Design Document

for

ProfInfo Central

Version 1.0

Prepared by

Group : 2

Group Name: Error 404: team not found

Pavani priya	220415	gppriya22@iitk.ac.in
Sontam Deekshitha	221075	sontamd22@iitk.ac.in
Lakshyta Mahajan	220581	lakshyta22@iitk.ac.in
Kartik	220503	kartik22@iitk.ac.in
Atharv Moghe	220250	atharvm22@iitk.ac.in
Prabhat Kumar Yadav	220774	prabhatky22@iitk.ac.in
Nilesh Maneshwar	220715	mnilesh22@iitk.ac.in
Kuldeep Sandip Thakare	220557	kuldeeps22@iitk.ac.in
Mohd Nasar Siddiqui	220661	snasar22@iitk.ac.in
Sanapala Jaswanth	220955	sjaswanth22@iitk.ac.in

Course: CS253

Mentor TA: Abhilash Chandra

Date: 09-02-2024

Revisions

Version	Primary Author(s)	Description of Version	Date Completed
1.0	Error 404: team not found	Built a website where students can access all the professional and the research-oriented information regarding many different professors of the campus from different departments and can also request for projects under the professors according to their specific interests.	09/02/2024

1 Context Design

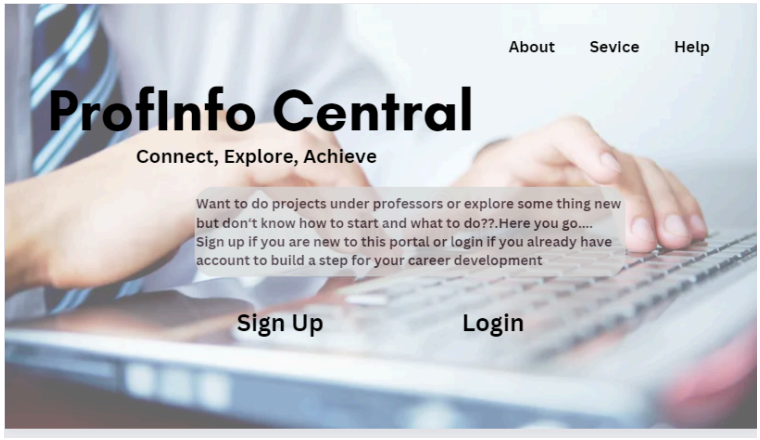
1.1 Context Model

The software system is designed to operate independently, without any dependencies on existing software systems or the need for integration with additional extensions. It is intended to function as a standalone solution, and as such, there is no requirement for interaction with other software systems in the environment. Consequently, the application of a context model or diagram is not relevant for the development of this software system.

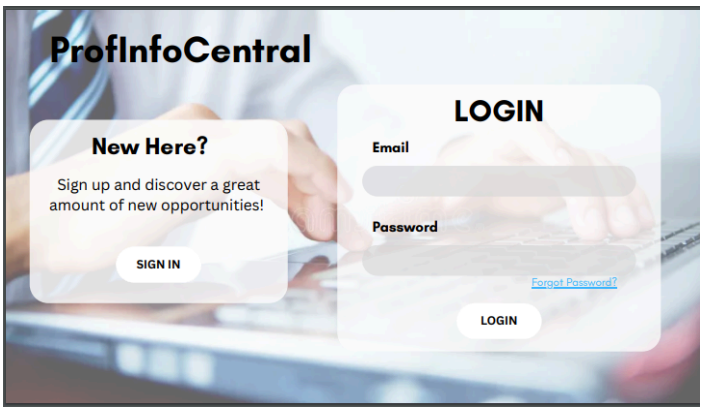
1.2 Human Interface Design

The system will be accessed by all users through one website – users will log in with their accounts and be shown different pages depending on whether their account is marked as Student or Professor.

User Interface :
HOME PAGE:

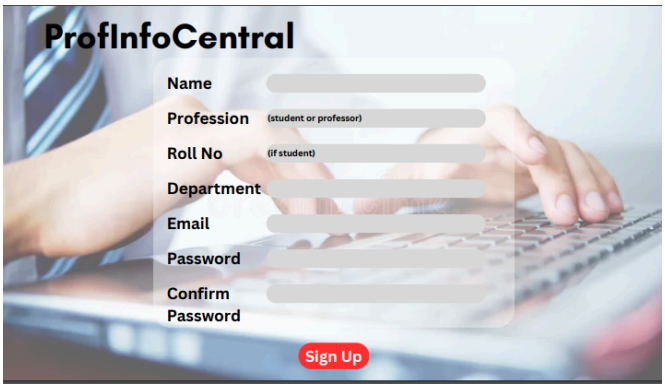


1. Login page
   We can get into the website by entering our username and password in the login page.



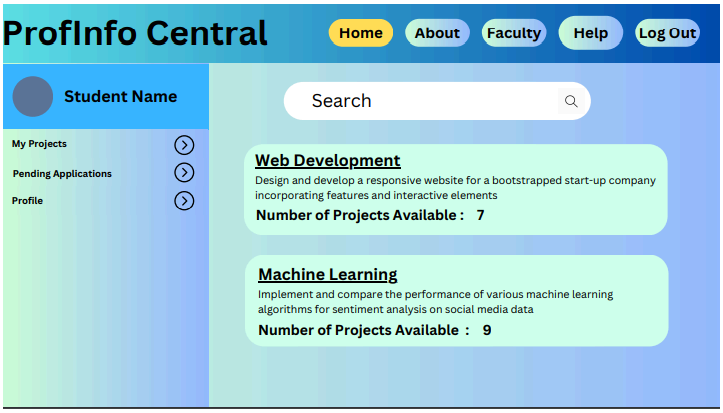
2. Registration Page

Students who are new to the website, register here.



Students' Interface :

1. Home Page
   - Students' home page consists of all the projects available for them on the basis of a particular field.

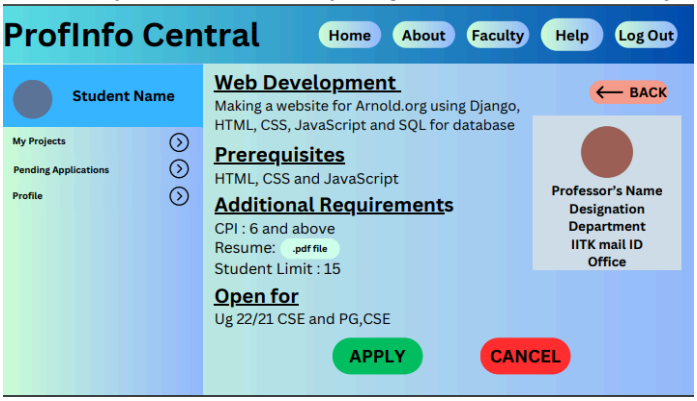


2. Explore Projects
   - Students can select the topics they are interested in, which shows them all
   - 

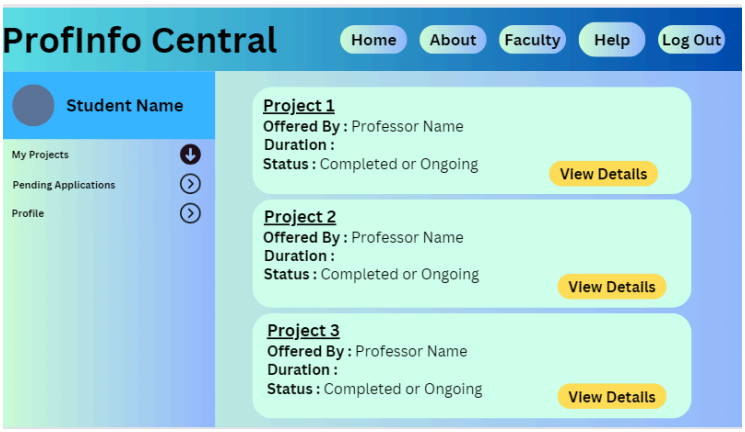
the projects related to that topic.

3. Project Details

- After selecting a particular project, it shows all the details related to that project. And then, they can either apply or go for some other project.

