DEFAULT NULL,
    student_id INT(11) DEFAULT NULL
);

```

```

CREATE TABLE `schedule_list` (
    `id` int(30) NOT NULL,
    `title` text NOT NULL,
    `description` text NOT NULL,
    `start_datetime` datetime NOT NULL,
    `end_datetime` datetime DEFAULT NULL
);

```

```

) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

--
-- Dumping data for table `schedule_list`

ALTER TABLE `schedule_list`
  ADD PRIMARY KEY (`id`);

ALTER TABLE `schedule_list`
  MODIFY `id` int(30) NOT NULL AUTO_INCREMENT, AUTO_INCREMENT=5;
COMMIT;


CREATE TABLE IF NOT EXISTS upload (
  id int(11) NOT NULL AUTO_INCREMENT,
  name varchar(255) NOT NULL,
  fname varchar(255) NOT NULL,
  group_id int(11) NOT NULL,
  student_id int(11) NOT NULL,
  PRIMARY KEY (id),
  FOREIGN KEY (group_id) REFERENCES groups(group_id),
  FOREIGN KEY (student_id) REFERENCES users(id)
);


-- Create notifications table
CREATE TABLE notifications (
  id INT AUTO_INCREMENT PRIMARY KEY,
  message TEXT NOT NULL,
  recipient VARCHAR(20) NOT NULL,
  coordinator_id INT NOT NULL,
  date_time DATETIME NOT NULL,
  FOREIGN KEY (coordinator_id) REFERENCES coordinators(user_id)
);


CREATE DATABASE profile_management;
USE profile_management;
CREATE TABLE profile_details (
  id INT(11) NOT NULL AUTO_INCREMENT,
  role_id INT(11) NOT NULL,
  department VARCHAR(255) NOT NULL,
  semester VARCHAR(50) NOT NULL,
  college_name VARCHAR(255) NOT NULL,
  batch VARCHAR(50) NOT NULL,
  PRIMARY KEY (id)
);


ALTER TABLE profile_details ADD COLUMN profile_picture VARCHAR(255) DEFAULT
NULL;

```

```

-- Database: `chatapp`
CREATE DATABASE chatapp;
USE chatapp;

-- Table structure for table `users`
CREATE TABLE `users` (
  `user_id` int(11) NOT NULL AUTO_INCREMENT,
  `unique_id` int(255) NOT NULL,
  `fname` varchar(255) NOT NULL,
  `lname` varchar(255) NOT NULL,
  `email` varchar(255) NOT NULL,
  `password` varchar(255) NOT NULL,
  `img` varchar(255) NOT NULL,
  `status` varchar(255) NOT NULL,
  PRIMARY KEY (`user_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- Table structure for table `groups`
CREATE TABLE `groups` (
  `group_id` int(11) NOT NULL AUTO_INCREMENT,
  `group_name` varchar(255) NOT NULL,
  `group_icon` varchar(255) NOT NULL,
  `group_description` varchar(255) NOT NULL,
  PRIMARY KEY (`group_id`)
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- Table structure for table `messages`
CREATE TABLE `messages` (
  `msg_id` int(11) NOT NULL AUTO_INCREMENT,
  `incoming_msg_id` int(255) DEFAULT NULL,
  `outgoing_msg_id` int(255) NOT NULL,
  `group_id` int(11) DEFAULT NULL,
  `msg` varchar(1000) NOT NULL,
  PRIMARY KEY (`msg_id`),
  FOREIGN KEY (`group_id`) REFERENCES `groups` (`group_id`) ON DELETE CASCADE
) ENGINE=InnoDB DEFAULT CHARSET=utf8mb4;

-- Create the student_marks table
CREATE TABLE IF NOT EXISTS student_marks (
  id INT AUTO_INCREMENT PRIMARY KEY,
  student_id INT NOT NULL,
  project_id INT NOT NULL,
  attendance INT DEFAULT 0,
  guide_marks INT DEFAULT 0,
  presentation_marks INT DEFAULT 0,
  implementation_marks INT DEFAULT 0,
  total_review_marks INT DEFAULT 0,
  guide_evaluation_marks INT DEFAULT 0,
  project_report_marks INT DEFAULT 0,
  total_cie_marks INT DEFAULT 0,
  created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
  FOREIGN KEY (student_id) REFERENCES users(id) ON DELETE CASCADE,

```

```
        FOREIGN KEY (project_id) REFERENCES projects(id) ON DELETE CASCADE
    );

CREATE TABLE guide_marks (
    id INT AUTO_INCREMENT PRIMARY KEY,
    student_id INT NOT NULL,
    project_id INT NOT NULL,
    guide_marks INT NOT NULL,
    created_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP,
    updated_at TIMESTAMP DEFAULT CURRENT_TIMESTAMP ON UPDATE
CURRENT_TIMESTAMP,
    FOREIGN KEY (student_id) REFERENCES users(id),
    FOREIGN KEY (project_id) REFERENCES projects(id)
);
```