STUDENT COLLABORATION PLATFORM

A Project Report

Submitted by

Paras Bhosale	**112103025**
Spandan Jathar	**112103054**
Abhishek Kakade	**112103063**

of

TY (Computer Engineering)

Under the guidance of

Dr. Tanuja Pattanshetti

COEP Technological University

DEPARTMENT OF COMPUTER ENGINEERING

COEP Technological University

April, 2024

DEPARTMENT OF COMPUTER ENGINEERING

COEP Technological University

CERTIFICATE

Certified that this project, titled "STUDENT COLLABORATION PLATFORM" has been successfully completed by

Paras Bhosale	**112103025**
Spandan Jathar	**112103054**
Abhishek Kakade	**112103063**

and is approved for the fulfilment of the requirements of "Software Engineering Mini Project- Stage II".

SIGNATURE

Dr. Tanuja R. Pattanshetti
Project Guide

Department of Computer Engineering

COEP Technological University,

Shivajinagar, Pune - 5.

Abstract

In today's educational landscape, effective collaboration among students is essential for achieving academic success and fostering valuable skills for the future workforce. The Student Collaboration Platform (SCP) presented in this project report is designed to streamline organizational processes and enhance communication and collaboration among students

The Student Collaboration Platform integrates key features including user authentication, group and project management, messaging capabilities, external tool integration,task management,Time table Scheduling, notifications, student profile analytics, security measures, and a feedback and recognition system. These features are aimed at improving organizational efficiency, facilitating seamless task management, and promoting effective communication and collaboration among students.

This report details the design, development, and implementation of the SCP, highlighting the importance of each feature and its contribution to enhancing student collaboration. The platform not only provides a centralized hub for managing tasks and projects but also offers insights through analytics to optimize student engagement and productivity. Additionally, the SCP ensures data security and privacy through robust authentication and access controls.

The findings and outcomes discussed in this report underscore the significance of the SCP in fostering a conducive environment for student collaboration and teamwork. Future enhancements and potential applications of the platform are also explored, demonstrating its potential impact on improving educational experiences and outcomes. The SCP represents a valuable tool for educational institutions seeking to promote collaboration and efficiency among their students in an increasingly digital and interconnected world.

Contents

List of Figures

Chapter 1

Synopsis

1.1 Project Title

STUDENT COLLABORATION PLATFORM

1.2 Internal Guide

Dr. Tanuja R. Pattanshetti

1.3 Problem Statement

The goal of this project is to design and develop a robust SCP that integrates essential features such as user authentication, group and project management tools, messaging functionalities, seamless integration with external tools, real-time notifications, student profile analytics, security measures, and a feedback and recognition system. By addressing these needs, the SCP aims to streamline task management processes, foster effective communication and collaboration, and ultimately enhance the overall educational experience for students.

1.4 Plan of Project Execution

Task	Start Date	End Date	Duration	Resources
Problem statement and Research	11-Jan-24	25-Jan-24	11	Paras,Spandan,Abhishek
Software Requirements and Specifications	26-Jan-24	07-Feb-24	9	Paras
Database Design and ER Model	08-Feb-24	15-Feb-24	6	Paras,Abhishek
Development-User Profiles	16-Feb-24	22-Feb-24	5	Paras,Spandan
Development-User Authentication	122-Feb-24	27-Feb-24	4	Paras
Development-Group Management	28-Feb-24	11-Mar-24	9	Spandan
Development-Timetable Management	12-Mar-24	18-Feb-24	5	Abhishek
Development-Task Tracking	19-Mar-24	25-Mar-24	5	Spandan,Paras
Development-Communication Tools	26-Mar-24	01-Apr-24	5	Abhishek,Paras,Spandan
Development- Additional Features	02-Apr-24	05-Apr-24	4	Paras,Spandan
Review	05-Apr-24	08-Apr-24	2	Paras,Abhishek,Spandan
Documentation and	09-Apr-24	11-Apr-24	3	Paras,Abhishek,Spandan

Chapter 2

Problem Definition and scope

2.1 Problem Definition

2.1.1 Goals and objectives

Goal and Objectives:

- Goals
  - The goal of this project is to design, develop, and implement a comprehensive Student Collaboration Platform (SCP) that enhances organizational efficiency, fosters effective communication, and promotes collaboration among students within educational institutions.
- Objectives
  - **Enhance Educational Experience:** Enhance the educational experience for students by providing a centralized platform that supports project-based learning, teamwork, and skill development in a digital environment.
  - **User-Centric Design:** Design the SCP with a focus on user experience, ensuring that it is intuitive, easy to use, and tailored to the specific needs of students.

2.1.2 Statement of scope

- **User Authentication and Access Control:** Define user roles (e.g., students, group admin, group member) with appropriate permissions.
- **Group and Project Management:** Facilitate task assignment, progress tracking, and resource allocation within projects.
- **Messaging and Communication:** Support file sharing, document collaboration, and discussion threads within groups and projects.

2.2 Software context

The development of the Student Collaboration Platform (SCP) involves a comprehensive selection of software tools, technologies, and frameworks tailored to create a robust and user-friendly solution for enhancing organizational efficiency and fostering effective communication and collaboration among students within educational institutions.The frontend development of the SCP will utilize modern web technologies including HTML, CSS, and JavaScript to craft intuitive and interactive user interfaces. On the backend, programming languages such as Python will power the server-side logic and application development, coupled with framework like Flask for building RESTful APIs to handle server operations and data management. Data will be stored and retrieved using database management systems like MySQL. Real-time communication will be facilitated through WebSocket protocols or third-party services like Socket.io or Pusher, enabling seamless messaging and notifications among platform users.

2.3 Outcome

The primary outcome of the STUDENT COLLABORATION PLATFORM project is the successful development and deployment of a comprehensive web application that empowers users to join groups, real time communication and manage task. Key components of the outcome include:

- **Improved Communication and Collaboration:** Facilitated seamless communication and collaboration among students, educators, and project teams. Enhanced teamwork, knowledge sharing, and peer-to-peer interactions within academic environments.
- **Empowered Student Engagement:** Empowerment of students through active participation in collaborative projects and activities. Increased student motivation, ownership of tasks, and accountability for project outcomes.
- **Enhanced User Experience:** The outcome results in an enhanced user experience, enabling users to make real time communication, group management, timetable scheduling.

2.4 Applications

- **Facilitation of Collaborative Learning:** Enable students to work together on projects, assignments, and group activities regardless of their physical locations. Foster peer-to-peer learning and knowledge sharing through collaborative discussions and teamwork.
- **Project Management and Task Coordination:** Streamline project management processes by providing tools for task assignment, progress tracking, and re-

source allocation. Enhance coordination among project teams and ensure accountability for individual and group tasks.

- **Communication and Engagement:** Improve communication channels between students, educators, and academic staff through real-time messaging, notifications, and announcements. Promote active engagement and participation in academic activities and extracurricular projects.
- **Resource Sharing and Collaboration:** Facilitate seamless sharing of documents, files, and resources among students and project teams. Encourage collaborative editing, brainstorming, and idea exchange within virtual workspaces.
- **Feedback and Continuous Improvement:** Gather feedback from users to iteratively improve the SCP based on usability, performance, and feature requests.