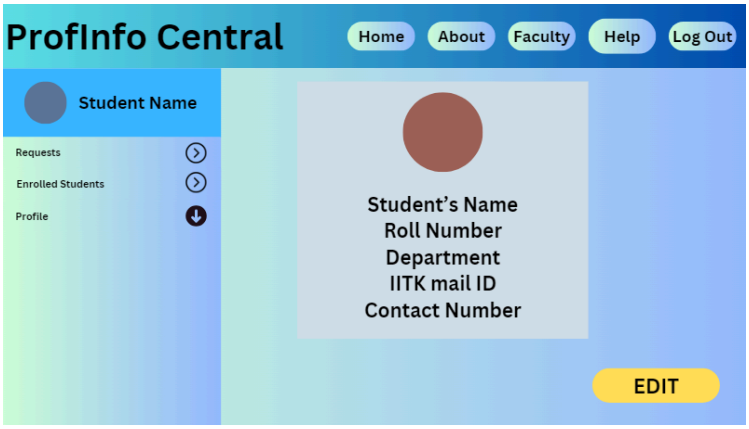


4. Check Details:
   As our website allows a student to apply for 3 projects at a time, students go to my projects where it shows the accepted projects.



5. Pending details
   Students can check the details of the projects for which they applied, whether they are accepted or rejected.

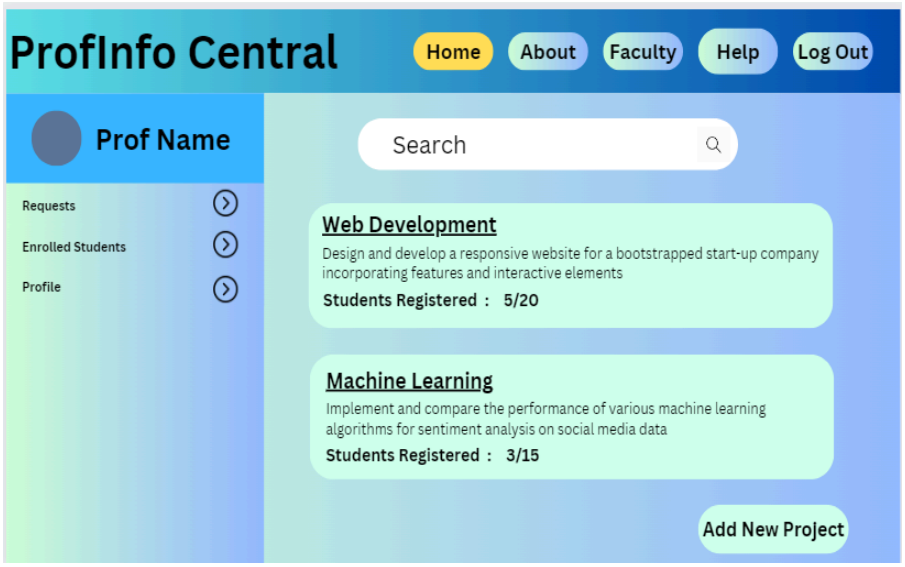
6. Student profile



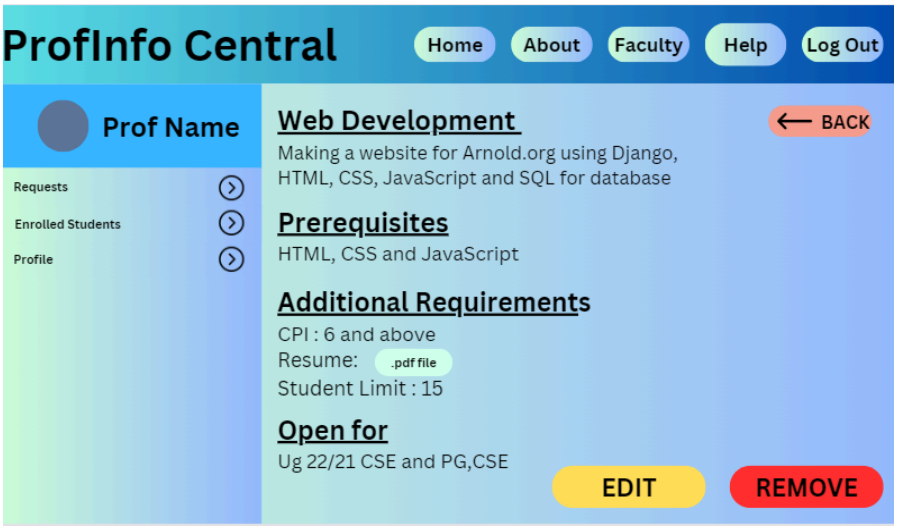
Professors' Interface :

1. Home Page

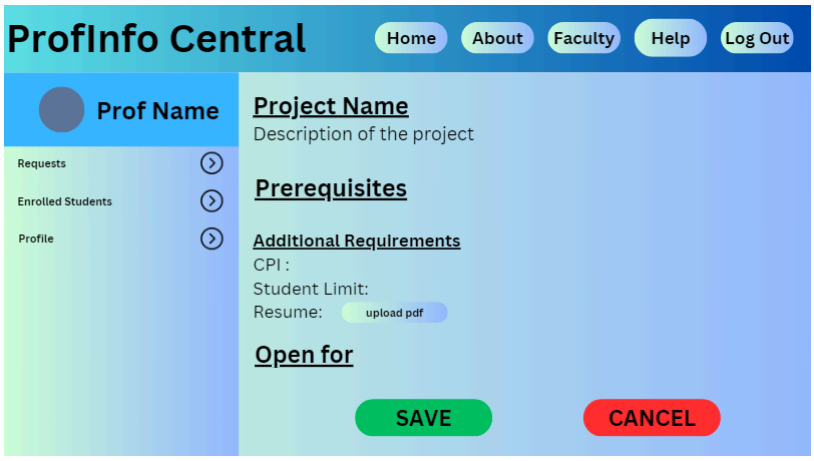
    - As the professor logs in to the website, this page opens which shows all the projects that the professors has offered for the semester.



2. Project Details
    - As the professors' selects a particular project, it shows all the details he provided and there are also options like edit and remove which he can use to edit the information and remove the project.

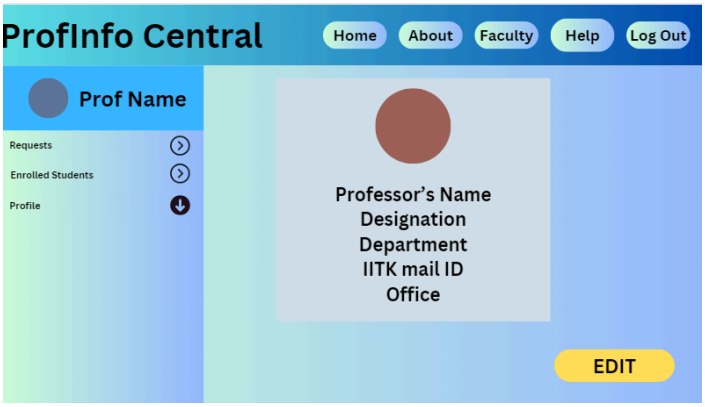


3. Add new project
    - Professor can use the below provided template to add a new project. After entering all the required information, the professor can either save or cancel the project.

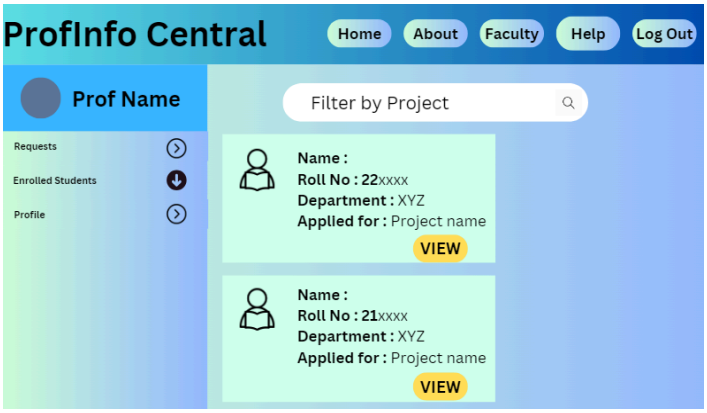


4. Dashboard

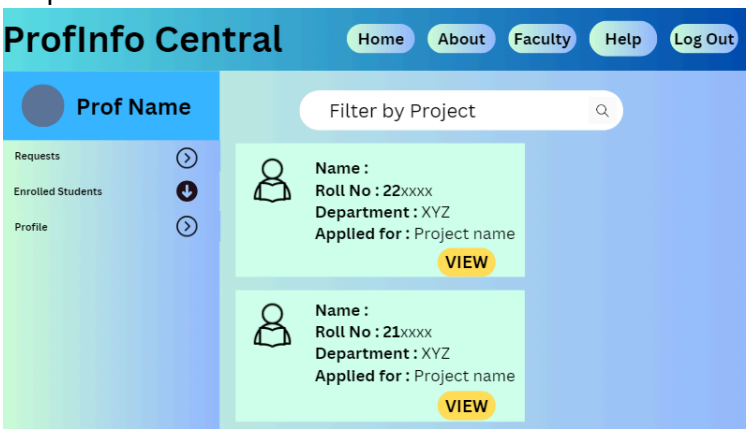
    There is a dashboard which shows the professor's profile, enrolled students and the requests for their projects.
    - Prof Name



