Project Report

On

“Student Mentor Guidance System”

Submitted in partial fulfillment for the award of

Post Graduate Diploma in Advanced Computing

(PG-DAC) from IACSD(Pune)

Guided By:

Mr. Milind Arjun

Presented by:

PRN	NAME
210541281017	Mr.Chetan Badgujar
210541281025	Mr. Bhushan Chopade

Centre of Development of Advanced Computing (C-DAC), Pune.

ACKNOWLEDGEMENT

This project "**Student - Mentor Guidance System**" was truly a great learning experience for us and we are submitting this work to Advanced Computing Training School (IACSD Akurdi).

We are very glad to mention the name of Mr. Milind Arjun for his valuable guidanceto work on this project. His guidance and support helped us to overcome various obstacles and intricacies during project work.

Our heartfelt thanks go to Mr. Prashant Karhale, our Course Coordinator, E-*DAC* who gave all the required support and kind coordination to provide all the necessities to complete the project and throughout the course up to the last day herein C-DAC ACTS, Pune.

From:

Mr. Bhushan Chopade (210541281025)
Mr. Chetan Badgujar (210541281017)

TABLE OF CONTENTS

1. Introduction

Development and securing of excellent human resources under both the internal and external environmental changes are a key deciding factor of national competitiveness. However, due to the poor vocational training or career guidance services in college.

The colleges have not been playing their role in the transition to the professional world for their students, who consequently cannot meet the demand from industry. Currently, most colleges provide students with relevant information and vocational guidance via systems such as an on/off-line career information office or consultation center, and an internship However, since a systematic connection between individual students is not made, its effect is utterly limited. Vocational training or career guidance service in the college is poor, And thus colleges cannot play their rightful role in the transition of college students to the professionals' stage after graduation. Therefore, it is considered that college graduates generally cannot meet the demand from Industry.

The main aim of our project is to ease the method and process to clear the concept to the students with effectiveness. We tried to open up N numbers of ways to conceptualize the learning as fast and effective as possible.

By our method, we gave 1 mentor to 20 students, which improves the working of administration as it will help the student to rely on and have a connection with a specific teacher which indeed will result in improvement in student's academics. Also, this leads teachers to teach with ease as there is no pressure to teach more students which is also beneficial in teaching and this improves the overall result.

2. Project Overview

2.1 Purpose

The purpose of this document is to give a detailed description of the requirements for the “Student - Mentor Guidance System” software. This software is intended to provide additional functionality of assigning students to the mentor. It will illustrate the purpose and complete declaration for the development of the system. It will also explain system constraints, interface, and interactions with other external applications

2.2 Scope

The “Student - Mentor Guidance System” is a web-based application that helps people to gather and analyze data related to students and mentors and efficiently assigns students to the mentors depending upon the total number of students and available mentors. The application provides functionality to maintain relationships between the two. Users can provide their details as well as course enrolled information using the web portal. This information will act as the basis for the assignment process. All system information is maintained in a database. The application interacts with the MySQL database and performs insertion, update as well as deletion as directed by the user.

2.3 Feasibility

- A feasibility study is an analysis that takes all of a project's relevant factors into account—including economic, technical, and scheduling

considerations—to ascertain the likelihood of completing the project successfully.

- A feasibility study is simply an assessment of the practicality of a proposed plan or project.
- The following feasibility studies were conducted to make sure that our software is feasible.

2.3.1 Technical feasibility

As per this study, we found that our choice of technology stack was conducive enough to bring the project to fruition. Irrespective of the system in which our backend ran, the results were as expected and platform dependency was not found. The system catered to the requirement of the end-user.

2.3.2 Operational feasibility

As per this study, we concluded that the system is user-friendly and easy to maintain.

The project offers a great deal of user experience and convenience to the target group.

2.3.3 Economical feasibility

As per this study, we concluded that the technology stack we are using in our project is open-sourced, freely available, and well-maintained by the community. This reduces the cost of the system as well as development cost, without compromising the quality of the product. This system was found to be ergonomic to the target customer base.