2.5 Software Resources Required

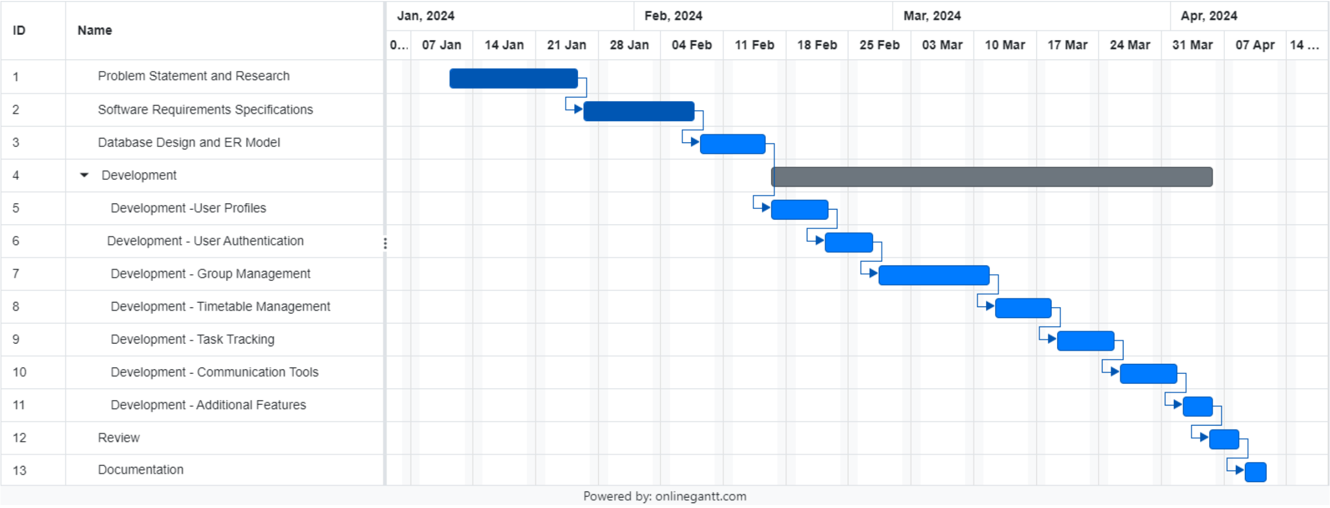
1. **Frontend Development:**HTML, CSS, JavaScript for building interactive user interfaces and client-side functionality.
2. **Backend Development:** Programming languages like Python (with frameworks such as Flask)
3. **Database management systems:** Database management systems (DBMS) like MySQL for data storage and retrieval.
4. **Real-Time Communication:** WebSocket protocols (e.g., Socket.io, Pusher) for implementing real-time messaging and notifications within the platform.
5. **Version Control Systems:** Version control software such as Git is necessary for managing the source code, tracking changes, and collaborating with members

Chapter 3

Project Plan

3.1 Project Schedule

3.1.1 Gantt Chart



ID	Name
1	Problem Statement and Research
2	Software Requirements Specifications
3	Database Design and ER Model
4	Development
5	Development -User Profiles
6	Development - User Authentication
7	Development - Group Management
8	Development - Timetable Management
9	Development - Task Tracking
10	Development - Communication Tools
11	Development - Additional Features
12	Review
13	Documentation

Chapter 4

Software requirement specification

4.1 Data Model and Description

Entity Relationship Diagram:

- **Users** ↔ *Groups* :
    - One user can be the leader of multiple groups.
    - One group has one leader.
    - One user can be a member of multiple groups.
    - One group can have multiple members.
- **Users** ↔ *Timetables* :
    - One user can create multiple events in their timetable.
    - One event belongs to one user.
- **Users** ↔ *Tasks* :
    - One user can be assigned multiple tasks.
    - One task is assigned to one user.

  - One task can belong to one group.

- **Groups** $\leftrightarrow GroupMembers$:

  - One group can have multiple members.

  - One member can belong to multiple groups.

- **Users** $\leftrightarrow Files$:

  - One user can upload multiple files.

  - One file is uploaded by one user.

  - One file can belong to one group.

- **Users** $\leftrightarrow Events$:

  - One user can create multiple events.

  - One event is created by one user.

- **Groups** $\leftrightarrow Tasks$:

  - One group can have multiple tasks.

  - One task can belong to one group.

- **Users** $\leftrightarrow PeerReviews$:

  - One user can provide multiple peer reviews.

  - One peer review is provided by one user.

  - One peer review is associated with one group.

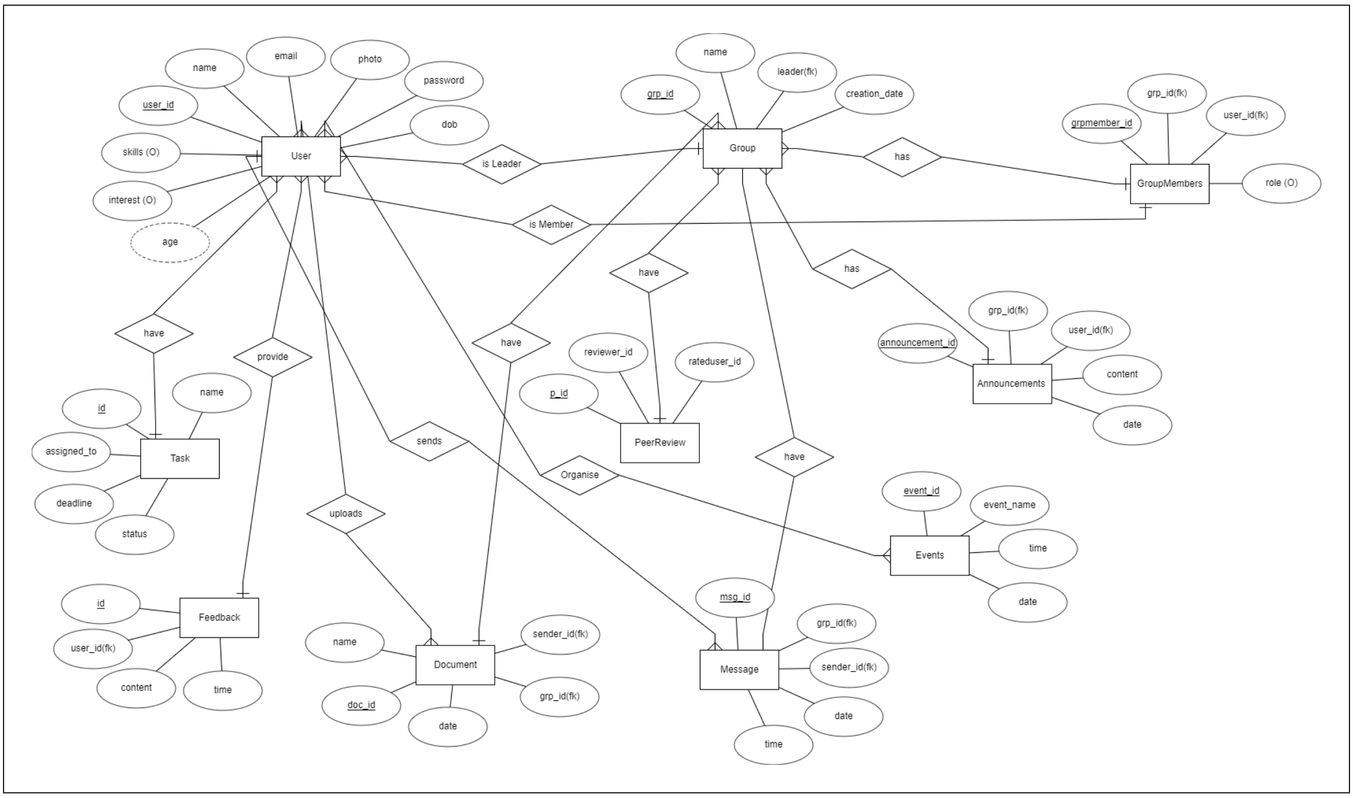


Figure 4.1: Entity Relationship diagram

4.2 Use Case View

Use Case Diagram:

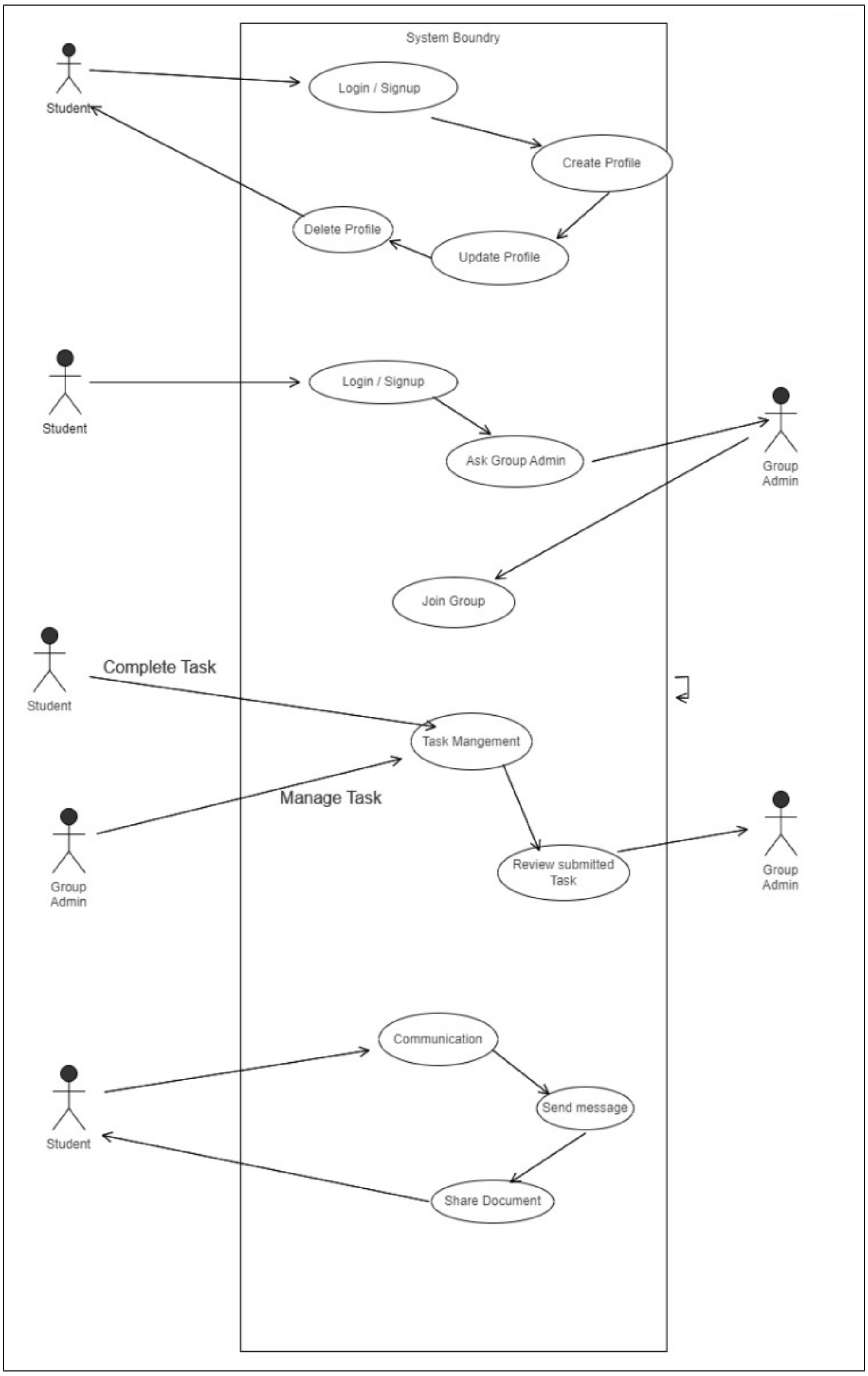


Figure 4.2: Use case diagram

4.3 Functional Model and Description

4.3.1 Data Flow Diagram

Level 0 Data Flow Diagram

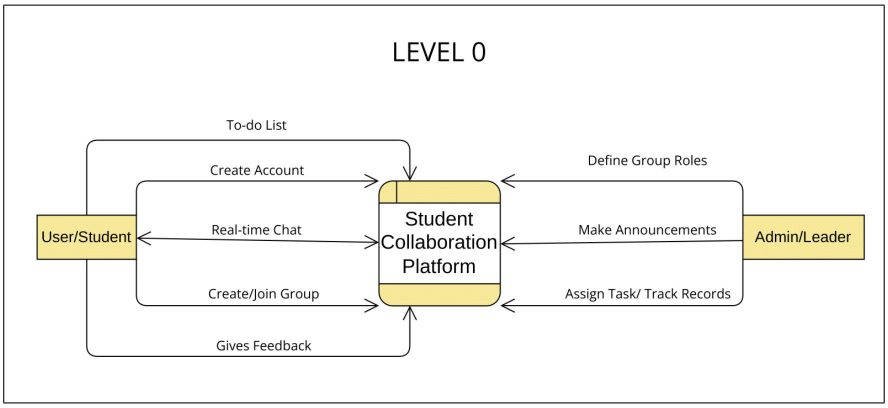


Figure 4.3: DFD Level0

Level 1 Data Flow Diagram

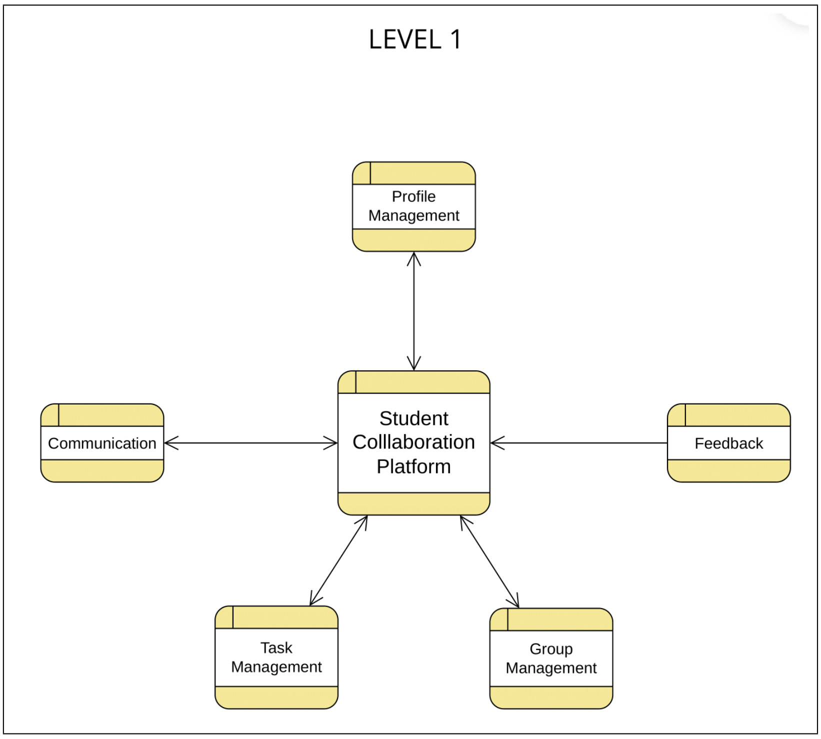


Figure 4.4: DFDLevel1

Level 2 Data Flow Diagram

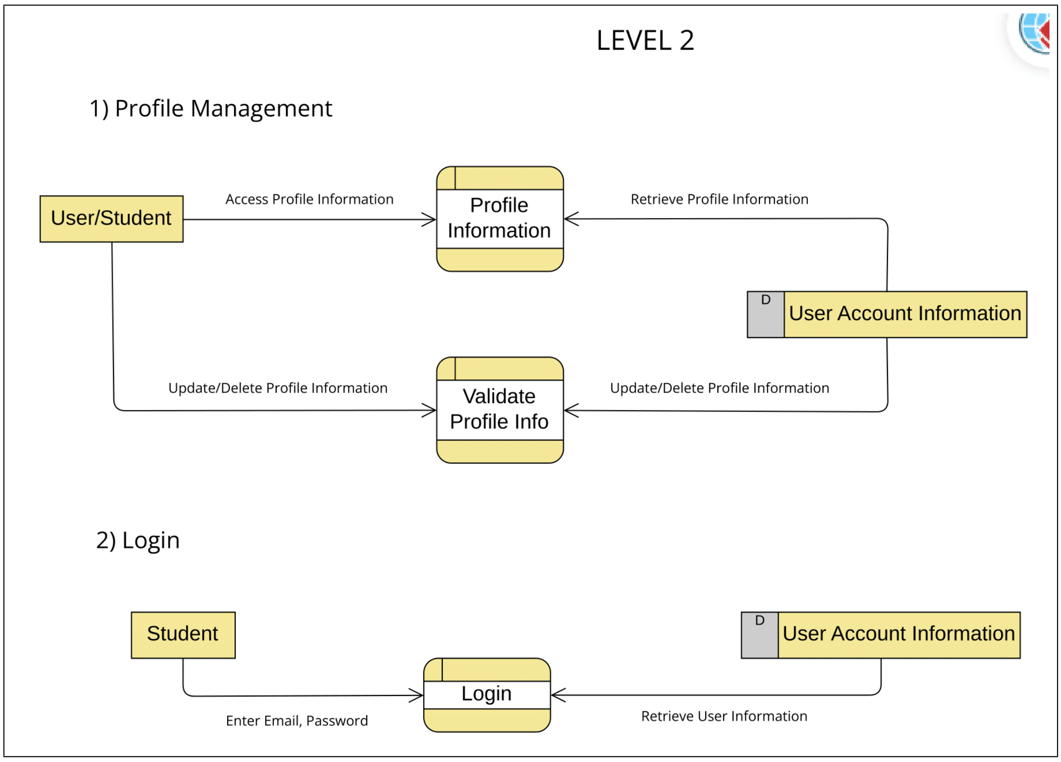


Figure 4.5: DFDLevel2_1

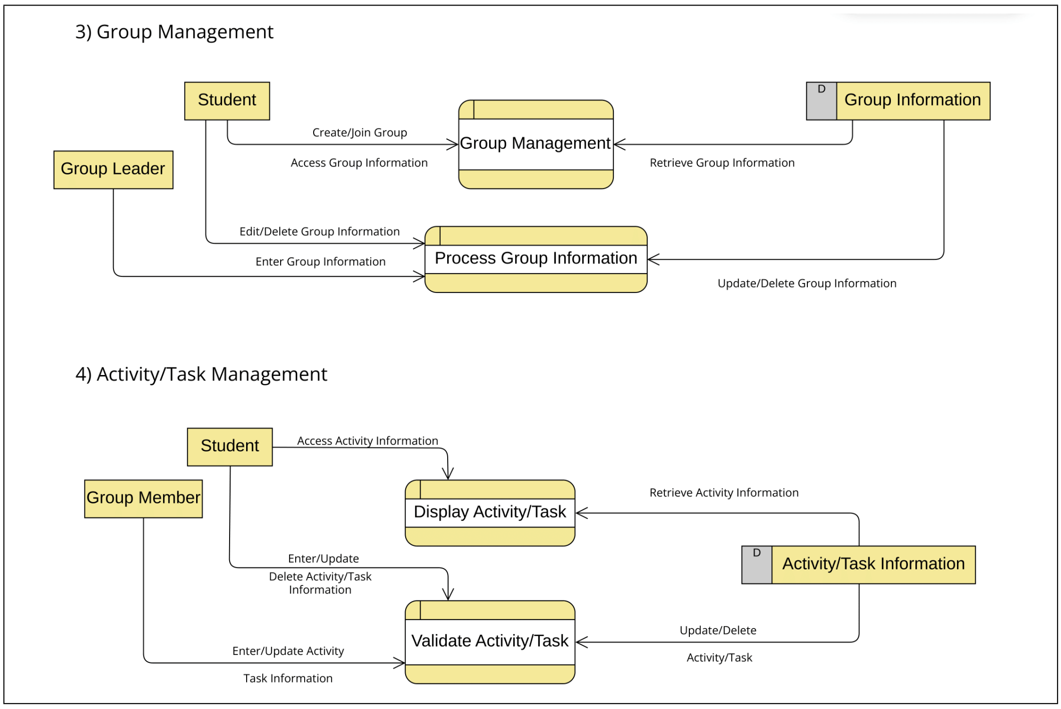


Figure 4.6: DFDLevel2_2

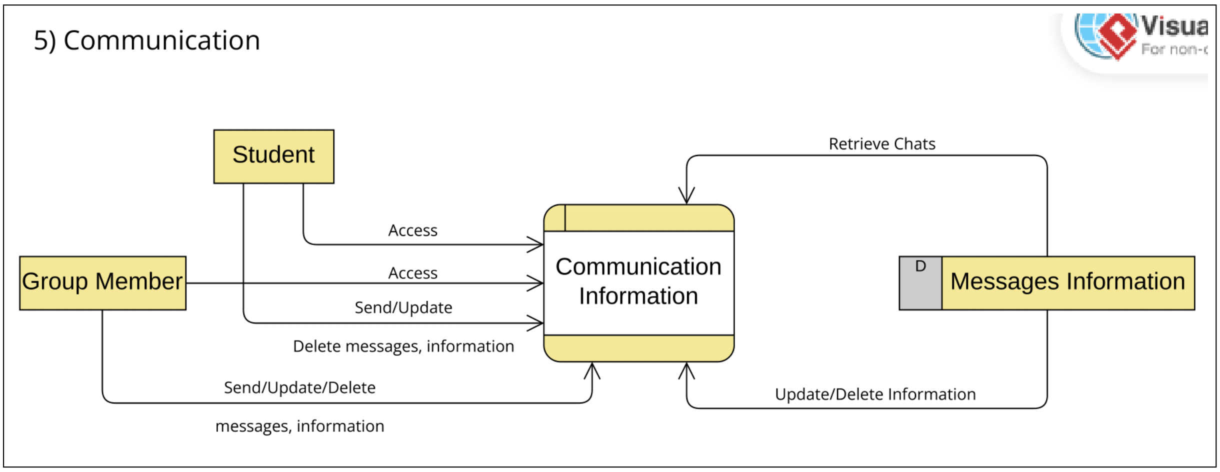


Figure 4.7: DFDLevel2_3

4.3.2 Description of functions

- **Registration:** : Users can register by providing necessary information.
- **Login:** Users can log in securely using their credentials.
- **Create Group:** Students can create groups for collaboration.
- **To-Do List:** Students can create and manage individual to-do lists.
- **Task Management:** Assign tasks to group members.
- **Real-Time Messaging:** Enable real-time messaging within groups

4.3.3 Swimlane Diagram:

- The Swimlane diagram represents the steps taken.