- Enrolled Students



- Requests

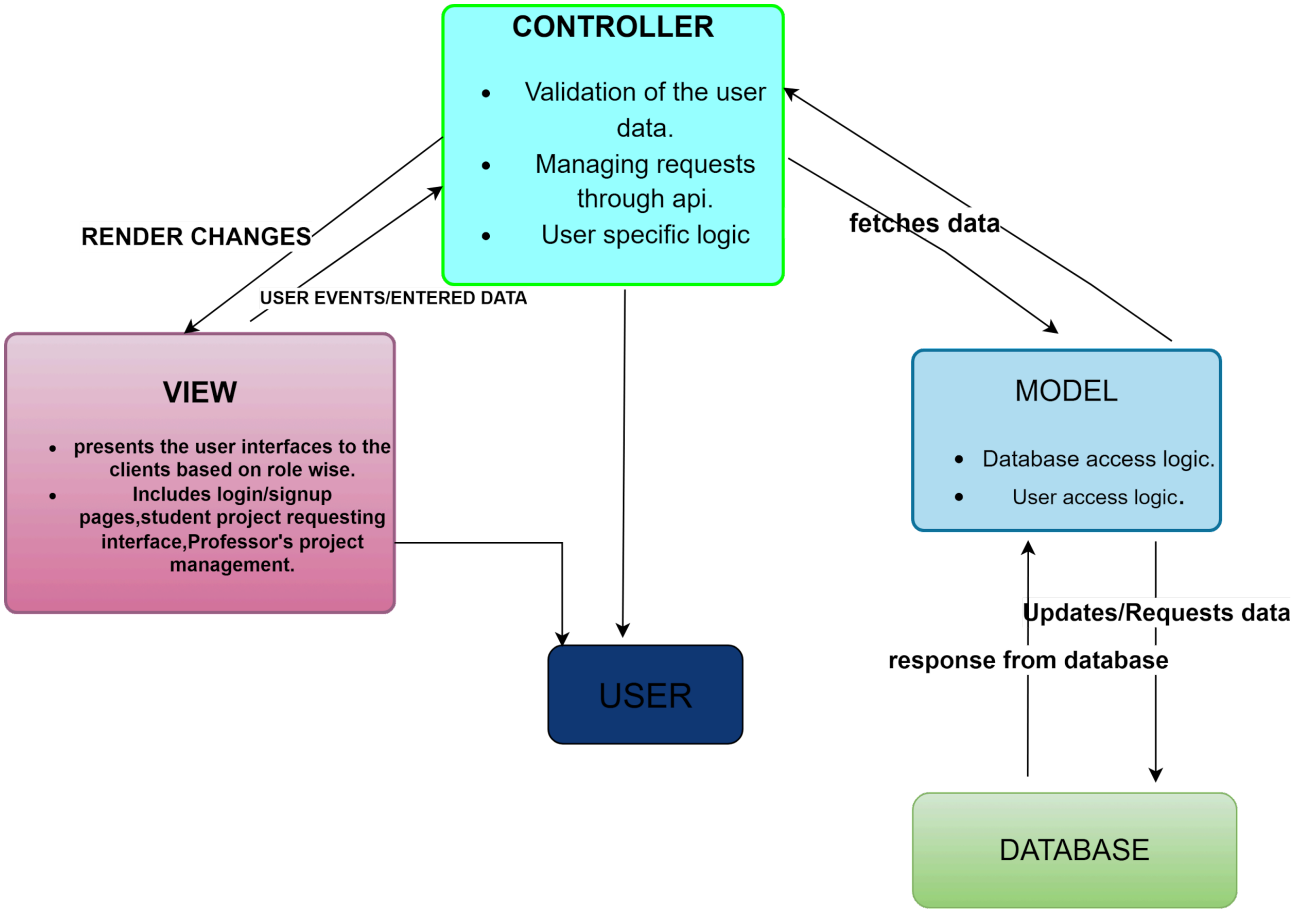


2 Architecture Design

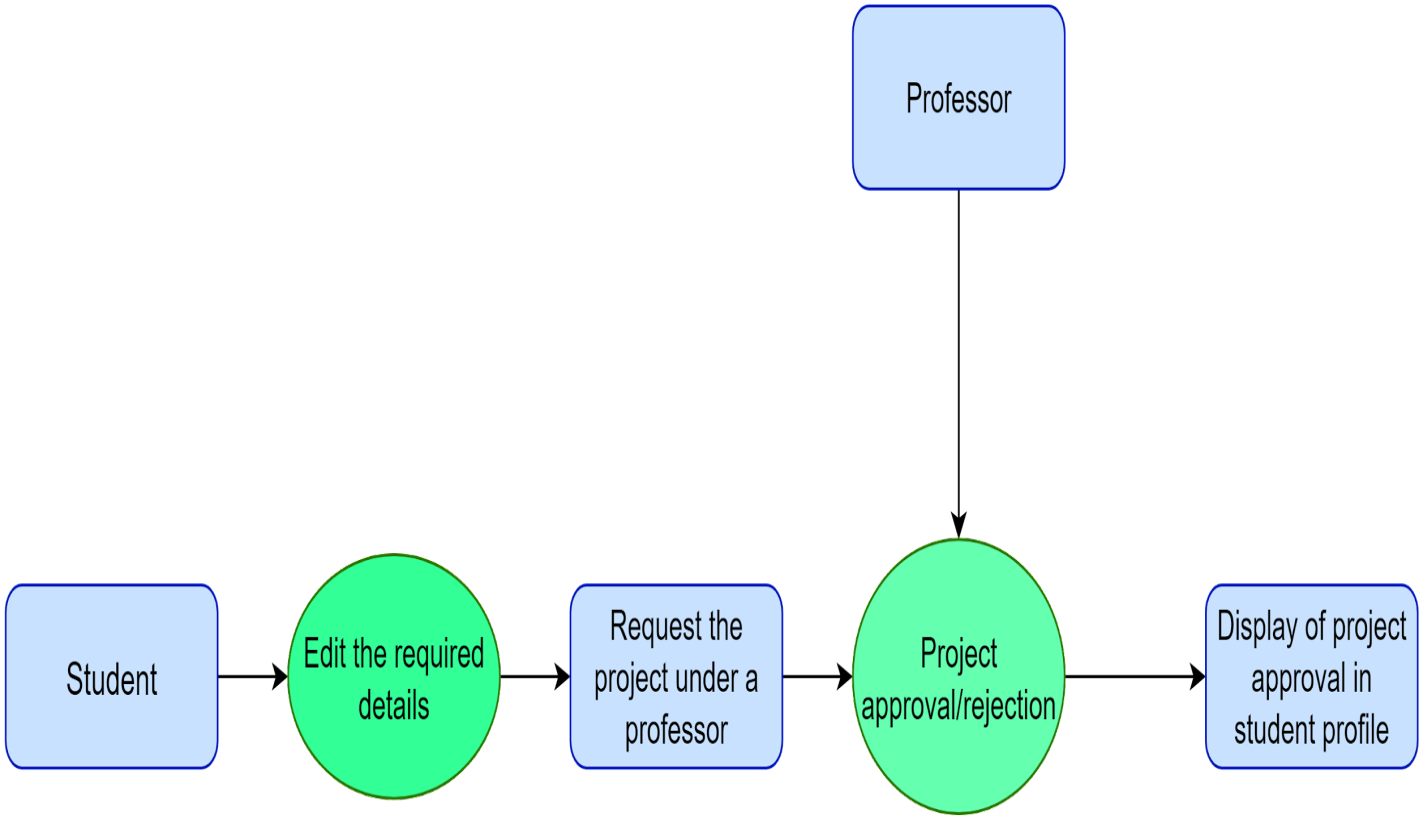
We plan to use a hybrid of Model View Controller Architecture along with pipe and filter Architecture and client server interaction design.

