PROJECTURE

UCS503 Software Engineering Project Report Mid-Semester Evaluation

Submitted by:

(101915027) Aman Srivastav (Offline)

(101915029) Alisha Mathur (Offline)

(101915032) Sparsh (Offline)

(101915045) Ekroop Kaur (Offline)

BE Third Year, ENC

Group No: 5

Submitted to:

Mr. Punit Kumar



Electronics and Communication Department TIET, Patiala

October 2021

**TABLE OF CONTENTS**

|  |  |  |
| --- | --- | --- |
| **S.No.** | **Assignment** | **Page No.** |
| **1.** | **Project Selection Phase** | 3-7 |
| 1.1 | Software Bid | 3-6 |
| 1.2 | Project Overview | 7 |
| **2.** | **Analysis Phase** | 8-39 |
| 2.1 | Use Cases | 8-16 |
| 2.1.1 | Use-Case Diagrams | 8 |
| 2.1.2 | Use Case Templates | 9-16 |
| 2.2 | Activity diagram and Swimlane Diagrams | 17-26 |
| 2.3 | Data Flow Diagrams (DFDs) | 27-33 |
| 2.3.1 | DFD Level 0 | 27 |
| 2.3.2 | DFD Level 1 | 28 |
| 2.3.2 | DFD Level 2 | 29-30 |
| 2.4 | Software Requirement Specification in IEEE Format 31-38 | |
| **3.** | **Design Phase** | 39-47 |
| 3.1 | Class Diagram | 39 |
| 3.2 | Sequence Diagram | 40-42 |
| 3.3 | Collaboration Diagram | 43-44 |
| 3.4 | Database Design: ER Diagram | 45 |
| **4.** | **Prototyping-GUI** | 46-47 |
| 4.1 | Screen Shots | 46-47 |

1. PROJECT SELECTION PHASE

1.1 SOFTWARE BID

Group: 5 Dated: 29/08/2021

# Team Name: Projecture

**Team ID (will be assigned by Instructor):**

Please enter the names of your Preferred Team Members.

* You are required to form **a three-to-four-person** team.
* Choose your team members wisely. You will not be allowed to change teams.

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Roll No** | **Project Experience** | **Programming Language used** |
| Aman Srivastav | 101915027 | E-commerce website | C/C++, JavaScript. |
| Alisha Mathur | 101915029 | Sudoku Game | C/C++, Python. |
| Sparsh | 101915032 | Website Landing Page | C/C++, JavaScript, Java, SQL. |
| Ekroop Kaur | 101915045 | Tic Tac Toe | C/C++, JavaScript. |

# Programming Language/Environment Experience: -

List the languages you are most comfortable developing in, **as a team**, in your order of preference. Many of the projects involve Java or C/C++ programming.

1. JavaScript
2. SQL

# Choices of Projects:

Please select **4 projects** your team would like to work on, by order of preference:

|  |  |
| --- | --- |
| **First Choice** | **Projecture**  **Motivation:**  LinkedIn is for working professionals, but we as students always wanted to collaborate/work together on projects and bring ideas come to reality. It is quite evident that projects play a vital role in developing the Tech skills of students. A lot of students are unable to accomplish completing a project either because they cannot find people with the relevant skill-set or they have the skills but have no idea in mind. Even if they have the above requirements, they might simply lack the guidance from the Mentor. “**Projecture”** aims at helping the students by providing a website that can cater to all these needs and act as a bridge between students and mentors or between students and students. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to a productive use.  **Feasibility:**  Our project is technically feasible as all the requirements for building the software are available and is economically feasible too. The operational feasibility is justified in the fact that there is no preexisting market software for the same. |
| **Second Choice** | **Thapar Drop**  **Motivation:**  Every year students have a lot of things that they want to donate/can provide things that can be re-used but are unable to find the people to donate to. “**Thapar Drop**” will help them finding people who are in need of items that they want to give away by connecting them to the local NGO’s or donation camps.  **Feasibility:**  This project is feasible given the fact that there is no such software preexisting. Also, for the users the website is feasible to use as internet connection is available all across the campus. |
| **Third Choice** | **Laundry Management System**  **Motivation:**  During the laundry collection there is a large crowd near laundry drop room. So in order to overcome the challenges of Hostel laundry we will be establishing an automated environment for laundry operations. This will include taking orders, scheduling delivery & pickup, and managing payment. Our prime aim is to increase operational efficiency, improve productivity and to save time and effort.  **Feasibility:**  This project is viable because there is no dedicated software for the problem stated. Also for the users the website is feasible to use because of fair internet connection in campus and hence saves time. |
|  |  |
| **Fourth Choice** | **Parking Management System**  **Motivation:**  Finding a parking spot tends to be frustrating and time-consuming for students and staff which ideally doesn’t have to be. With technology i.e. through our website, parking spaces will be well managed and coordinated. These benefits will extend over the entire parking capacity of Campus. If we follow the best parking practices, we can make the parking experience seamless and stress-free for everyone  **Feasibility:**  This project is technically so feasible given the fact that there is no such software in the vicinity around. Hardware required i.e. Sensor and barcode’s cost is economical. Observations will continuously be updated and users can access it any time anywhere. |

# Additional Remarks/Inputs

Please tell us about any other factors that we should take into consideration.

We have researched about the project and have a clear idea for the need, the basics

Required to make it a fully functioning project and believe that it will indeed be very helpful for budding Engineers. An effortless collaboration and communication between faculties, staff, students, departments in a central web-based project management system. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to a productive use. We also believe that we will be able to create this project in the stipulated time and hence request you to kindly allow us to work on this project

* 1. PROJECT OVERVIEW

NOTE: -This is a working document and, as such, is subject to change. In its initial form, it is incomplete by definition, and will require continuing refinement. Requirements may be modified and additional requirements may be added as development progresses and the system description becomes more refined. This information will serve as a framework for the current definition and future evolution of the University Academic Portal.

LinkedIn is for working professionals, but we as students always wanted a platform to collaborate/work together on projects and bring ideas come to reality. It is quite evident that projects play a vital role in developing the technical skills of students. A lot of students are unable to accomplish completing a project due to several reasons. **“Projecture”** aims at helping the students by providing a website that can cater to all the needs and act as a bridge between students and mentors or between students and students. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to productive use. An effortless collaboration and communication between faculties, staff, students, departments in a central web-based project management system. Accordingly, this software will be used by many different classes of users:

* Mentors
* Students (creators, consumers)
* Working professionals
* Reviewer/moderator

1. **ANALYSIS PHASE**

**2.1 USE CASE**

**2.1.1 USE-CASE DIAGRAMS**

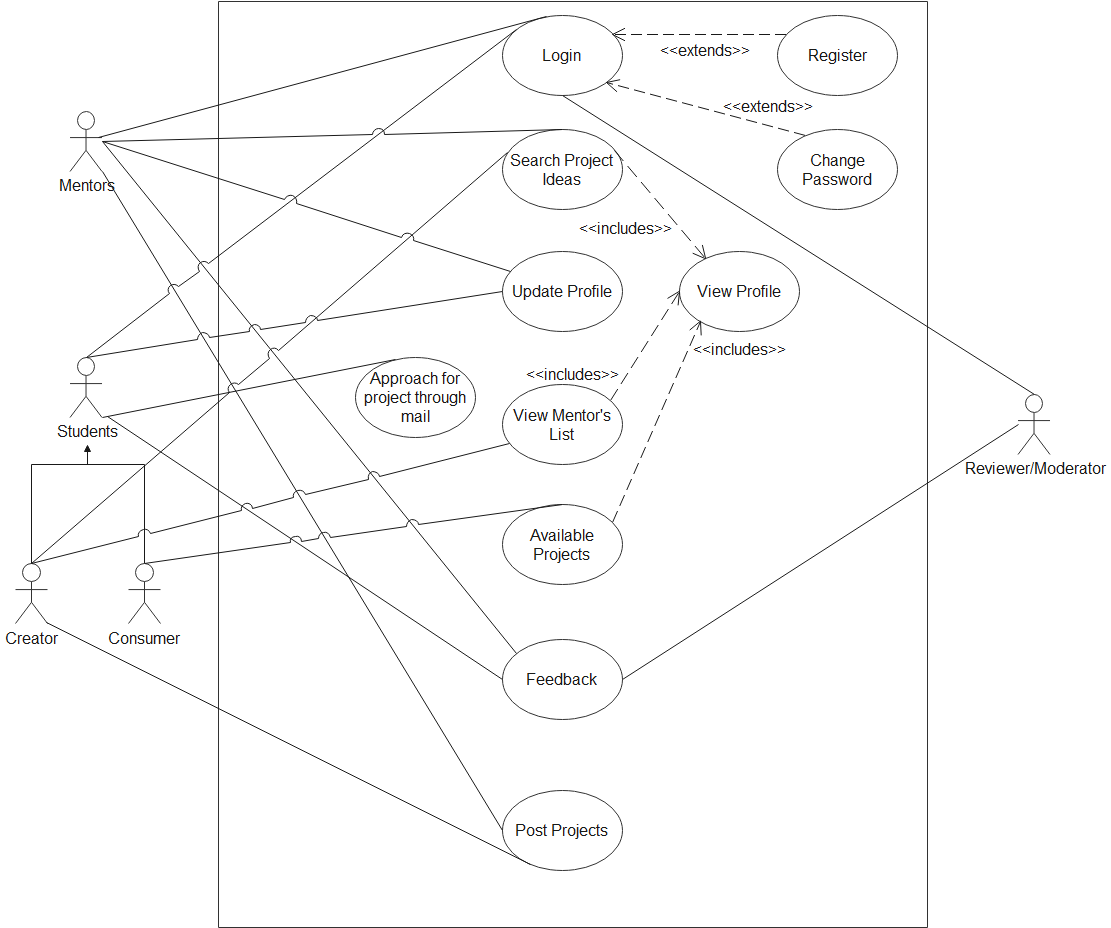


Fig1: Use case diagram

A use case diagram is a graphical depiction of a user's possible interactions with a system. A use case diagram shows various use cases and different types of users the system has and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. The actors are often shown as stick figures. Considering Use case Diagram for 'Projecture' there are three actors viz Mentor, Reviewer/Moderator, Student (Creator, Consumer). They interact via use cases like login, search project ideas, update profile, view mentor list, available projects, feedback, post projects.