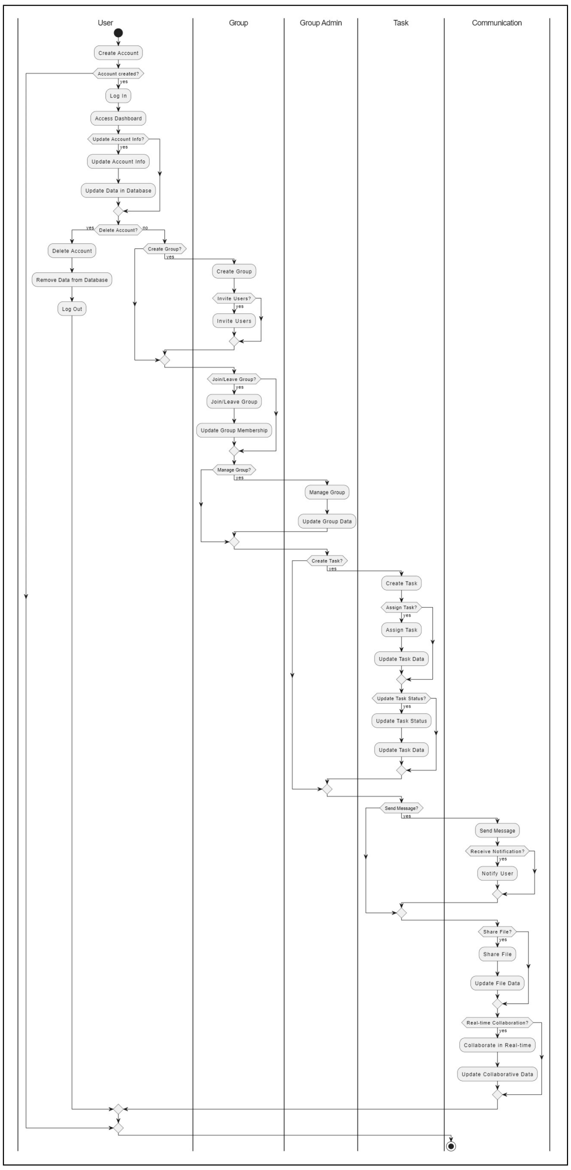


Figure 4.8: Swimlane diagram

4.3.4 Non Functional Requirements:

- **Scalability:** The platform should handle a growing number of users and data.
- **Response Time:** Ensure low latency for real-time communication features.
- **User Data Security:** Implement secure data transmission and storage.
- **Access Controls:** Define permissions and access controls for user roles.
- **User Interface:** Design an intuitive and user-friendly interface.
- **Mobile Responsiveness:** Ensure accessibility on various devices, especially mobile.
- **Backup and Recovery:** Regularly backup critical data and implement a recovery plan.
- **Continuous Improvement:** Plan for regular updates and improvements based on user feedback.

Chapter 5

Detailed Design Document

5.1 Component Design

5.1.1 Class Diagram

- **Student:** Represents a user of the system, specifically a student.

  Attributes:

  - Name
  - Email
  - ProfilePicture
  - AcademicDetails
  - Skills
  - Interests

  Relationships:

  - A student can be a member of one or more groups.
  - A student can create tasks within groups.
  - A student can be assigned tasks.

- **GroupAdmin:** Represents a user who has administrative privileges within a group.

  Attributes: Inherits attributes from the Student class.

  Relationships:

  - A group admin is a type of student.
  - A group admin manages one or more groups.
  - A group admin creates and manages tasks within their groups.

- **Group:** Represents a group formed by students for collaboration.

  Attributes:

  - GroupName
  - Leader (Reference to a Student who is the leader of the group)
  - Members (List of Student objects who are members of the group)

  Relationships:

  - A group has one leader who is a student.
  - A group can have multiple members who are students.
  - A group contains tasks assigned to its members.

- **Task:** Represents a task assigned within a group.

  Attributes:

  - TaskName
  - Description
  - Deadline
  - AssignedTo (Reference to a Student who is assigned the task)

  Relationships:

- A task is created within the context of a group.
- A task is assigned to a student (either a member or a group admin).