Model view Controller Architecture allows the system to display the different data based on user/role which is accessing the system.Controller Processes incoming requests, updates the model as necessary, and selects the appropriate view to render and handles user authentication, project submission, and profile management.The model is responsible for implementing the user access control and communication with the database.The server Handles incoming requests from clients and generates responses and also manages user authentication, user profiles, and project management.The database stores persistent data such as user profiles, projects, and project submissions.The client side contains the user's interface to the web application.It sends requests to the server and displays the responses. Thus this architecture allows implementation of various functional requirements such as role-dependent views and functions.It also ensures that the system's privacy/security related non-functional requirements are fulfilled.

The processes of requesting for a project, and the approval of a project necessitate the use of a pipe and filter architecture, because object for requesting for a project created are acted upon by a number of functions by various users. The result is finally appended to the applied projects history of the student. For the implementation of this flow of data from a source to sink, the pipe and filter architecture is well suited.

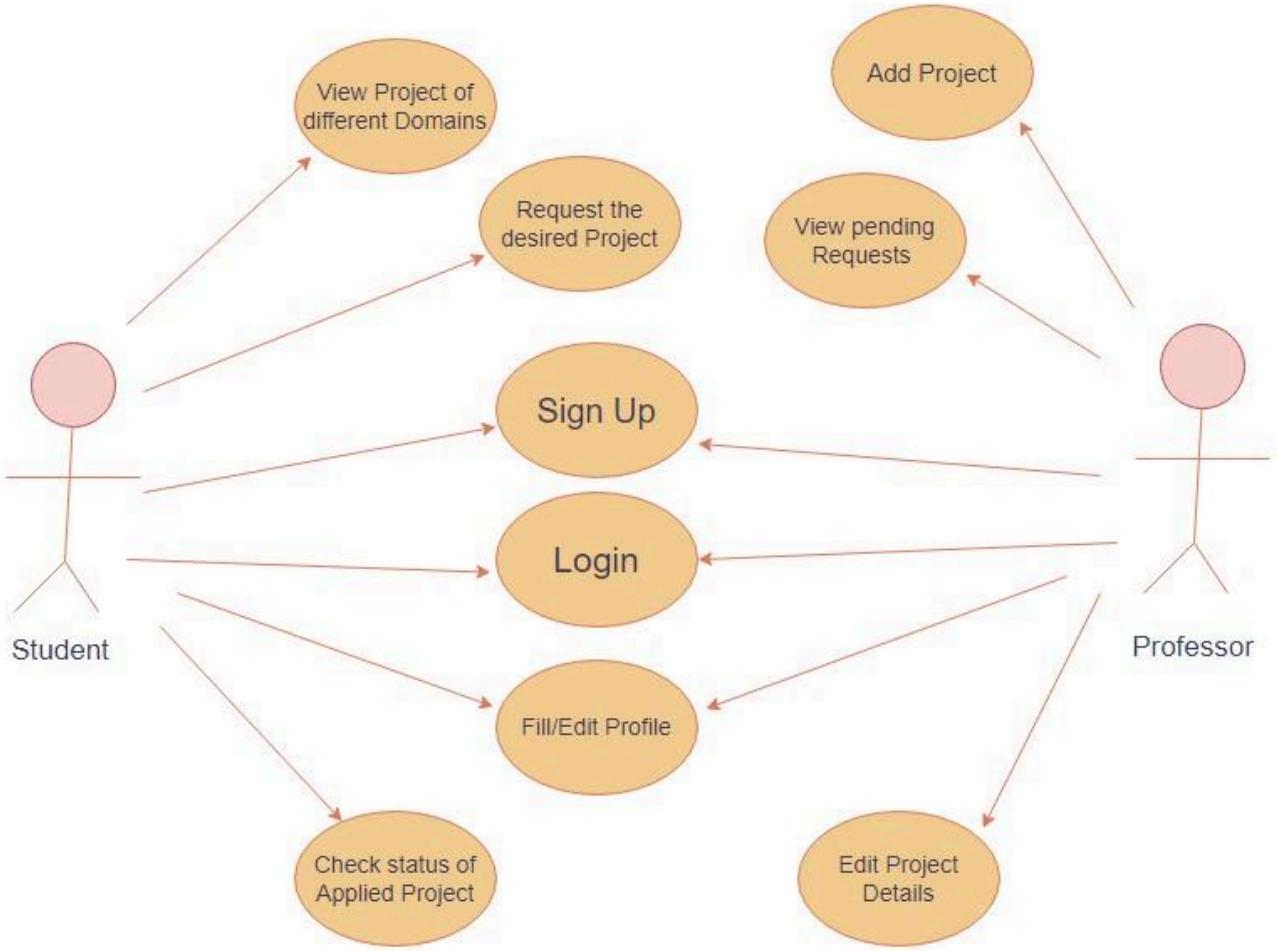


Pipe and Filter Architecture for Appointment and Prescription processing:

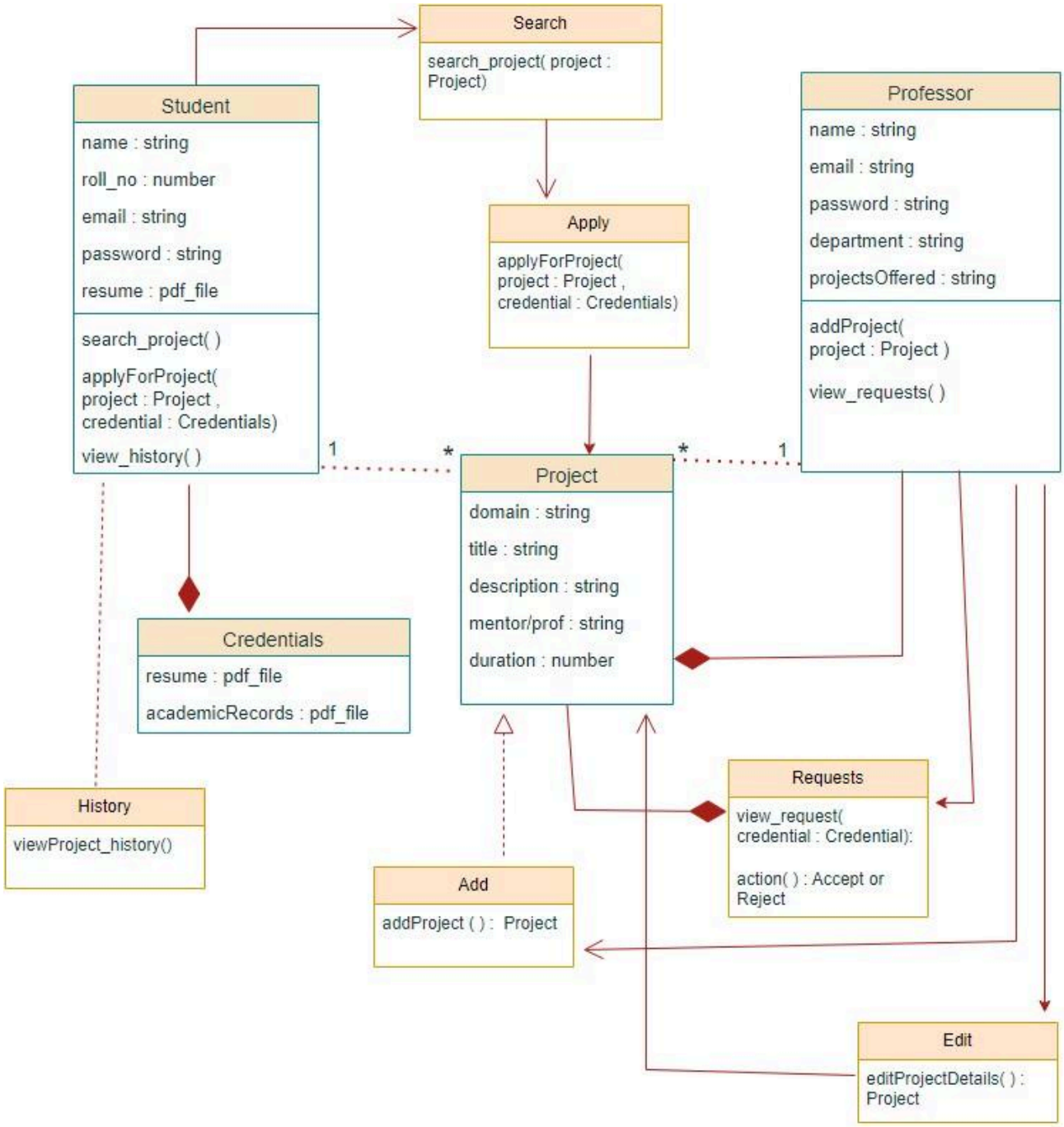


3 Object Oriented Design

3.1 Use Case Diagrams

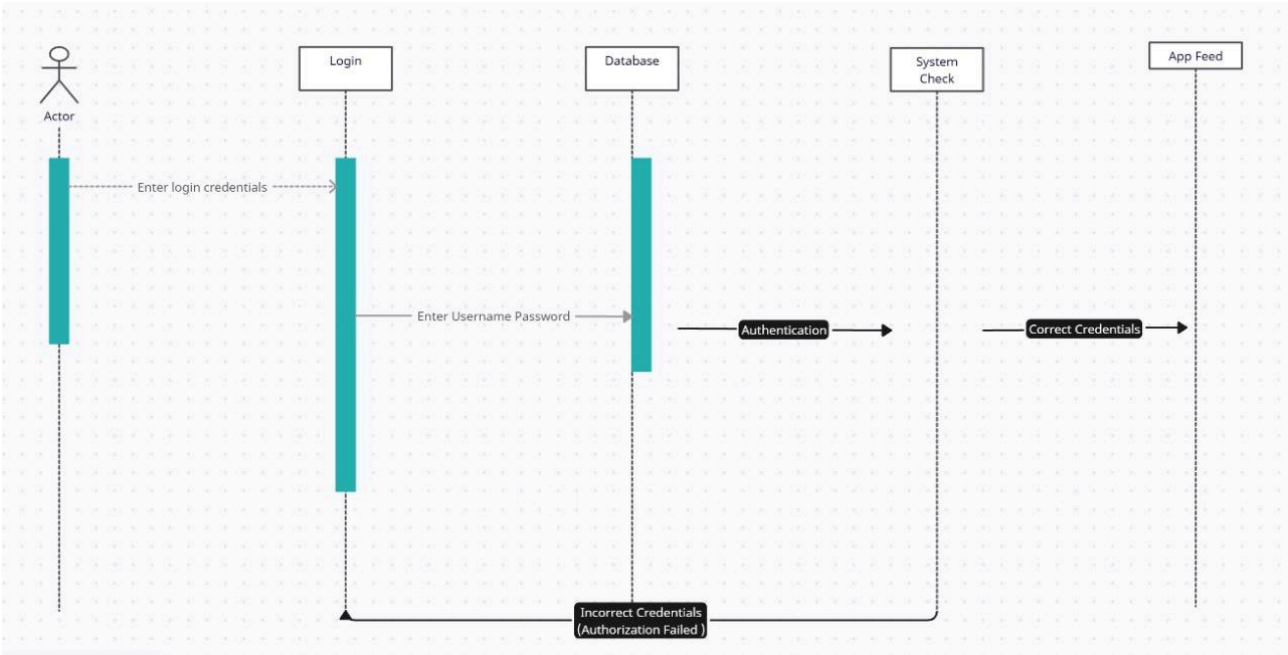


3.2 Class Diagrams

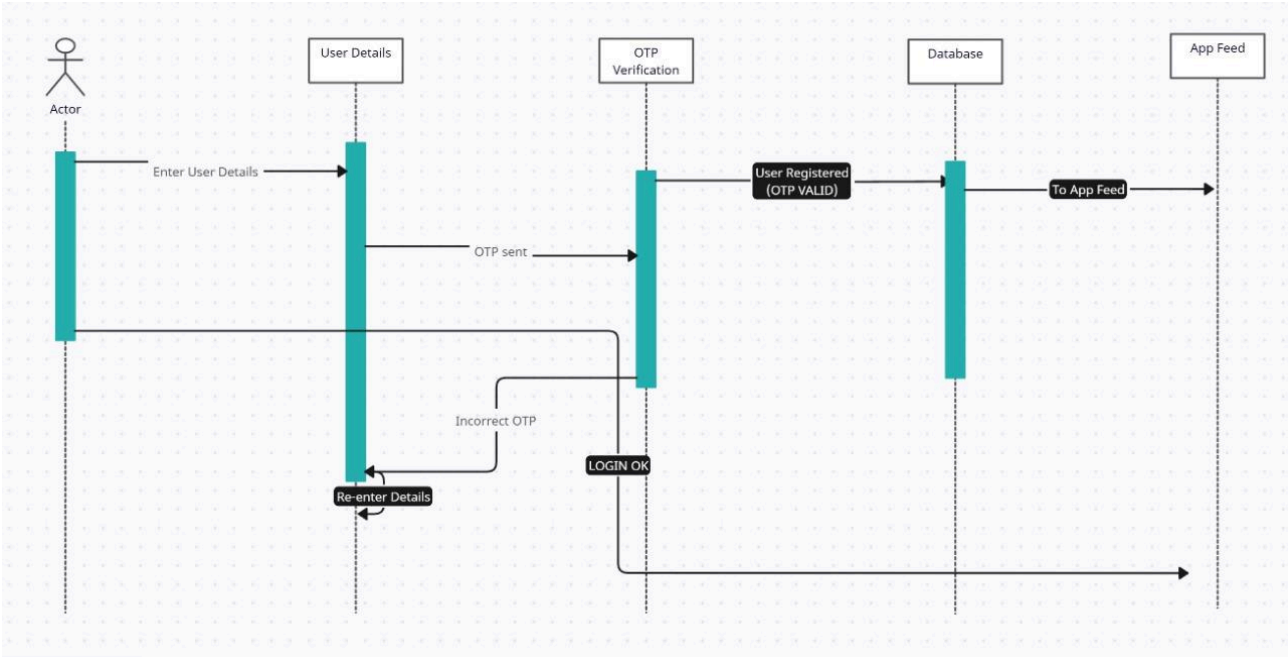


3.3 Sequence Diagrams