**2.1.2 USE CASE TEMPLATE**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | login |
| 2.ABBREVIATED TITLE | login |
| 3.USE CASE ID | 1 |
| 4.ACTORS | mentor, student (creator, consumer), reviewer/moderator. |
| 5.DESCRIPTION | a user login to the system to access the functionality of the  system. |
| 5.1. PRE-CONDITIONS: | system must be connected to the  network. |
| 5.2. TASK-SEQUENCE | 1. login page will be displayed to the user. 2. if the user is a registered user, then he/she will enter his/her username and password and click “login”. 3. if not, then the user will first have to “register” in the site. 4. a new account is created for the user. 5. the user creates a new username and password. 6. the user information is stored in the database. 7. if the user is a registered user, but forgot password then he/she can “change password.” |
| 5.3. POST- CONDITIONS: | after a successful login, an email is sent to the user’s entered mail id, which upon clicking opens the  main page of the site. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**Use-Case Template for Searching for Project ideas to work on**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | search for project idea to work on |
| 2.ABBREVIATED TITILE | search for project ideas |
| 3.USE CASE ID | 2 |
| 4.ACTORS | mentors, students |
| 5.DESCRIPTION | this functionality can only be accessed by the user registered under the category of “mentor or student” wherein he/she will be able to search the project idea’ for a particular category, with the category name being typed in the search box. on clicking “view projects”, the projects of that genre will be displayed to start with. |
| 5.1. PRE-CONDITIONS: | user must be logged in to the site  under the category of “mentor or student”. |
| 5.2. TASK-SEQUENCE | 1. user clicks on the “search” bar 2. user enters the category of the project which she/he wants to work on. 3. then, he/she presses the “view profile” button. 4. a list of profiles whose subject matches with the entered category is shown in a grid form. 5. to view the complete profile   “detailed view” button  can be clicked. |
| 5.3. POST-CONDITIONS: | user shall be able to see all the projects of the entered  category in a grid form. |
| 6. MODIFICATION HISTORY |  |
| 7.AUTHOR: |  |

**Use-Case Template for Updating of Profile**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | update profile |
| 2.ABBREVIATED TITILE | update profile |
| 3.USE CASE ID | 3 |
| 4.ACTORS | mentor, student |
| 5.DESCRIPTION | a user logged in to the site can edit his/her profile by changing username, or email-id or phone number or any of the other parameters mentioned on the edit  profile page. |
| 5.1. PRE-CONDITIONS: | user must be logged in to the site |
| 5.2. TASK-SEQUENCE | 1. user clicks on the “edit profile”   button   1. user is directed to the edit profile page. 2. of all the edit parameters available, user chooses the part that he/she wants to change. 3. after user finishes his edit   process, he/she clicks on the “save changes” button. |
| 5.3. POST-CONDITIONS: | user’s profile is updated, showing  the changes that he/she has made in his/her profile |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**Use-Case Template for Approach for project through mail**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | approach for project through mail |
| 2.ABBREVIATED TITILE | approach for project through mail |
| 3.USE CASE ID | 4 |
| 4.ACTORS | student |
| 5.DESCRIPTION | student can contact through mail amongst themselves and  with mentors. |
| 5.1. PRE-CONDITIONS: | user must be logged in to the site and can obtain the contact  details only if they match with a certain profile. |
| 5.2. TASK-SEQUENCE: | 1. user clicks on a contact, email address of that profile will be visible. 2. on clicking the email, user will be redirected to gmail account for profile of the matched user. |
| 5.3. POST-CONDITIONS: | user who searched is shown the  email address of the selected user. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**Use-Case Template View Mentor’s List**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | view mentor’s list |
| 2.ABBREVIATED TITILE | view mentor’s list |
| 3.USE CASE ID | 5 |
| 4.ACTORS | student (creator) |
| 5.DESCRIPTION | a search is initiated by the student (creator). requirements for the desired mentor are entered. |
| 5.1PRE-CONDITIONS: | system must be connected to the network. |
| 5.2 TASK-SEQUENCE | 1. the student(creator) would click on   the “view mentor list”.   1. a window would open prompting the student(creator) to enter their requirements. 2. the inputted data is then published. 3. the data is passed on to the matching algorithm. 4. the mentor’s profiles are   displayed and more information can be fetched by clicking “view profile” tab |
| 5.3POST-CONDITIONS: | after a successful mentors’ details are entered, data is fed into the matching algorithm and results are displayed. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**Use-Case Template for Available Projects**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | available projects |
| 2.ABBREVIATED TITILE | available projects |
| 3.USE CASE ID | 6 |
| 4.ACTORS | student (consumer) |
| 5.DESCRIPTION | it is a functionality available to  the student(consumer) to look for the available projects |
| 5.1PRE-CONDITIONS: | user must be logged in to the site |
| 5.2TASK-SEQUENCE | 1. the student(consumer) clicks on the “available projects” tab. 2. the results of all the available projects are displayed. 3. the student(consumer) would be able to see all the available projects in the drop-down menu. 4. in addition student(creator) can also “view profile”of the mentor are other student(creators). |
| 5.3POST-CONDITIONS: | all available projects are displayed. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**Use-Case Template for Feedback**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | feedback |
| 2.ABBREVIATED TITILE | feedback |
| 3.USE CASE ID | 7 |
| 4.ACTORS | mentor, student |
| 5.DESCRIPTION | a feedback is given by mentor and student, and details remain anonymous and admin should be able to comment on the feedback |
| 5.1PRE-CONDITIONS: | system must be connected to the  network. users should be logged in. |
| 5.2TASK-SEQUENCE | 1. the mentor / student would click on the “feedback”.  2. a window would open prompting the please give your feedback.  3. the inputted data is then sent to the moderator/ reviewer.  4. moderator/ reviewer should give a comment to the employee. |
| 5.3POST-CONDITIONS: | after getting the proper resolve  feedback and the comment is saved in a database. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

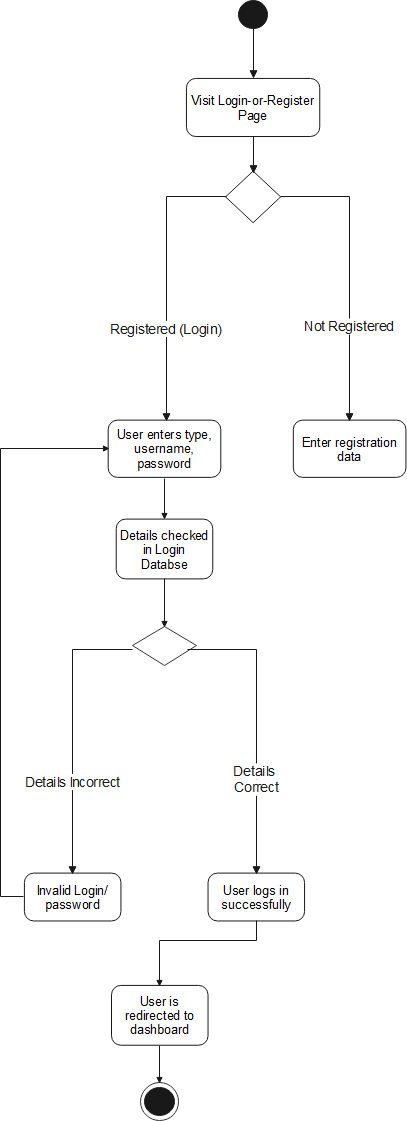
**Use-Case Template for Post Projects**

|  |  |
| --- | --- |
| 1. USE CASE TITLE | post projects |
| 2.ABBREVIATED TITILE | post projects |
| 3.USE CASE ID | 8 |
| 4.ACTORS | mentors, student(creator) |
| 5.DESCRIPTION | this functionality can only be accessed by the user registered under the category of “mentor or student(creator)” wherein he/she will be able to post the project’ for a particular category, with proper description. |
| 5.1. PRE-CONDITIONS: | user must be logged in to the site  under the category of “mentor or student(creator)”. |
| 5.2. TASK-SEQUENCE | 1. user clicks on the “post project” bar 2. user enters the category of the project which she/he wants to initiate. 3. then, he/she presses the “post project” button. |
| 5.3. POST-CONDITIONS: | user shall be able to see all the details of project in a grid form. |
| 6. MODIFICATION HISTORY: |  |
| 7.AUTHOR: |  |