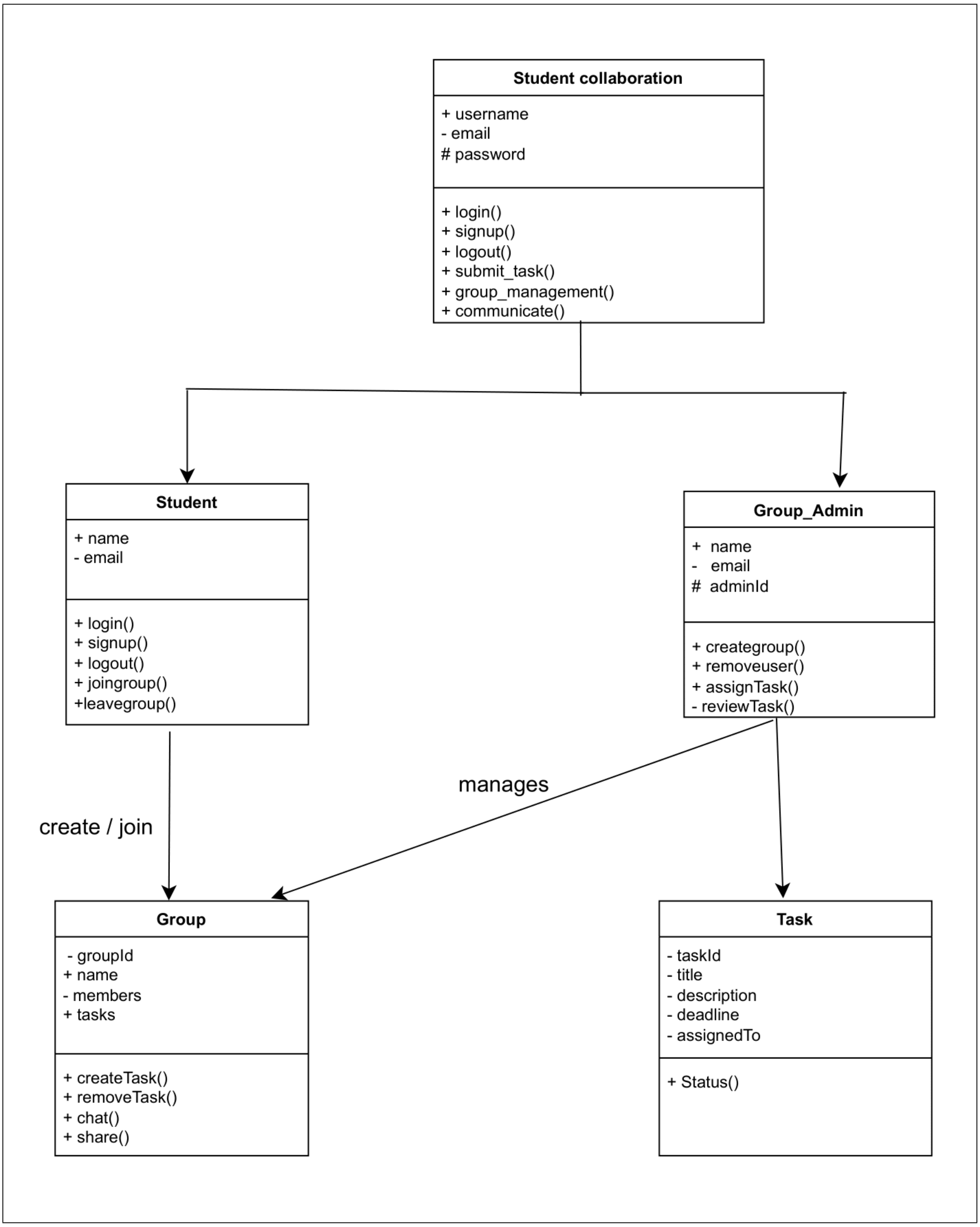


Figure 5.1: Class Diagram

5.1.2 Sequence Diagram

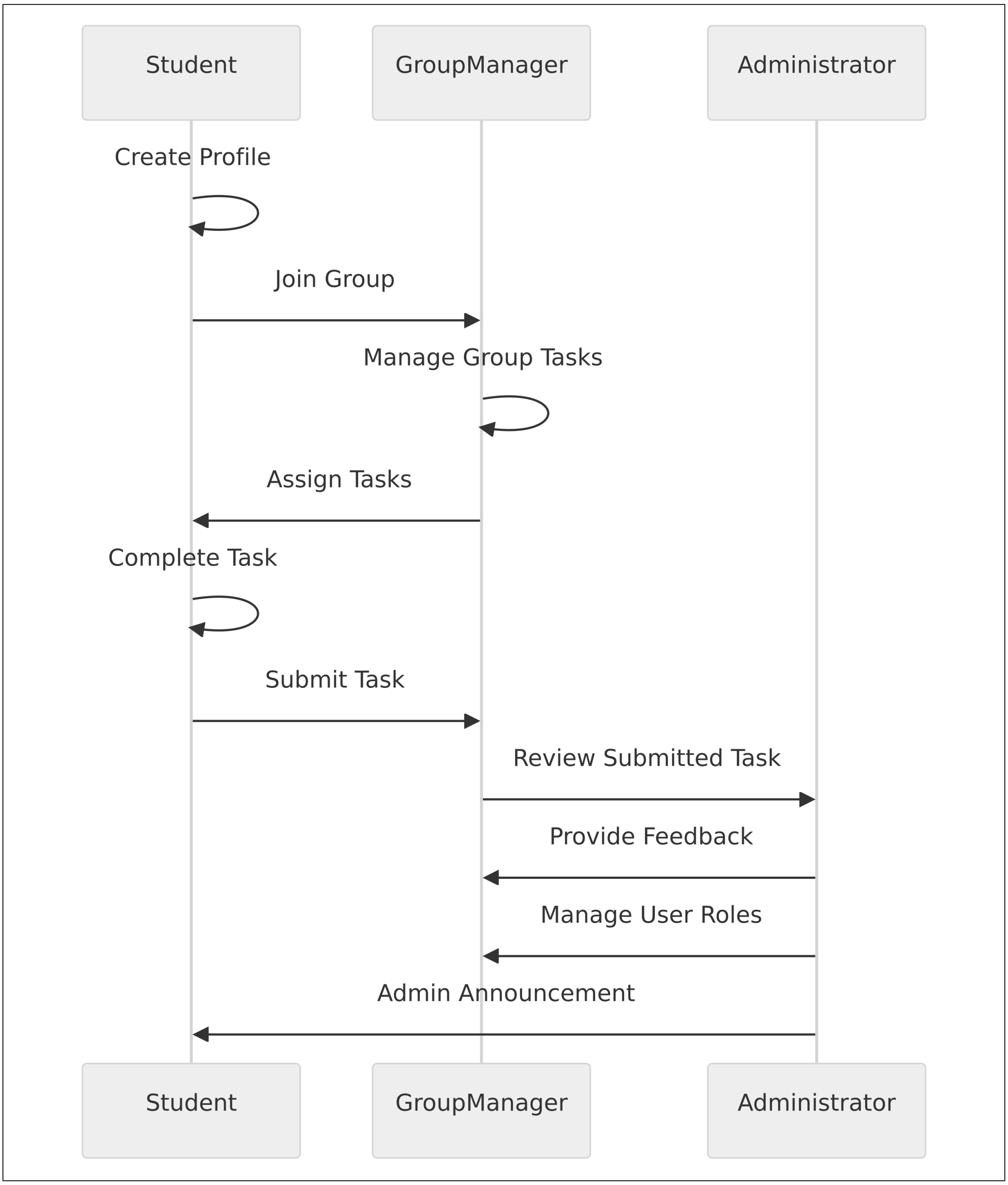


Figure 5.2: Sequence Diagram

5.1.3 Component Diagram

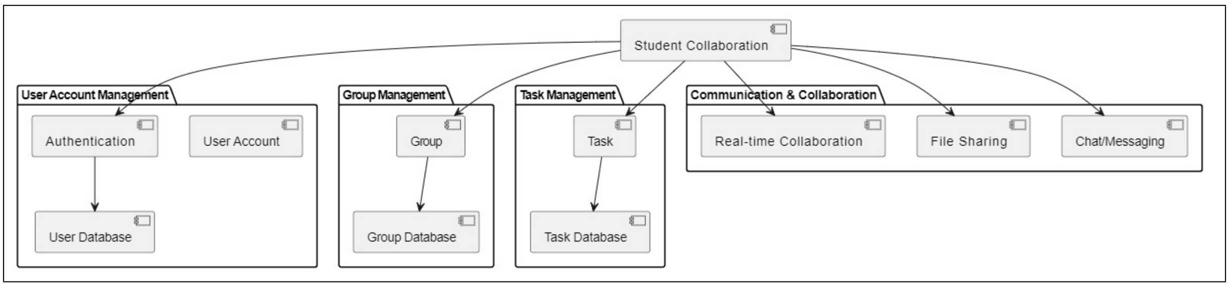


Figure 5.3: Component Diagram

5.1.4 Deployment Diagram

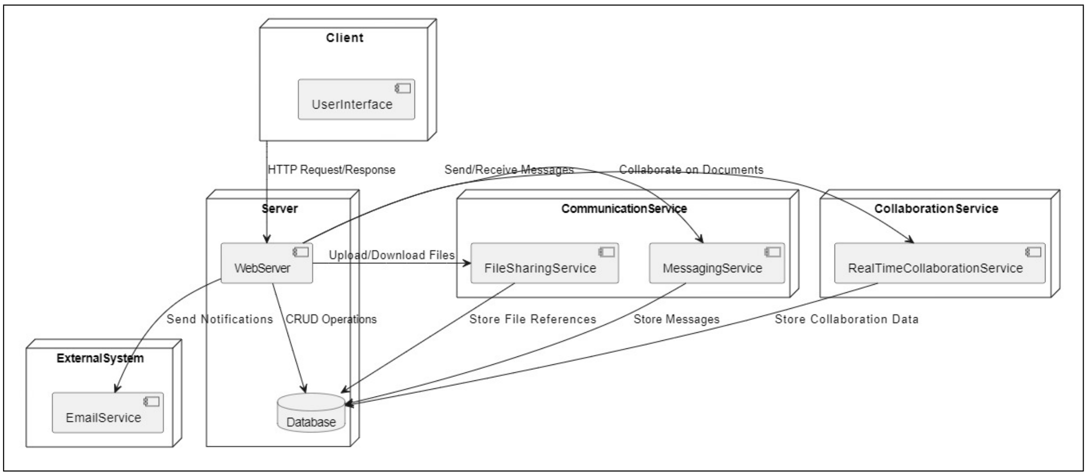
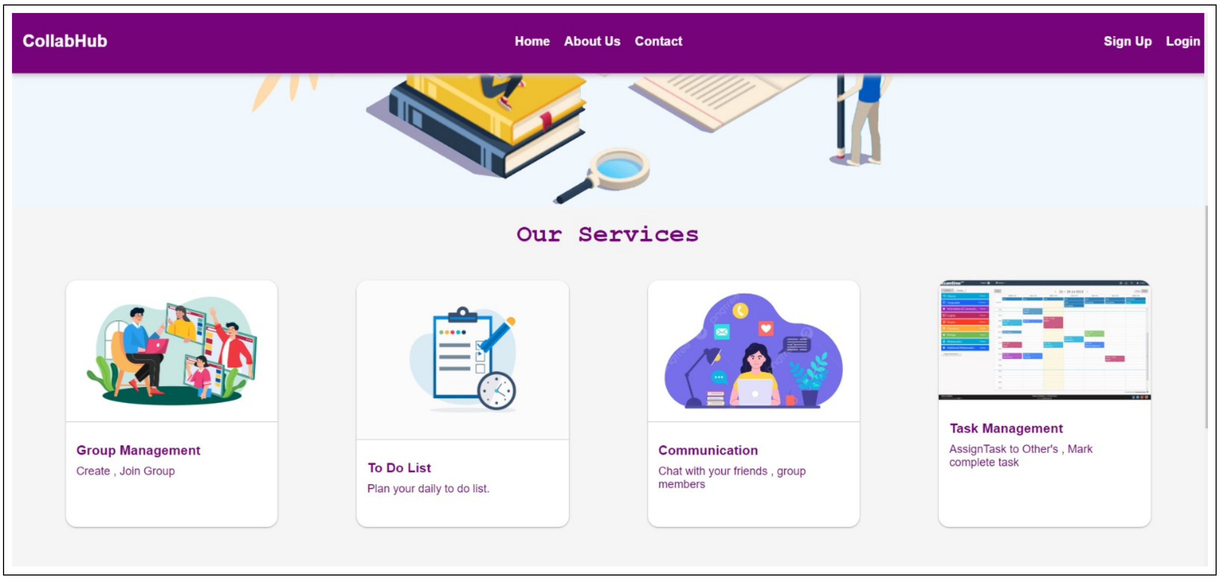


Figure 5.4: Deployment Diagram

5.2 Navigation Flow



CollabHub

Home About Us Contact

Sign Up Login

Our Services

Group Management

Create , Join Group

To Do List

Plan your daily to do list.