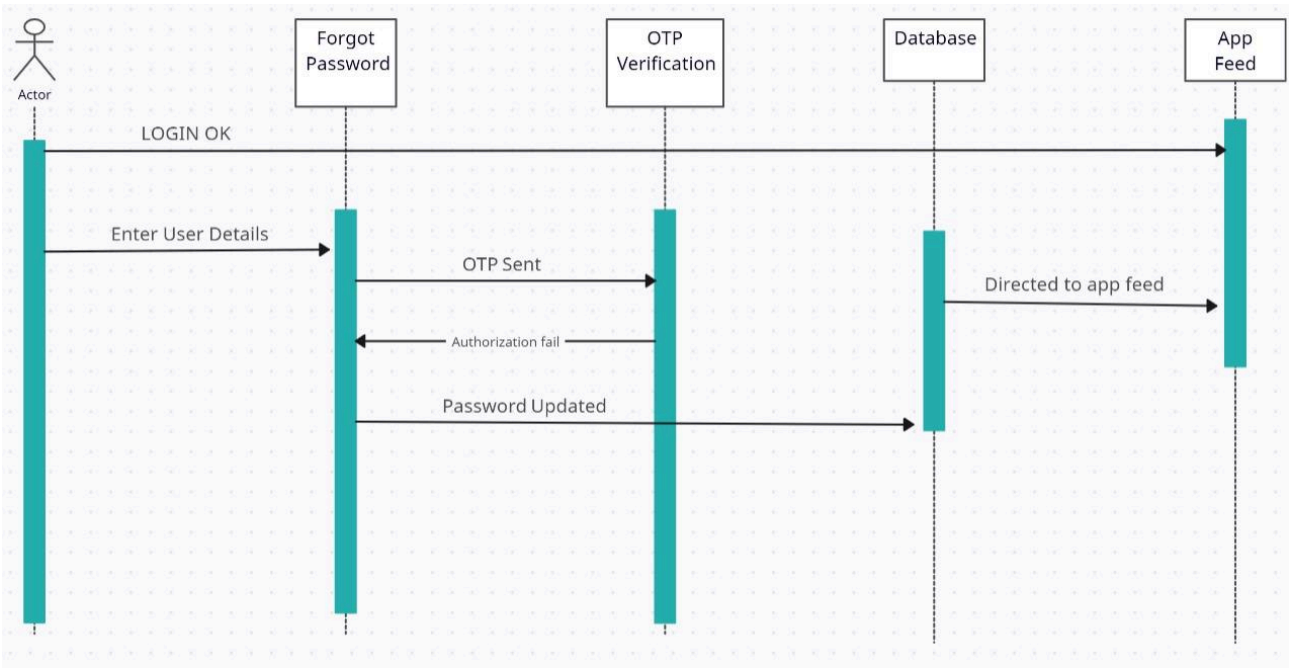
1. LOGIN



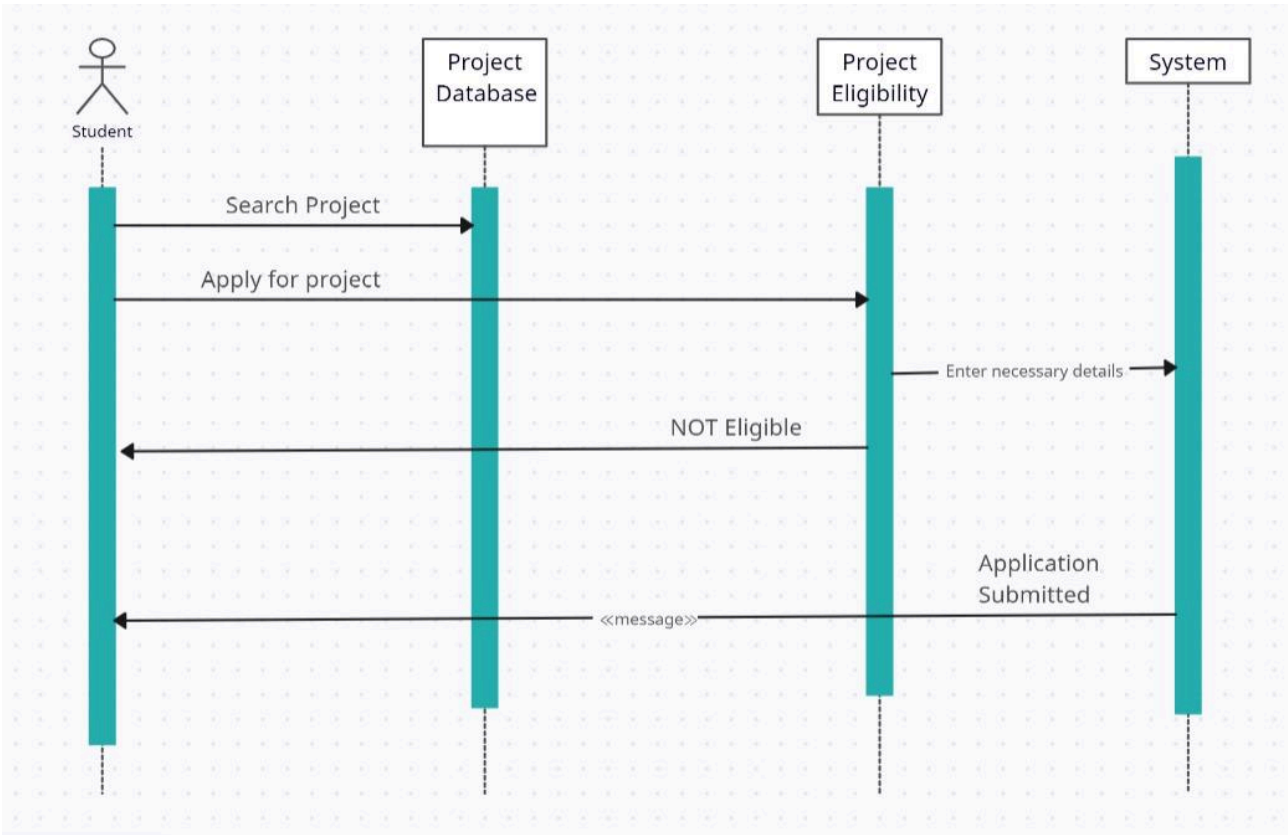
2. REGISTER



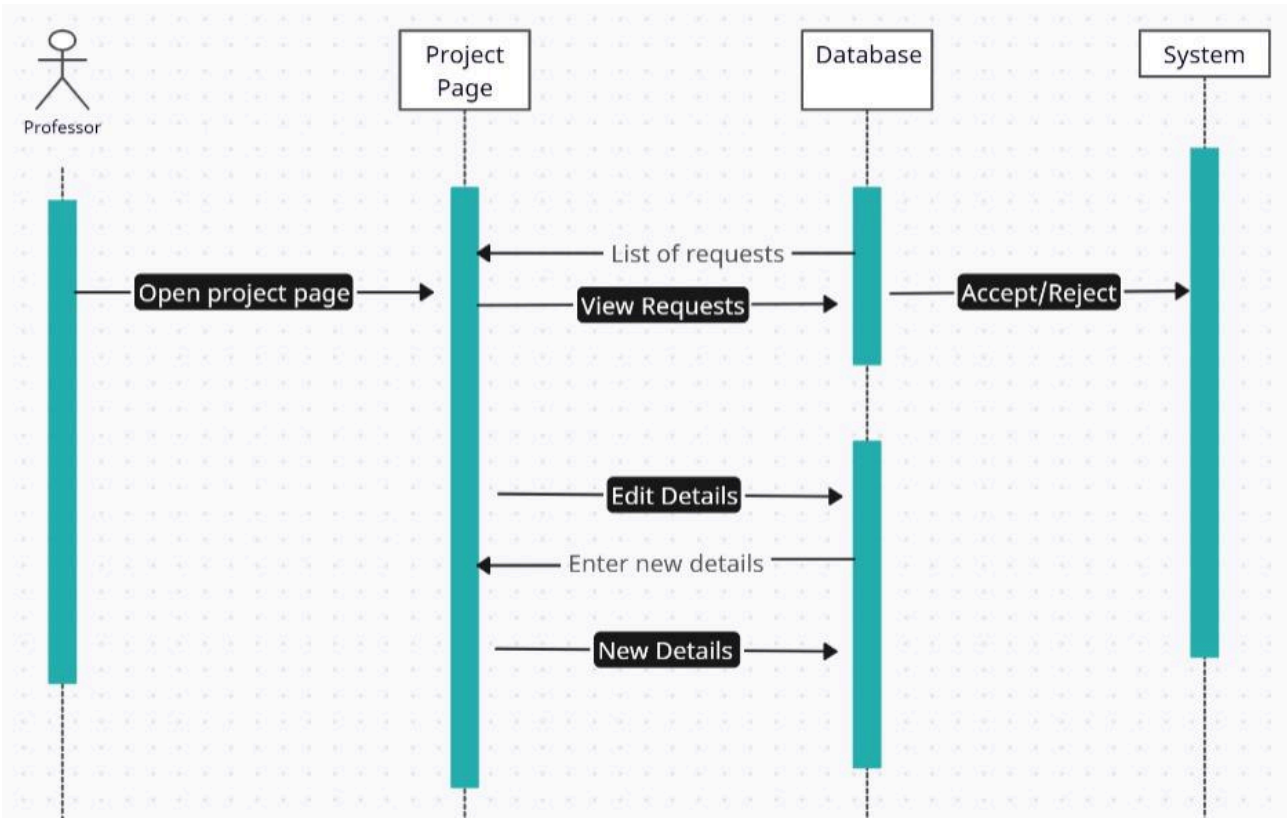
3. FORGOT PASSWORD



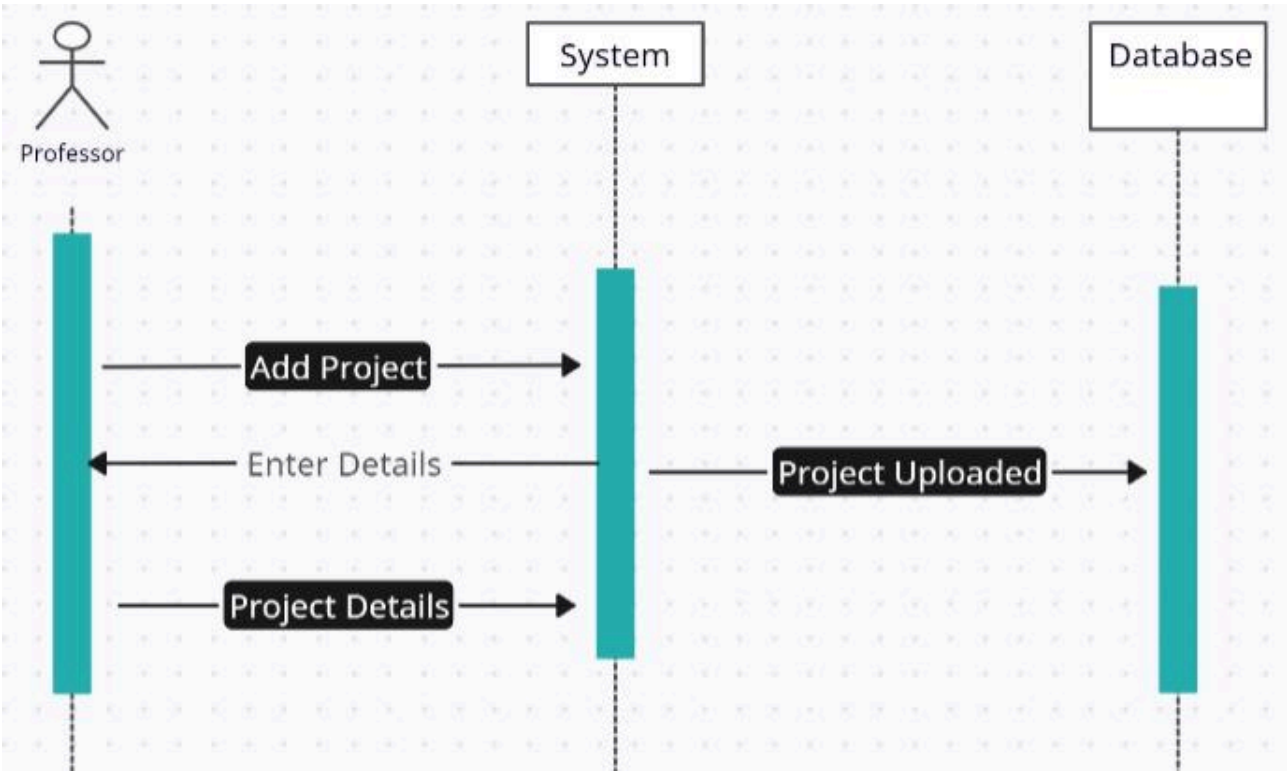
4. SEARCH AND APPLY FOR PROJECT