**2.2 ACTIVITY DIAGRAM AND SWIMLANE DIAGRAM**

* + 1. **ACTIVITY DIGRAM**

1. **Login/Register**



Click

Confirmation

Fig2: Login/Register use case diagram

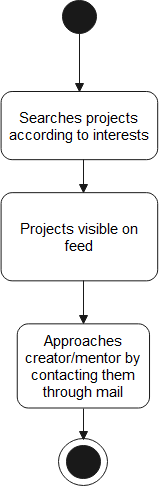
1. **Available projects**

Fig3: Available projects use case diagram

1. **Feedback**

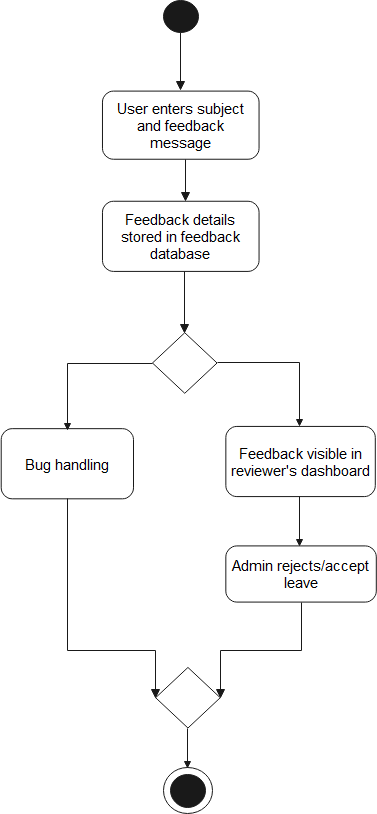


Fig4: Feedback use case diagram

1. **Update profile**

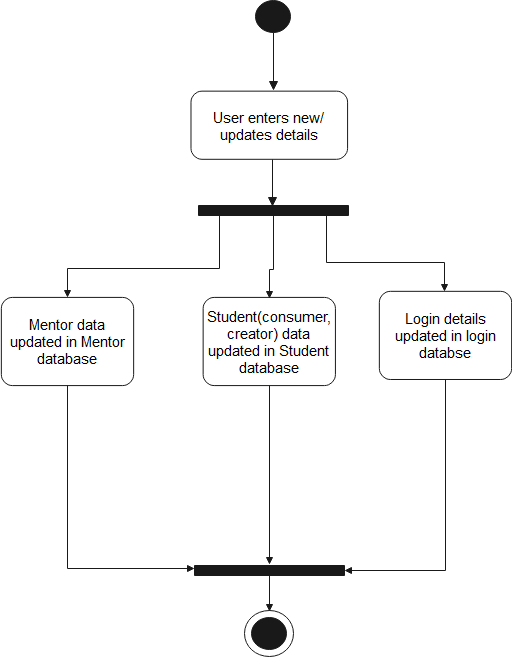


Fig5: Update profile use case diagram

1. **View Mentor’s list**

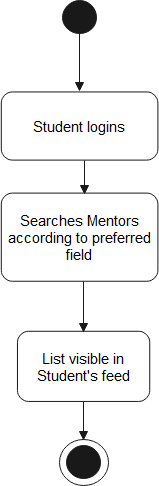


Fig6: View Mentor’s list use case diagram

1. **Post projects**

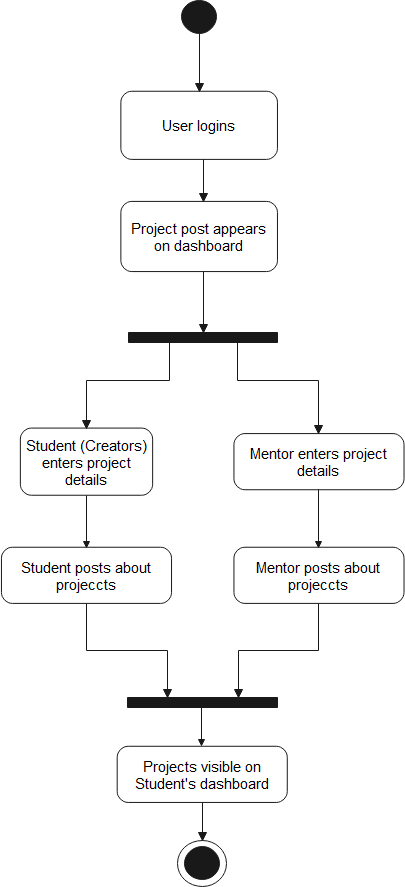


Fig7: Post Projects use case diagram

1. **Search Projects ideas**

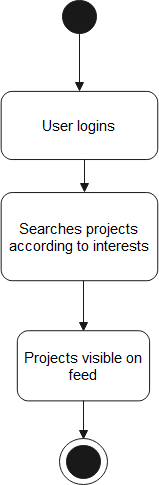


Fig8: Search Project Ideas use case diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency. In the Unified Modeling Language, activity diagrams are intended to model both computational and organizational processes, as well as the data flows intersecting with the related activities. Although activity diagrams primarily show the overall flow of control, they can also include elements showing the flow of data between activities through one or more data stores. Therefore, to formulate Activity Diagram For 'Projecture' activity diagram for all use cases have been made.

* + 1. **SWIMLANE DIGRAM**

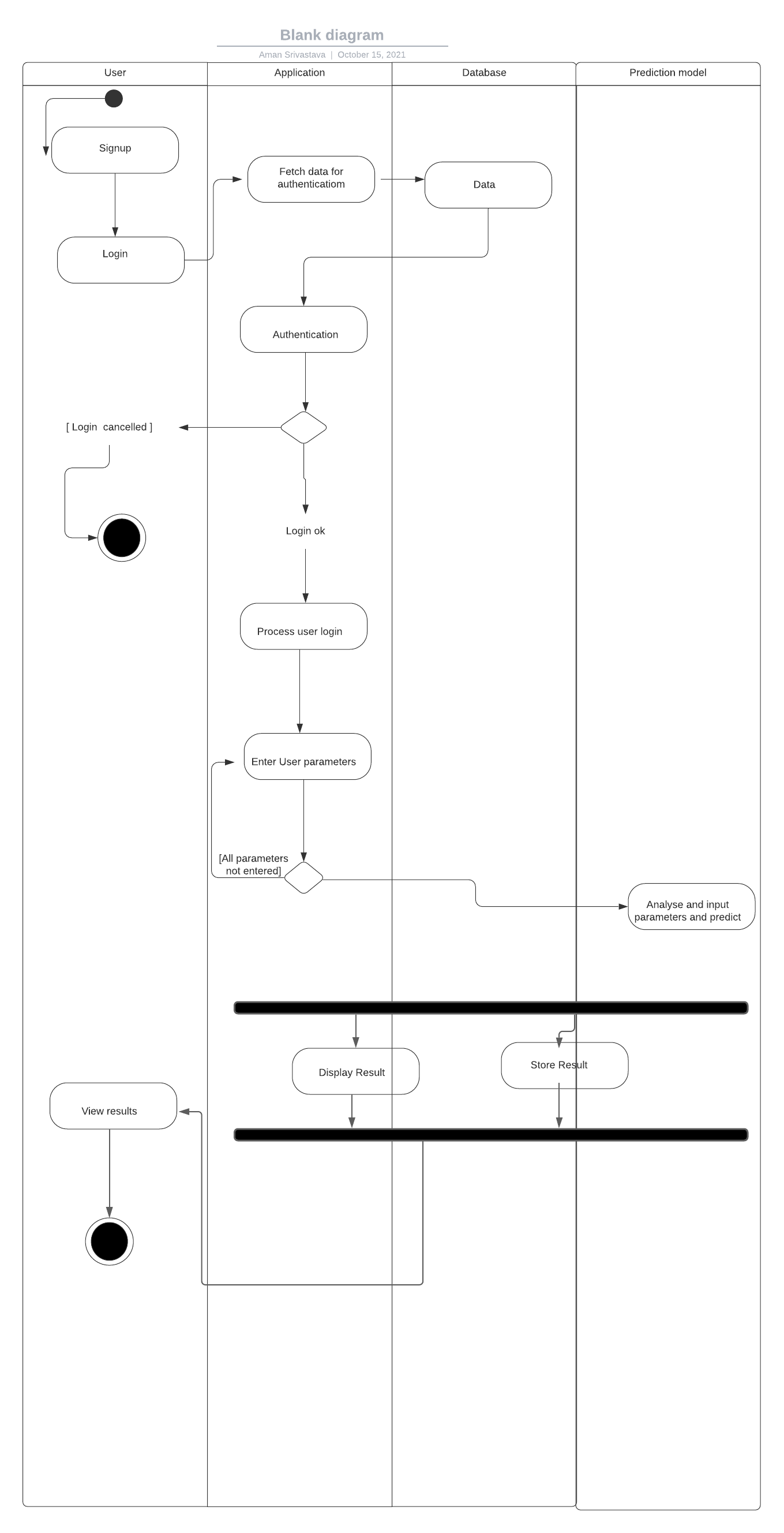


Fig9: Swimlane diagram

A swimlane is used in process flow diagrams, or flowcharts, that visually distinguishes job sharing and responsibilities for sub-processes of a business process. They are flowcharts that show both a process from start to finish and who is responsible for each step in the process