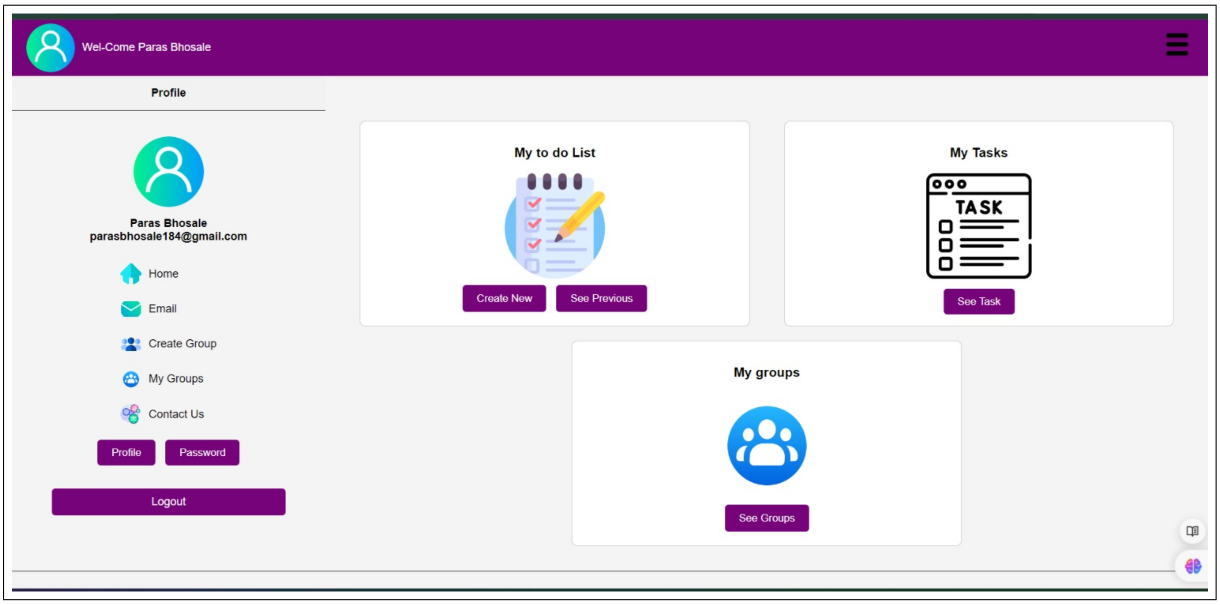
Communication

Chat with your friends , group members

Task Management

AssignTask to Other's , Mark complete task

Figure 5.5: Home



Wel-Come Paras Bhosale

Profile

Paras Bhosale
parasbhosale184@gmail.com

Home

Email

Create Group

My Groups

Contact Us

Profile Password

Logout

My to do List

Create New See Previous

My Tasks

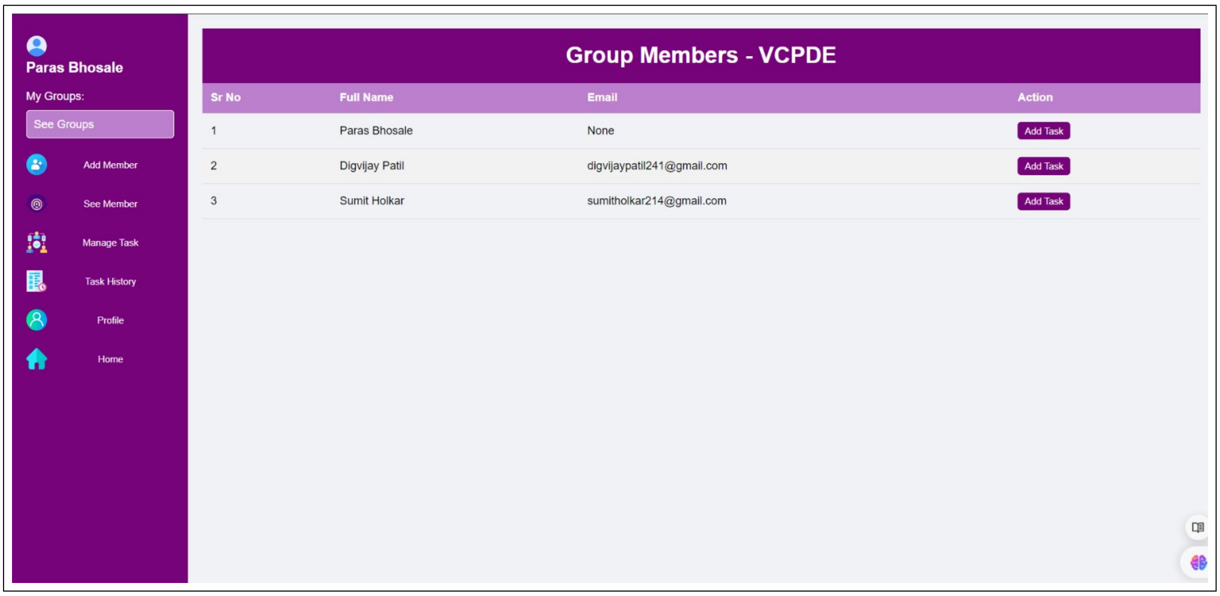
TASK

See Task

My groups

See Groups

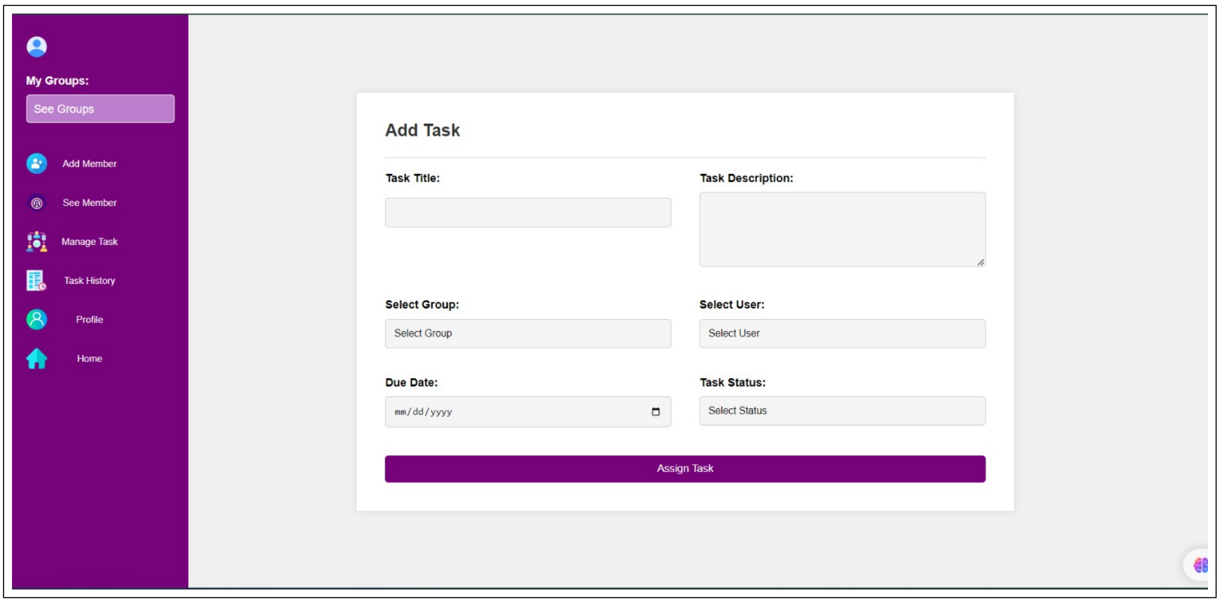
Figure 5.6: Profile



Group Members - VCPDE

Sr No	Full Name	Email	Action
1	Paras Bhosale	None	Add Task
2	Digvijay Patil	digvijaypatil241@gmail.com	Add Task
3	Sumit Holkar	sumitholkar214@gmail.com	Add Task