5.VIEW REQUEST AND EDIT PROJECT DETAILS

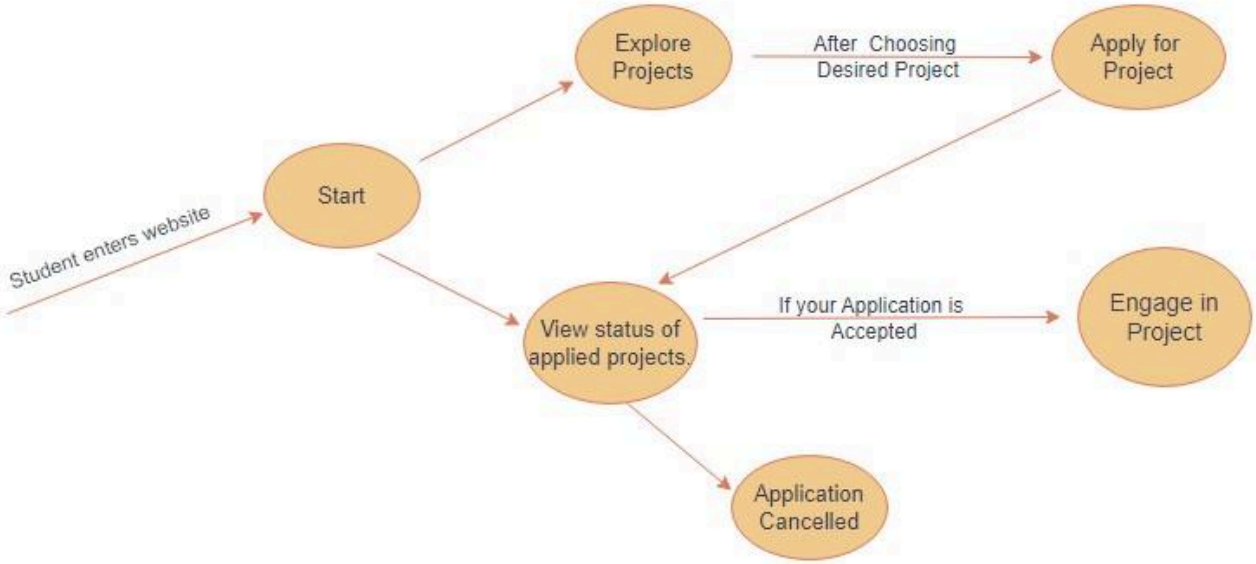


6. ADD PROJECT

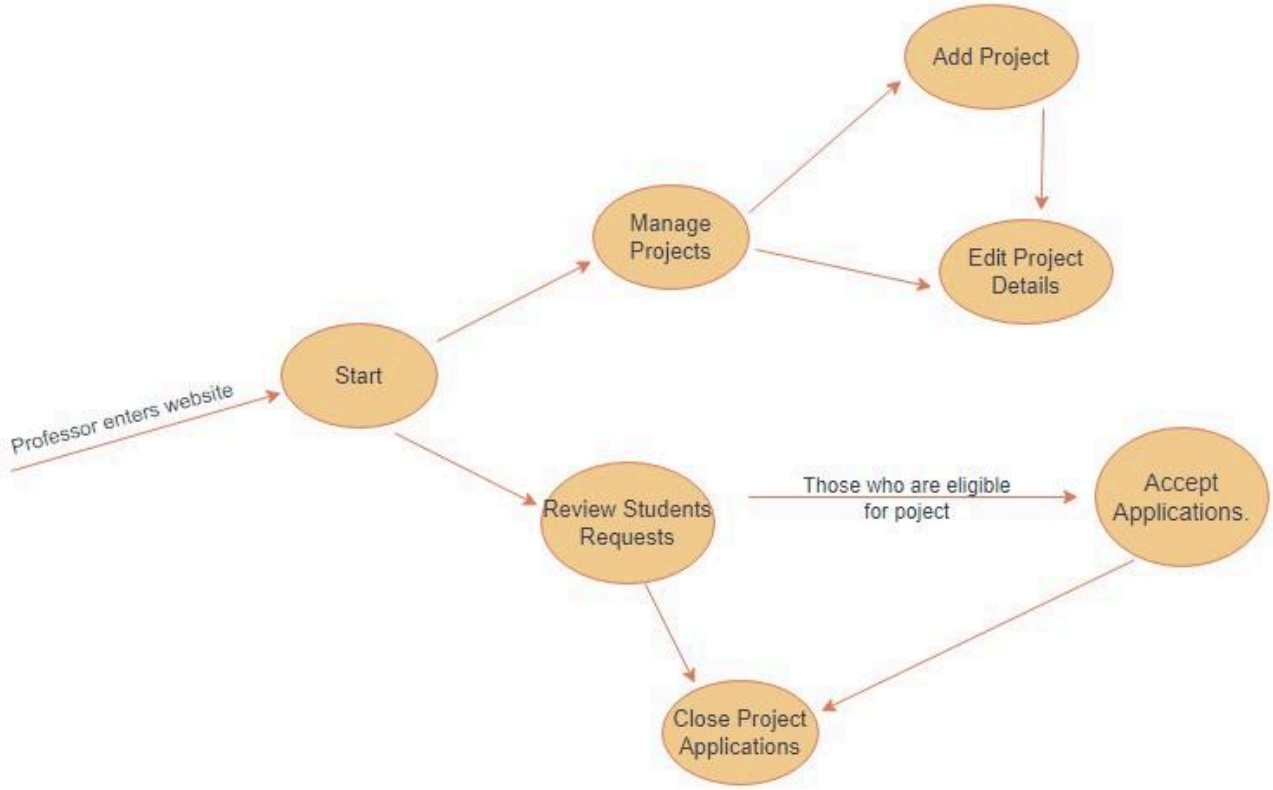


3.4 State Diagrams

Student's State Diagram :



Professor's State Diagram :