**2.3 DATA FLOW DIAGRAMS (DFDS)**

**2.3.1 DFD Level 0**

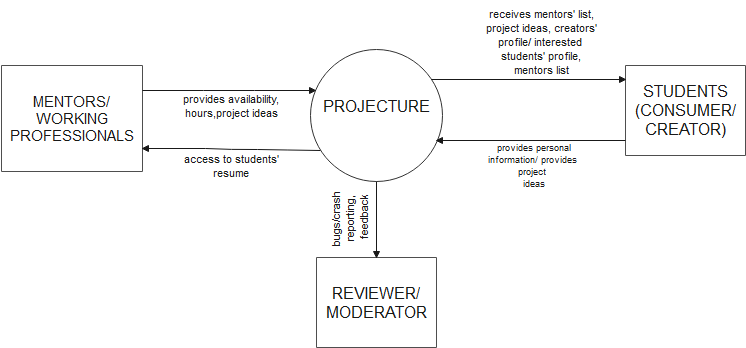


Fig10: DFD Level 0

**2.3.2 DFD Level 1**

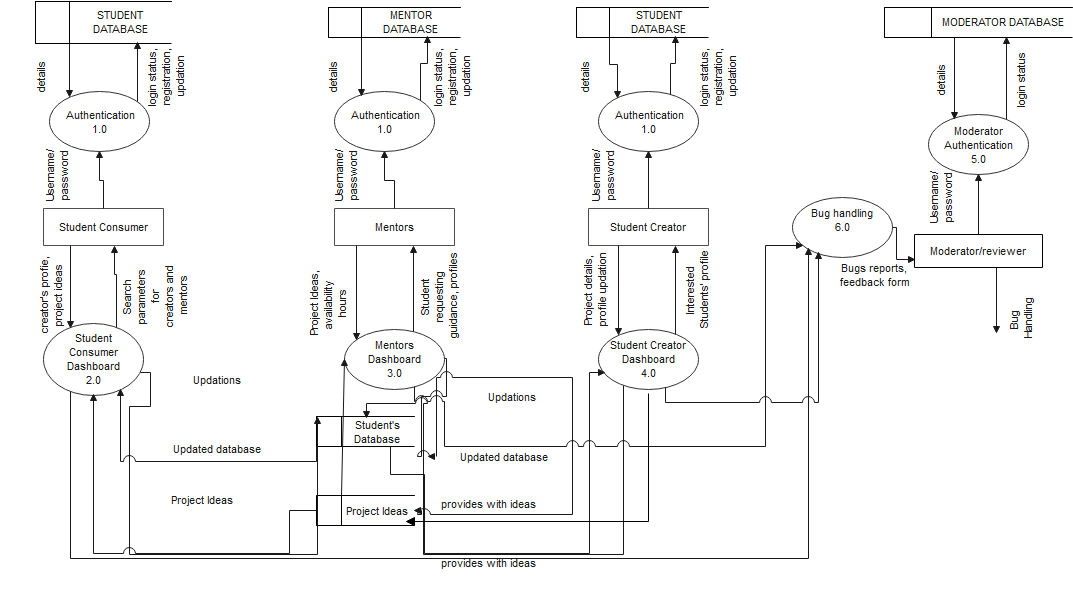


Fig11: DFD Level 1

**2.3.3 DFD Level 2**

**2.3.3.1 Login/Registration Functionality (Authentication) DFD**

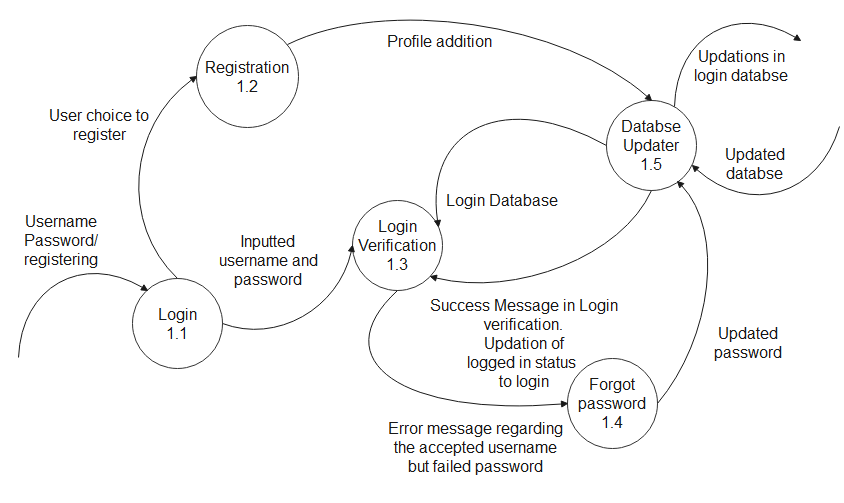


Fig12: Login/Registration Functionality (Authentication)

**2.3.3.2 Student (Consumer) Functionality DFD**

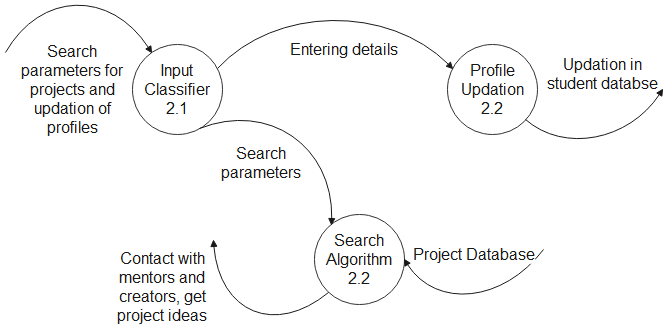


Fig13: Student (Consumer) Functionality DFD

**2.3.3.3 DFD Mentor Functionality**

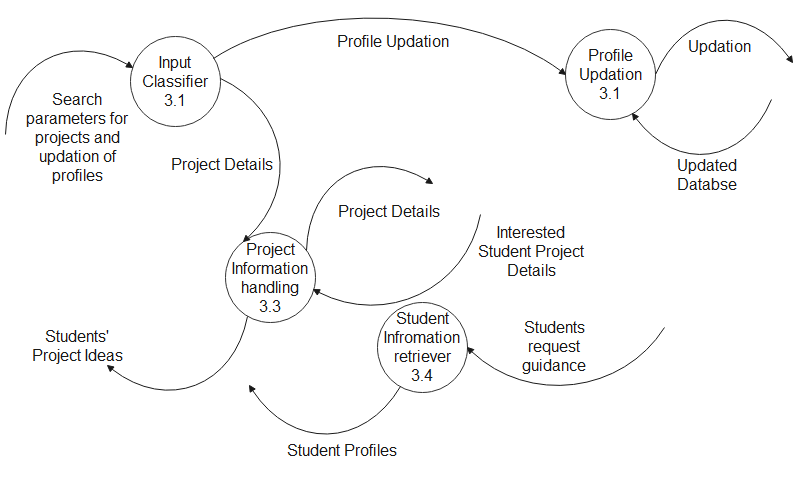


Fig14: DFD Mentor Functionality

**2.3.3.4 Student (Creator) Functionality**

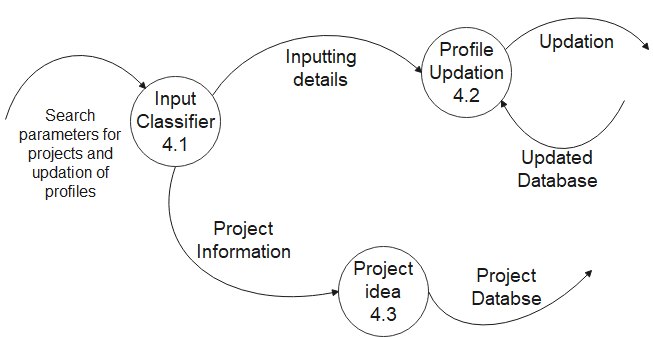


Fig15: Student (Creator) Functionality DFD

A data flow diagram is a graphical representation of flow of data which illustrates how the data is processed by a system in terms of inputs and outputs. As its name indicates its focus is on the flow of information, where data comes from, where it goes and how it gets stored. A new user must register to use the features of the discussion forum. Once the user registers, a login page displays where he has to login which leads to the home page. For our project we have made level-0, level-1, level-2 DFD's. From a broader viewpoint, the data flow mainly takes place between mentors, student(creators), student(consumer), admin.

**2.4 Software Requirement Specifications in IEEE format**

# Software Requirements

**Specification**

**For**

# Projecture

### Version 1.0 approved

Prepared by**:Aman Srivastav,**

**Alisha Mathur,**

**Ekroop Kaur,**

**Sparsh**

**Thapar Institute of Engineering and Technology, Patiala**

### **Oct 14, 2021**

***Copyright © 1999 by Karl E. Wiegers. Permission is granted to use, modify, and distribute this document.***

**Table of Contents**

[Table of Contents ii](#_TOC_250021)

1. [Introduction](#_TOC_250020) 35-36
   1. [Purpose](#_TOC_250019) 35
   2. [Document Conventions](#_TOC_250018) 35
   3. [Intended Audience and Reading Suggestions](#_TOC_250017) 35
   4. [Product Scope](#_TOC_250016) 35
   5. [References](#_TOC_250015) 35-36
2. [Overall Description](#_TOC_250014) 36-37
   1. [Product Perspective](#_TOC_250013) 36
   2. [Product Functions](#_TOC_250012) 36
   3. [User Classes and Characteristics](#_TOC_250011) 36-37
   4. [Operating Environment](#_TOC_250010) 37
   5. [Design and Implementation Constraints](#_TOC_250009) 37
   6. [User Documentation](#_TOC_250008) 37
3. System Features………………………………………………………………………37-38
   1. System Features [3](#_TOC_250006)7
   2. [Functional Requirements 3](#_TOC_250005)8
4. External Interface Requirements………………………………………………………38
   1. [User Interfaces](#_TOC_250001) 38
   2. Hardware Interfaces……………………………………………………………………………38
   3. Software Interfaces……………………………………………………………………………38
   4. Communication Interfaces…………………………………………………………………….38
5. Other Nonfunctional Requirements 39
   1. Performance …………………………39
   2. Security 39
   3. Maintainability 39
   4. Availability……………………………………………………………………………………39

## **INTRODUCTION**

### **Purpose**

This software being developed by our team is for *Projecture*. It aims at helping the students by providing a website that allows them to easily collaborate and can act as a bridge between students and mentors or between students and students. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to a productive use.

### **Document Conventions**

This document follows IEEE specified format for writing Software Requirements Specification. The features specified further in this document are arranged in order of their priorities and are described in paragraphs.