Figure 5.7: SeeMember



Add Task

Task Title:

Task Description:

Select Group: Select Group

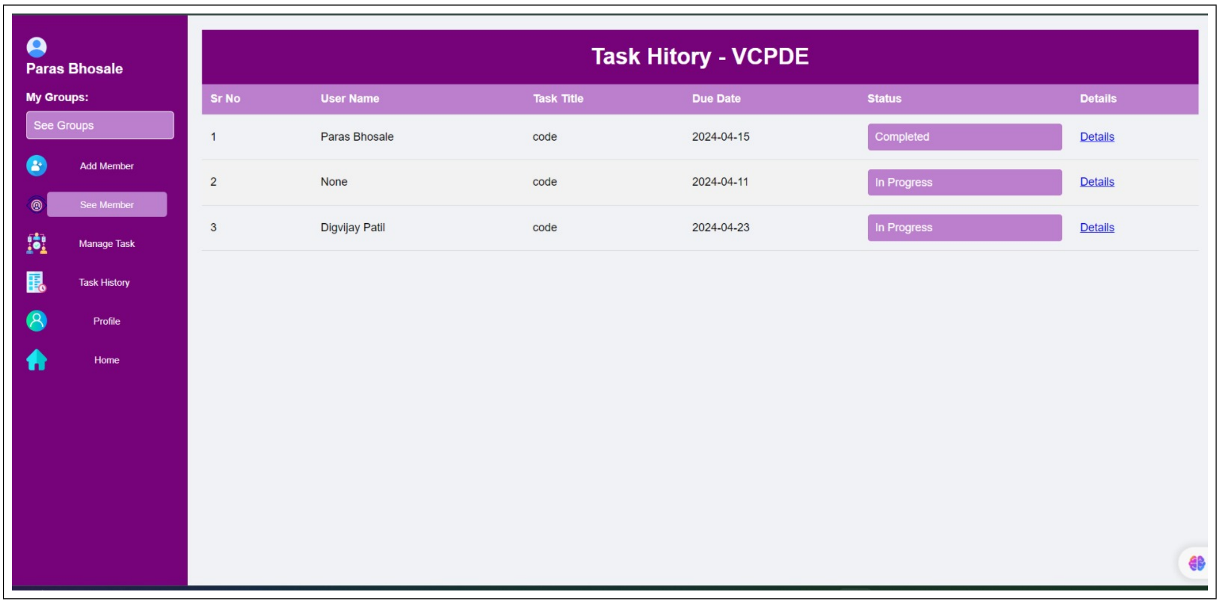
Select User: Select User

Due Date: mm/dd/yyyy

Task Status: Select Status

Assign Task

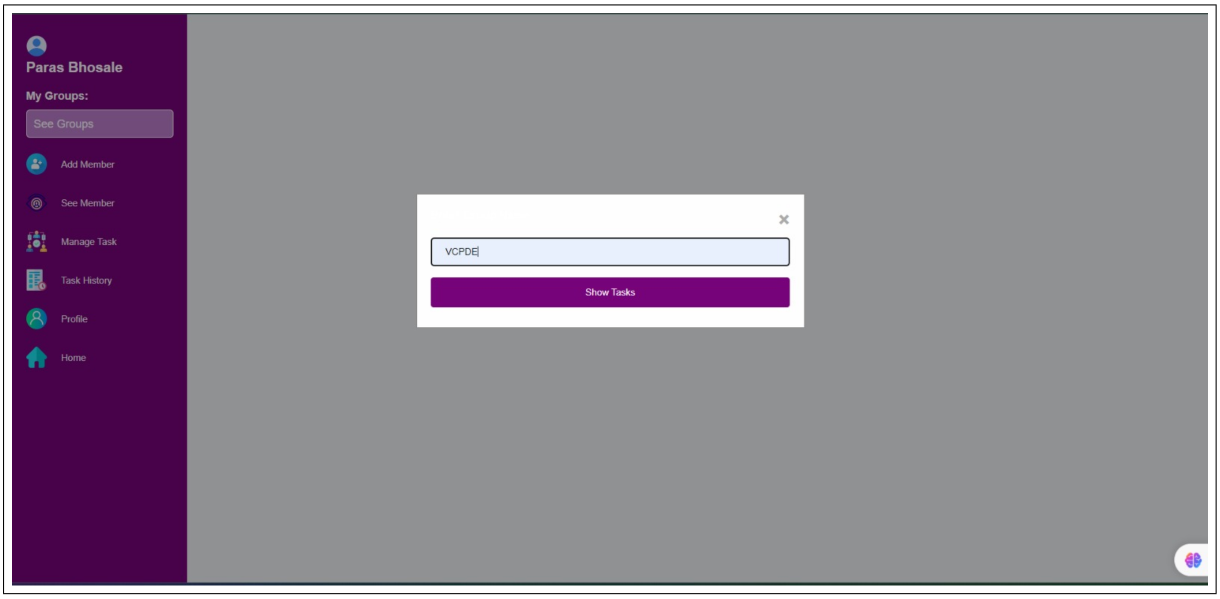
Figure 5.8: AddTask



Task Hitory - VCPDE

Sr No	User Name	Task Title	Due Date	Status	Details
1	Paras Bhosale	code	2024-04-15	Completed	Details
2	None	code	2024-04-11	In Progress	Details
3	Digvijay Patil	code	2024-04-23	In Progress	Details

Figure 5.9: TaskHistory



VCPDE

Figure 5.10: ShowTask

Chapter 6

Summary and Conclusion

The Student Collaboration Platform (SCP) project aims to address the challenges faced by students in facilitating effective communication,project management, Task management ,organization, and collaboration . Through the design, develo,projectpment, and implementation of the Student Collaboration Platform, the project seeks to enhance organizational efficiency, promote collaborative learning, and empower students to leverage technology for academic success.

The Student Collaboration Platform will integrate key features such as user authentication, group and project management tools, messaging functionalities, integration with external tools, notifications, student profile analytics, security measures, and a feedback and recognition system. These features are designed to streamline task management processes, foster seamless communication and collaboration, and provide valuable insights to educators for optimizing student engagement and productivity.

By leveraging modern technologies and best practices in software development, the SCP project aims to deliver a user-friendly and innovative platform that transforms traditional educational practices into dynamic and interactive learning experiences. The platform will empower students to work together, share resources, and collaborate on projects, ultimately enhancing the quality and outcomes of collaborative learning within educational settings.

In conclusion, the development of the Student Collaboration Platform (SCP) represents a significant opportunity to modernize educational environments and promote collaborative learning practices. By harnessing the power of technology and digital tools, the SCP project will enable students to create inclusive and engaging learning communities where users can thrive and succeed. The successful implementation of the SCP will not only improve organizational efficiency and communication within users but also contribute to the development of essential 21st-century skills among students, including teamwork, communication, and problem-solving. Through continuous feedback, iteration, and adaptation, the SCP will evolve to meet the evolving needs of educational stakeholders and contribute to positive educational outcomes.