4 Project Plan

Timeline of the development of ProfInfo Central :-

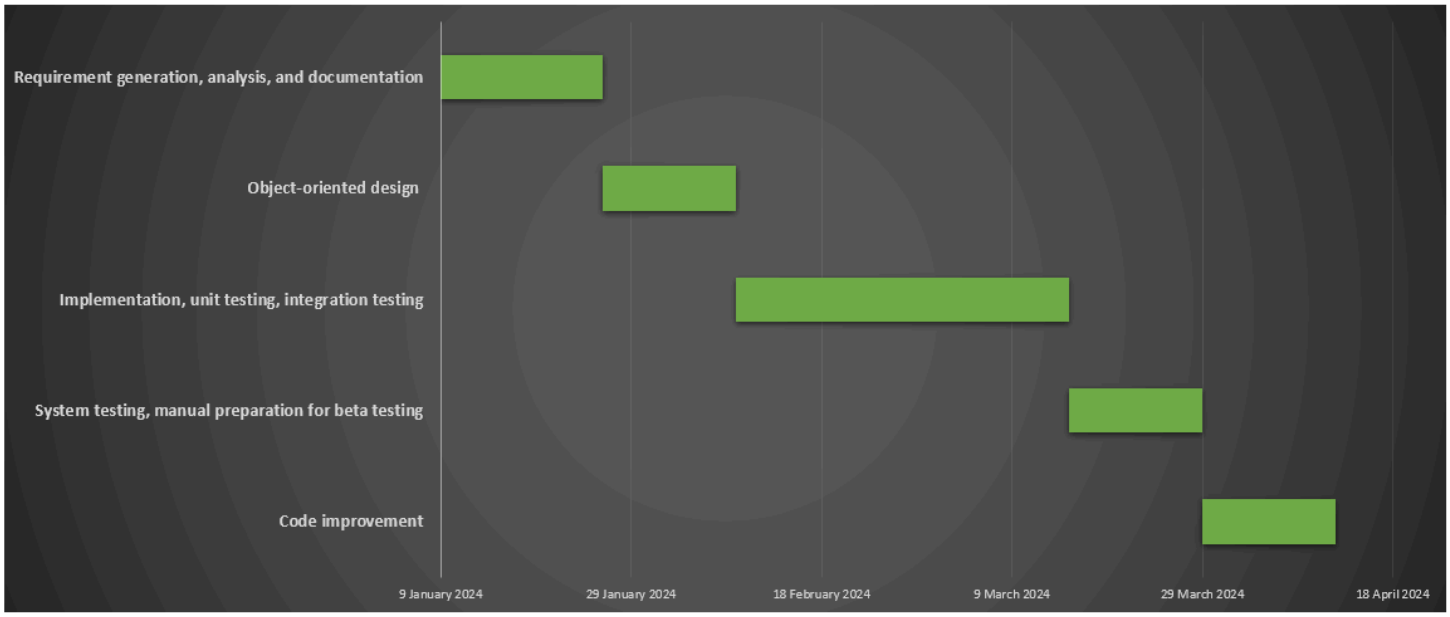
1. 9th January 2022 - 26th January 2022: Documentation of Software Requirements Specification (SRS).
2. 27nd January 2022 - 9th February 2022: Documentation of Software Design Documentation.

3. 9th February 2022 - 15th March 2022: During the implementation and testing phase, the project is methodically segmented into tasks, each assigned to team members based on their expertise. Key activities encompass writing source code, designing user-friendly web pages, and developing/testing both unit and integrated cases. Implementing a secure firewall is a priority for protection against unauthorized access. Testing is conducted by function-specific members, ensuring comprehensive examination within each domain of expertise. Overall testing includes performance, security, and user acceptance, validating the system's functionality. Documentation of the testing process, including test cases and results, is crucial for future reference. This structured approach ensures an efficient progression towards a high-quality, secure final product within the given timeframe.

4. 16th March 2022 - 29th March 2022: System and Manual Beta Testing:- Further necessary testing would be done during this week. This would include testing the system as a developer as well as (Beta) testing as an End User.

5. 30th March 2022 - 12th April 2022: Code Improvement and Beta Testing:- With the results of the previous testing sessions, improvements on the software would be done to increase speed, security and its overall quality.

6. 12th April 2022 - 19nd April 2022: Addressing Beta Testing Feedback and Delivering the Final Project Report.



The following are the broad classification of the tasks along with the names of team member responsible for it:-

1. Front End Development:

The front end of a website refers to the user interface and experience that visitors interact with directly. It encompasses the design, layout, and presentation of content, including text, images, and multimedia elements. Front-end development involves utilizing technologies like HTML, CSS, and JavaScript to create responsive, visually appealing, and user-friendly websites. It focuses on optimizing the website's appearance across various devices and browsers, ensuring seamless navigation and an engaging experience for users. Front-end developers collaborate with designers to bring the visual aspects to life, enhancing accessibility and functionality to deliver a compelling and intuitive web interface.

The members responsible for Front End Development are:-
a. Nilesh , Atharv , Kuldeep and Prabhat : - (Student Interfaces: Explore Projects, Project Details, Project Requesting)
b. Jaswanth , Pavani and Lakshyta (Faculty Interfaces: Projects Page, Project Details, Users Details)
c. Kartik, Sontam Deekshitha, Mohd Nasar Siddqui (Interfaces: Log-in, Signup, Faculties (Department Wise) , Contact us)

The following software might be useful for Front End Development:-

a. React JS:- React.js is a JavaScript library for building user interfaces, widely used for creating dynamic and efficient web applications. Developed and maintained by Facebook, React employs a component-based structure, facilitating modular and reusable code. Its declarative approach simplifies UI development, while a virtual DOM ensures optimal rendering performance. React is popular for its scalability, ease of maintenance, and extensive community support.

b. JavaScript:- JavaScript in frontend development enhances user interfaces by enabling dynamic content, interactivity, and asynchronous communication. It's essential for building responsive and interactive web applications, ensuring seamless user experiences across various devices and browsers.

c. HTML:- A standard markup language for documents designed to be displayed in a web browser.

d. CSS:- Cascading Style Sheets is a style sheet language used for describing the presentation of a document written in a markup language such as HTML.

2. Back End Development:-

It is everything that the users don't see and contains behind-the scenes activities that occur when performing any action on a website. It focuses primarily on databases, backend logic, APIs, and Servers. In the MERN (MongoDB, Express.js, React, Node.js) stack, backend development involves Node.js for server-side JavaScript, Express.js for building APIs and handling HTTP

requests, and MongoDB as the NoSQL database. Backend developers using MERN manage data storage, retrieval, and server-side logic. Node.js enables JavaScript execution on the server, Express.js simplifies API development, and MongoDB stores data in a flexible, JSON-like format. The backend communicates with the frontend (built with React) to create dynamic and responsive web applications. The MERN stack provides an end-to-end JavaScript solution, streamlining development and ensuring a cohesive ecosystem for building scalable and efficient web applications.

The members responsible for Back End Development are:-

a. Mohd Nasar Siddqui, Sontam Deekshitha, Kartik (Databases, Authentication and Api)

The following software might be useful for Back End Development:-
a. JavaScript:- JavaScript is a versatile scripting language primarily used for web development. It runs in browsers, enabling dynamic content, interactivity, and asynchronous communication. It is essential for front-end and back-end web development.

c. Node.JS: Node.js is an open-source, cross-platform, back-end JavaScript runtime environment that runs on the V8 engine and executes JavaScript code outside a web browser.

b. Express.JS:- Express.js is a minimalist and flexible Node.js web application framework. It simplifies server-side development, providing a robust set of features for building web and mobile applications with ease.

d. MongoDB: MongoDB is a free and open-source document-oriented database that is very much popular among web developers.

5 Other Details

<This section is Optional. Please provide any other details that are suitable for being included in the design document .>

Appendix A - Group Log

S. No.	Date	Time	Venue	Description
1	28/01/24	7:00 pm to 8:30 pm	RM	All gathered together and planned the rough sketch designs of webpages.
2	30/01/24	3:00 pm to 4:00 pm	RM	Divided the Documentation work into 3 parts(i.e. user interface,student interface and professor interface) and planned a timeline on how to complete each part.
3	01/02/24	9:30 pm to 11:00 pm	RM	All gathered and Designed User Interface webpages and finalised the design of User interface.
4	01/02/24	5:00 pm to 7:00 pm	RM	Designed Students Interface webpages and finalised the design of Student Interface.
5	02/02/24	4:00 pm to 5:00 pm	RM	Designed Professor's Interface webpages and finalised the design of Professor Interface.
6	03/02/24	4:00 pm to 6:00 pm	RM	Completed the first part of the documentation by inserting the designed web pages into the document.
7	04/02/24	2:00 pm to 5:00 pm	RM	Completed the 2nd and 4th part of Documentation.
8	06/02/24	2:00 pm to 6:00 pm	RM	Sketched the Use case Diagrams and completed the 3rd part of Documentation.
9	07/02/24	7:00 pm to 8:00 pm	RM	Revised the entire documentation and checked if there are any errors or modifications and made it.
10	08/02/24	6:00 pm to 7:00 pm	Zoom	Completed the Design Documentation and Submitted.