### **Intended Audience and Reading Suggestions**

* Students
* Mentors

### **Product Scope**

LinkedIn is for working professionals, but we as students always wanted to collaborate/work together on projects and bring ideas come to reality. It is quite evident that projects play a vital role in developing the Tech skills of students. A lot of students are unable to accomplish completing a project either because they cannot find people with the relevant skill-set or they have the skills but have no idea in mind. Even if they have the above requirements, they might simply lack the guidance from the Mentor. It aims at helping the students by providing a website that can cater to all these needs and act as a bridge between students and mentors or between students and students. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to a productive use.

### **References**

Functional & Non-functional Requirements -https://www.altexsoft.com/blog/business/functional-and-non-functional-requirements-specification-and-types/

UI & UX design and wireframing - <https://uxdesign.cc/a-guide-to-minimalist-design-36da72d52431> and https://reactjs.org

## **OVERALL DESCRIPTION**

### **Product Perspective**

LinkedIn is for working professionals, but we as students always wanted to collaborate/work together on projects and bring ideas come to reality. It is quite evident that projects play a vital role in developing the Tech skills of students. A lot of students are unable to accomplish completing a project either because they cannot find people with the relevant skill-set or they have the skills but have no idea in mind. Even if they have the above requirements, they might simply lack the guidance from the Mentor. It aims at helping the students by providing a website that can cater to all these needs and act as a bridge between students and mentors or between students and students. A platform where students, creators and mentors can fulfill their needs hassle free and put their crucial time to a productive use.

### **Product Functions**

* Collaborate: Students can easily find projects that pique their interests and/or suitable for growing their skill-sets. They can also post projects and find people that are willing to collaborate on the project.
* Request Guidance: Students can seek guidance from mentors regarding their projects or can apply for a project under the guidance of mentors.
* Connect: Users can connect with people having skills that complement their and grow their networks and grow their resume by completing more and more projects.

### **User Classes and Characteristics**

The various user classes for our system are:

* + 1. Students
    2. Mentors
    3. Reviewer

### **Operating Environment**

The system is designed to work properly on both touchscreen smartphone having features to connect to the Internet and laptops/desktop PCs. The system is built to run on browser versions IE11.0 or later.

### **Design and Implementation Constraints**

There are some constraints which are holding us from implementing some of the advanced features into our system. The real-time database system we are going to use can serve a maximum of 50 users as of now, due to the constraints on free users of their service. If in future, we are able to upgrade to the Premium plan of that service we will be able to address many more users overall. Apart from this, we are a team of 4 people working on this project and keeping in mind the time constraint within which we have to develop this system, it is difficult for us to design each and every feature neat.

### **User Documentation**

* A document explaining how to use every feature of our system is supposed to be provided within the software itself.

## **SYSTEM FEATURES**

### **System Features**

1. **Searching Projects:** Students can easily search for a project they like or that suits them based on their skill-set.
2. **Posting Projects**: Students and Mentors can share their ideas for projects easily by creating a project so that the interested students can join them easily.
3. **Connecting to each other:** Students can easily connect with like-minded people and grow their network and collaborate on projects.

### **Functional Requirements**

1. **Authenticity:** The various users that registers as mentors must be faculty of the TIET (for the beginning stage) or must have some experience in their respected field.
2. **Reporting Requirements:** Reports submitted by the users must viewed frequently.

## **EXTERNAL INTERFACE REQUIREMENTS**

### **User Interfaces**

* **Student Interface: -** Through this interface students will be able to access the details of the desired mentor and can request for project under their guidance.
* **Mentor Interface: -** By virtue of this interface, Mentors can view the project groups and details of students engaged in projects.
* **Working Professional: -** This interface will provide referral or job opportunities/projects to the students.
* **Moderator Interface: -** This Interface is for Bug fixing, Server and UI management.

### **Hardware Interfaces**

* Laptop/PC with at least 4GB RAM
* Android Smart Phone

### **Software Interfaces**

* MongoDB (Database)
* Nodejs and ExpressJS(Backend)
* ReactJS (Frontend)

### **Communications Interfaces**

### Email (for verification and contacting)

## **OTHER NONFUNCTIONAL REQUIREMENTS**

* **Performance:** The delay of communication between the professor and student should be

minimum.

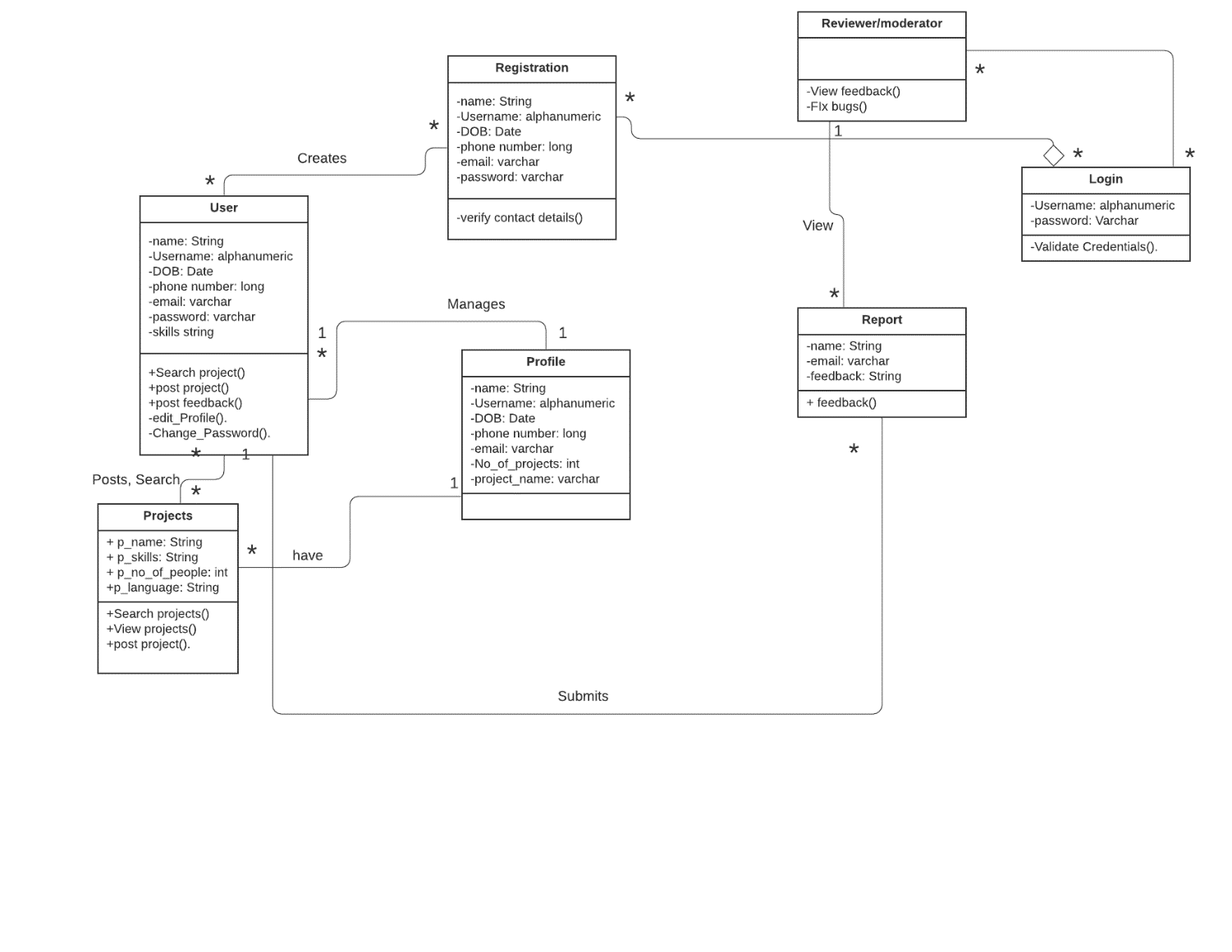
* **Security:** The software should be able to differentiate between Students, Professor and

Admin/Moderator.

* **Maintainability:** The system will maintain logs in order to help troubleshooting by the Admin/Moderator
* **Availability:** Should be available round the clock (24/7)

1. **DESIGN PHASE**

**3.1 Class Diagram**



1

1

Fig16: class diagram

The class diagram is the main building block of object-oriented modeling. It is used for general conceptual modeling of the structure of the application, and for detailed modeling, translating the models into programming code. Class diagrams can also be used for data modeling. The classes in a class diagram represent both the main elements, interactions in the application, and the classes to be programmed. After thorough analysis of use case diagram, it is concluded that seven classes can completely justify object orientation of our project. These classes are Registration, User, Projects, Profile, Report, Login, Reviewer/ Moderator.

**3.2 Sequence Diagram**

**3.2.1 Login sequence**

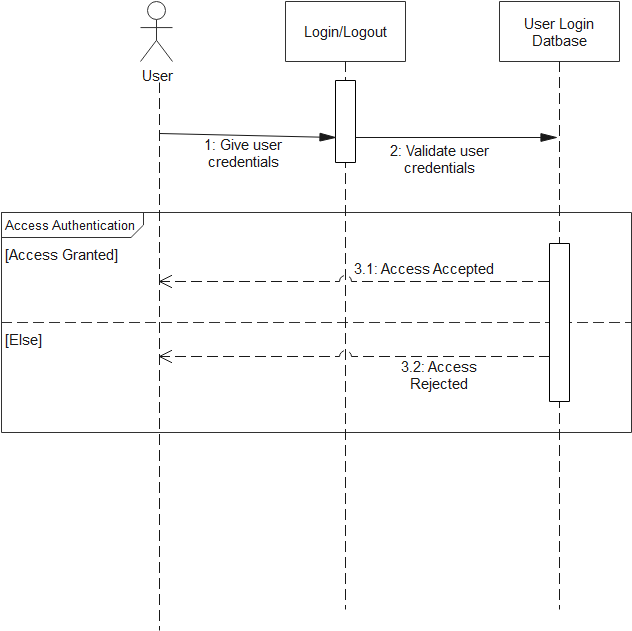


Fig17: Login sequence diagram

**3.2.2 Project and Profile sequence**

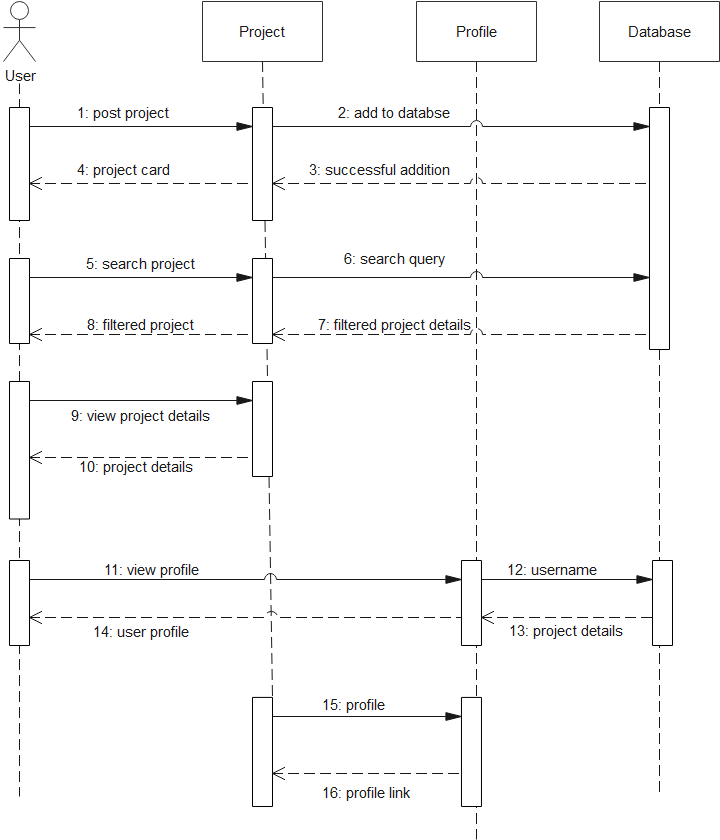


Fig18: Project and Profile sequence

diagram

**3.2.3 Registration Sequence**

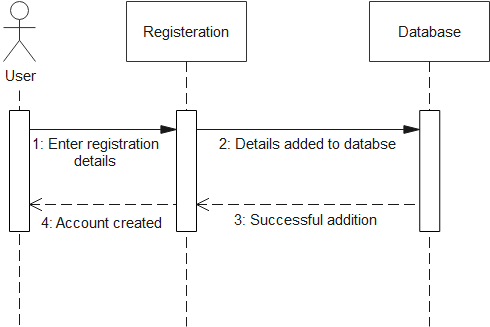


Fig19: Registration sequence

diagram

**3.2.4 Feedback sequence**

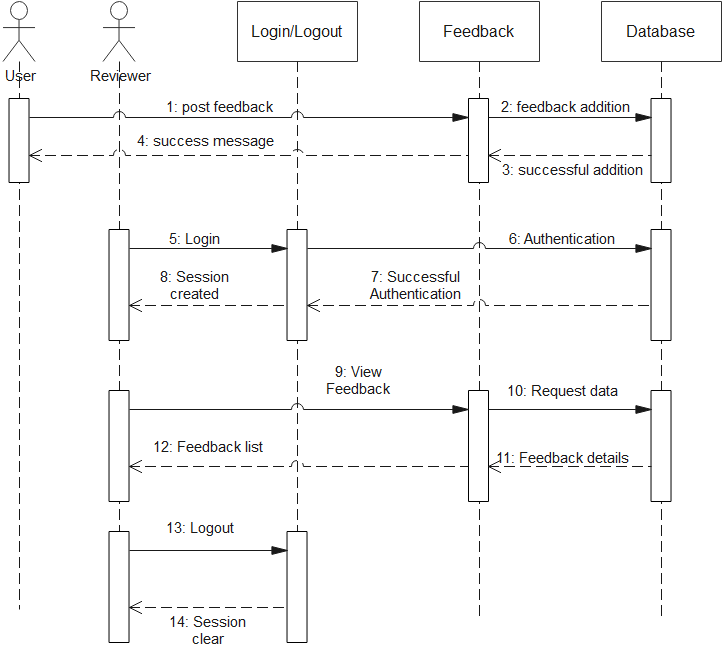


Fig20: Feedback sequence diagram

A sequence diagram shows object interactions arranged in time sequence in the field of software engineering. It depicts the objects involved in the scenario and the sequence of messages exchanged between the objects needed to carry out the functionality of the scenario. Sequence diagrams are typically associated with use case realizations in the logical view of the system under development. Sequence diagrams are sometimes called event diagrams or event scenarios.

**3.3 Collaboration Diagram**

**3.3.1 Mentors, Students**

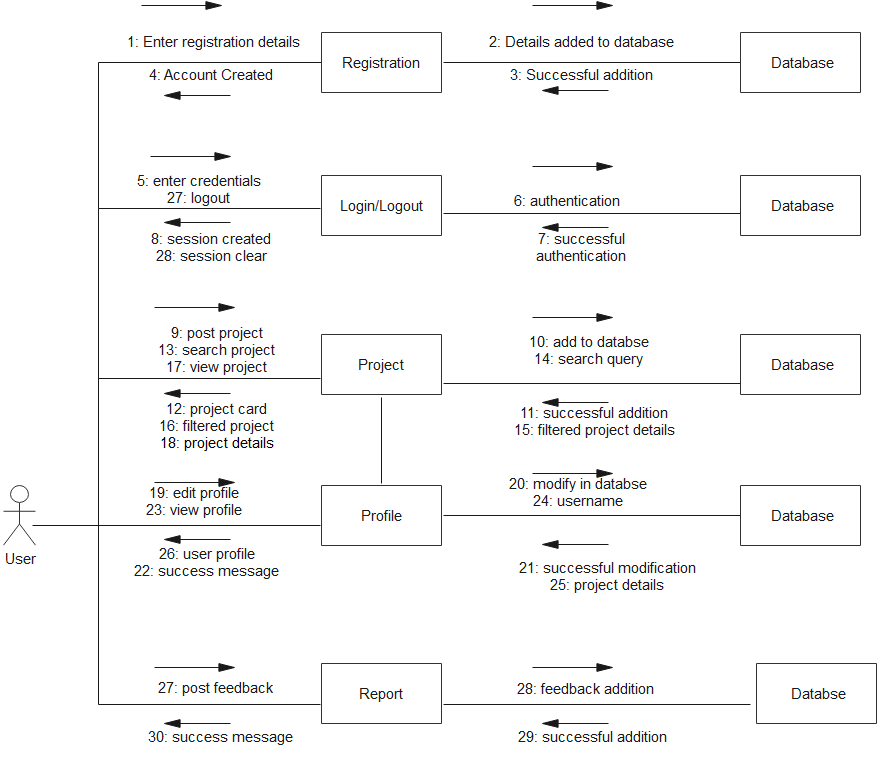


Fig21: Mentors, students collaboration diagram

**3.3.2 Reviewer/Moderator**

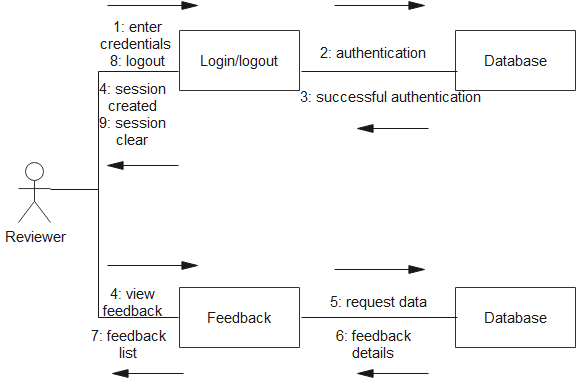
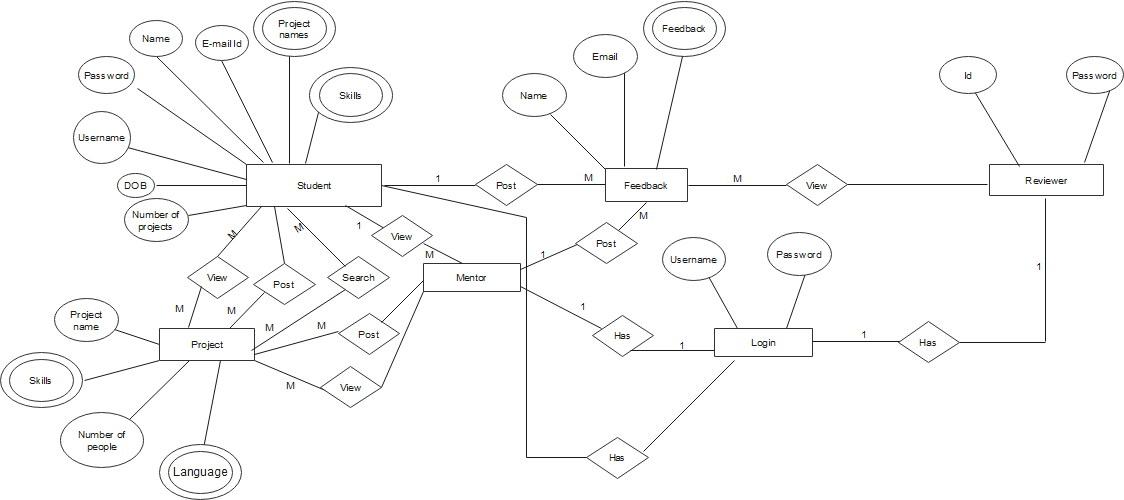


Fig22: Reviewer collaboration diagram

A Collaboration Diagram is a diagram that shows object interactions organized around the objects and their links to each other. Unlike a Sequence Diagram, a Collaboration Diagram shows the relationships among the objects. Sequence diagrams and collaboration diagrams express similar information, but show it in different ways.

**3.4 Database Design: ER Diagram**



M

1

M

M

M

Same as student entity (except number of projects)

1

Fig23: Er diagram

An entity–relationship model (or ER model) describes interrelated things of interest in a specific domain of knowledge. A basic ER model is composed of entity types and specifies relationships that can exist between entities.ER model is commonly formed to represent things a business needs to remember in order to perform business processes. Consequently, the ER model becomes an abstract data model, that defines a data or information structure which can be implemented in a database, typically a relational database.

|  |  |
| --- | --- |
|  | 1. **PROTOTYPING-GUIS** |
| **4.1** | **Screen Shots** |

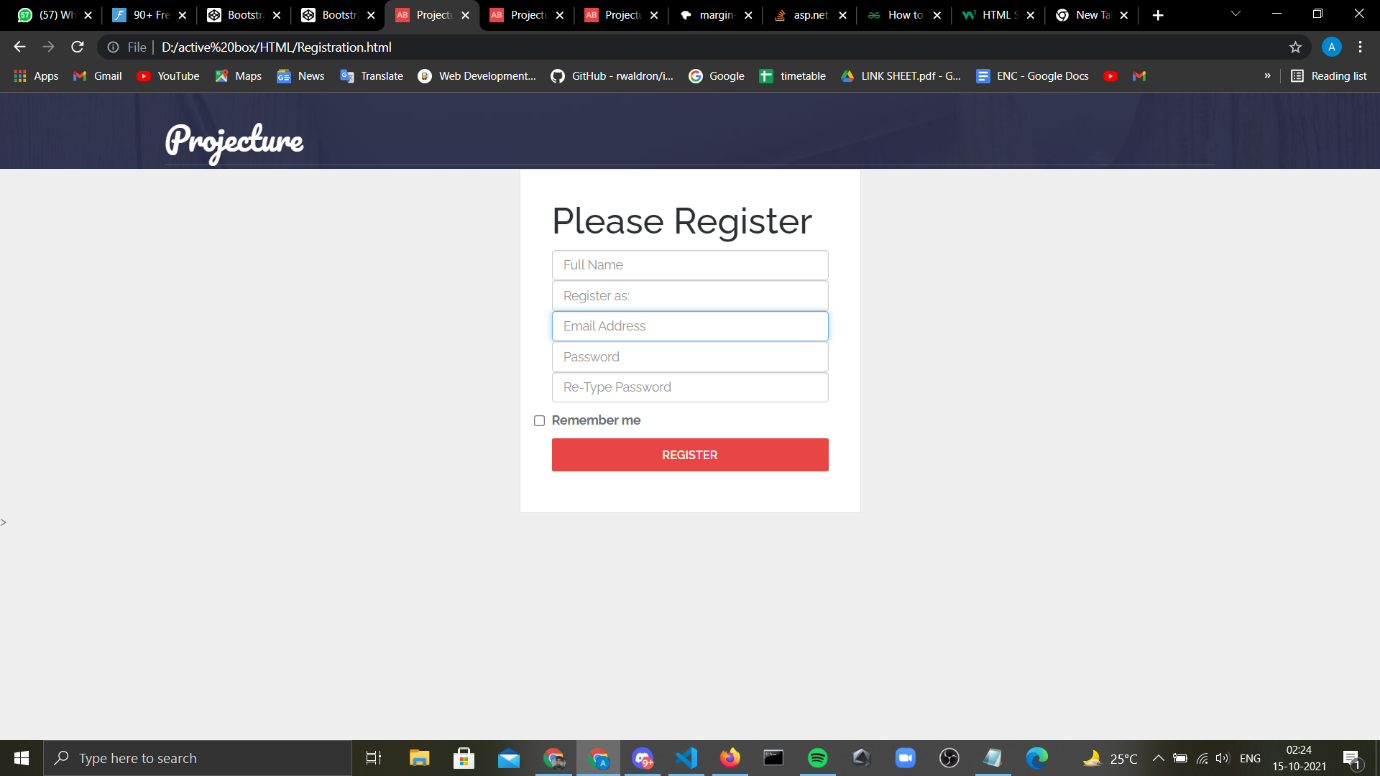


Fig 24: Registration

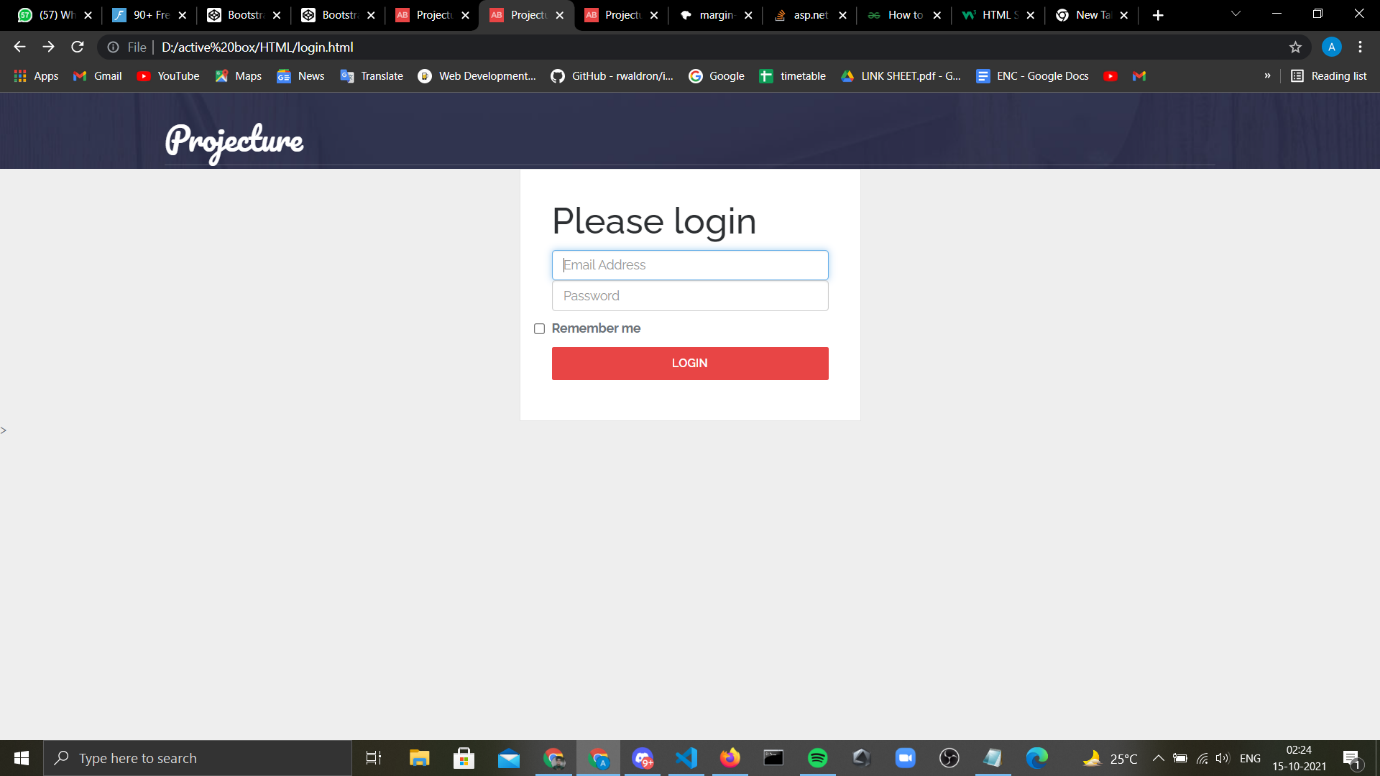
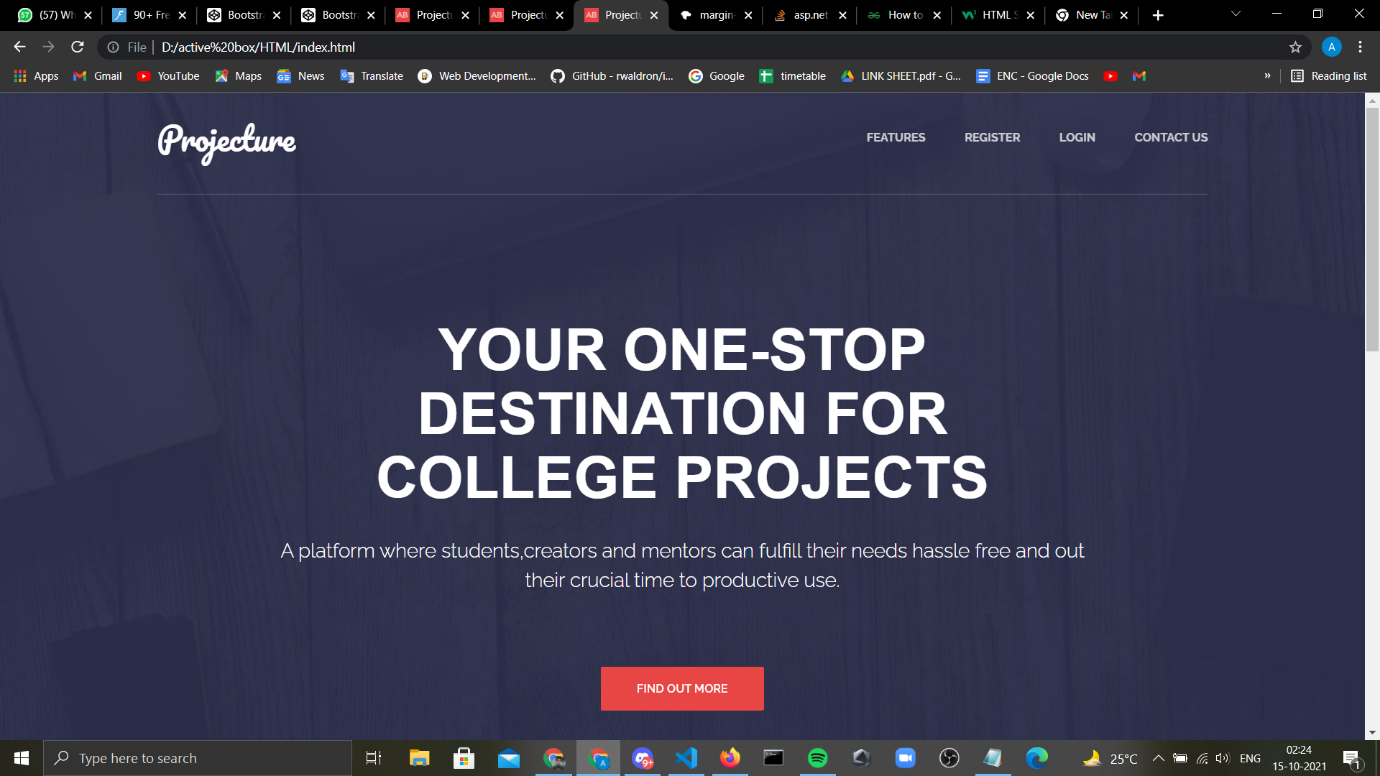
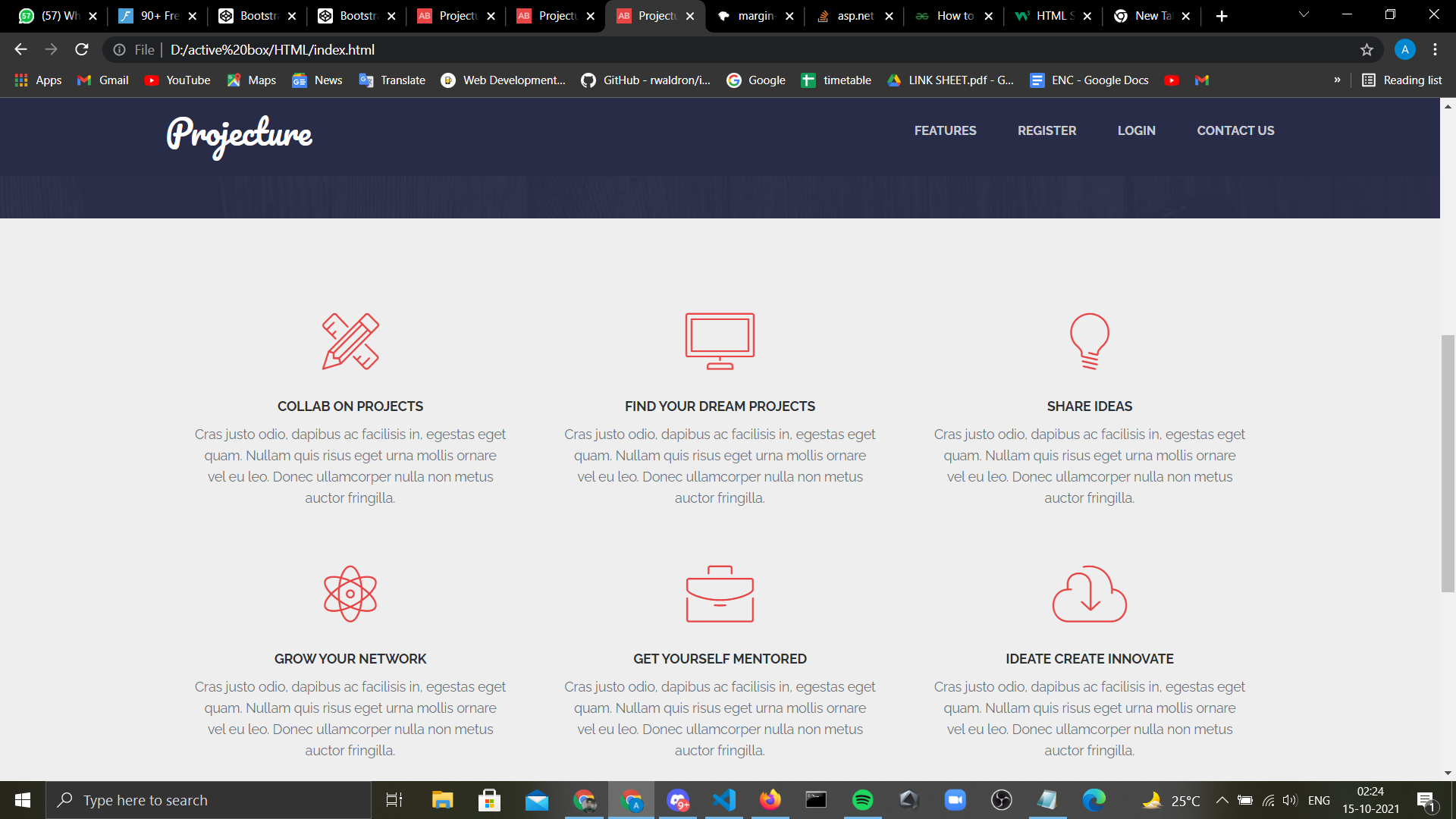


Fig 25: Login





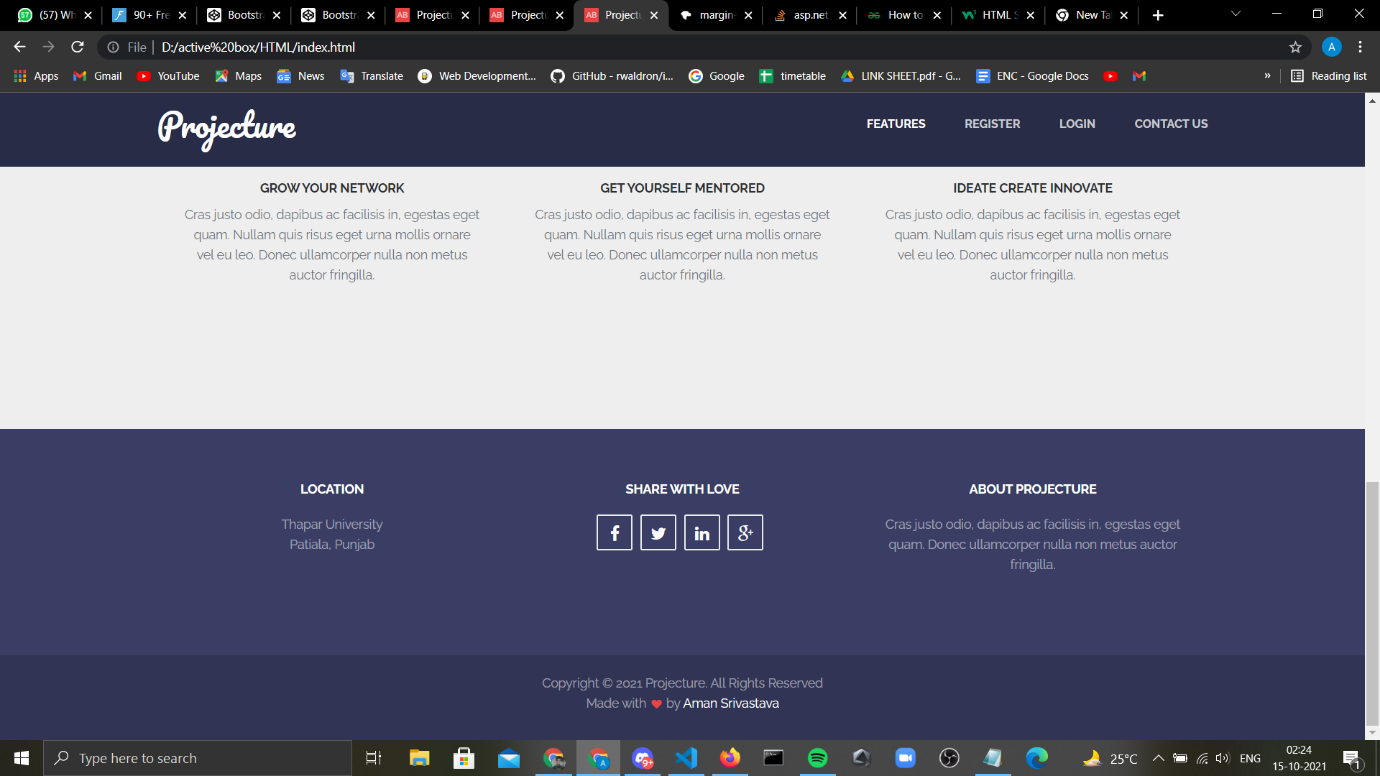


Fig 26: Landing page