3. Project Description

3.1 Technology Stack

➢ **Backend**

Category	Technology Name
Framework	**Spring Boot**
ORM Tool	**Hibernate**
Database	**MySQL**
Build Tool	**Maven**
Language	**Java**

➢ **Frontend**

Category	Technology Name
Framework	**REACT-JS**
Language	**HTML, CSS, JS, ES6**

4. User classes

4.1 Admin

The superuser, the admin class represents complete authority over the system. An admin can,

a. Registers both mentor and student
b. View the list of students and mentors who have successfully registered in the system.
c. Deletion of the accounts of mentor and student.
d. View the progress of the course which has been selected by students. The progress consists
e. of tasks performed for each milestone by students related to the course.
f. Manually assign mentors to the students.
g. Log in and log out for each session.

4.2 Mentor

a. Guide class represents a user who is responsible for guiding the students through their course phase.
b. A Guide is registered by the admin. Upon receiving the login credentials into the system, a guide can perform various functionalities. These include,

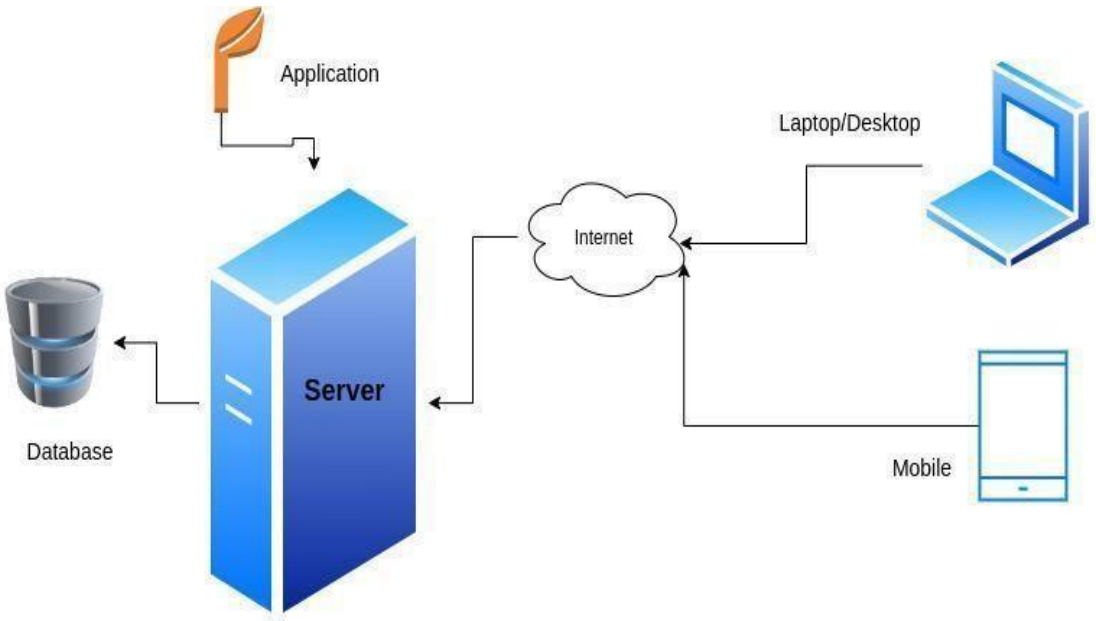
- View the list of students who have successfully registered under them.
- Update the number of students who can be handled (Size of the batch).
- Update the marks of the student after evaluation.
- .Login and log out for each session.
- Signup for the registration.
- Update the personal information registered during signup.

4.3 Student

Once the admin has registered a particular student, he/she can create a project. After project details are properly entered, the student can,

a) View a list of students who have not been assigned to any project yet.
b) Team members can be selected as decided between themselves.
c) Once a project has been successfully registered in the system, students can,
    i. Start creating tasks related to that project.
    ii. View progress of their project.
    iii. Set milestones for tasks that they have created.
    iv. Students can view all the activities taking place in the system like project creation, session start, and end by a guide.

5. Architecture Diagram



6. Software Requirements Specification

System:

The following diagram describes the entire flow of the system:

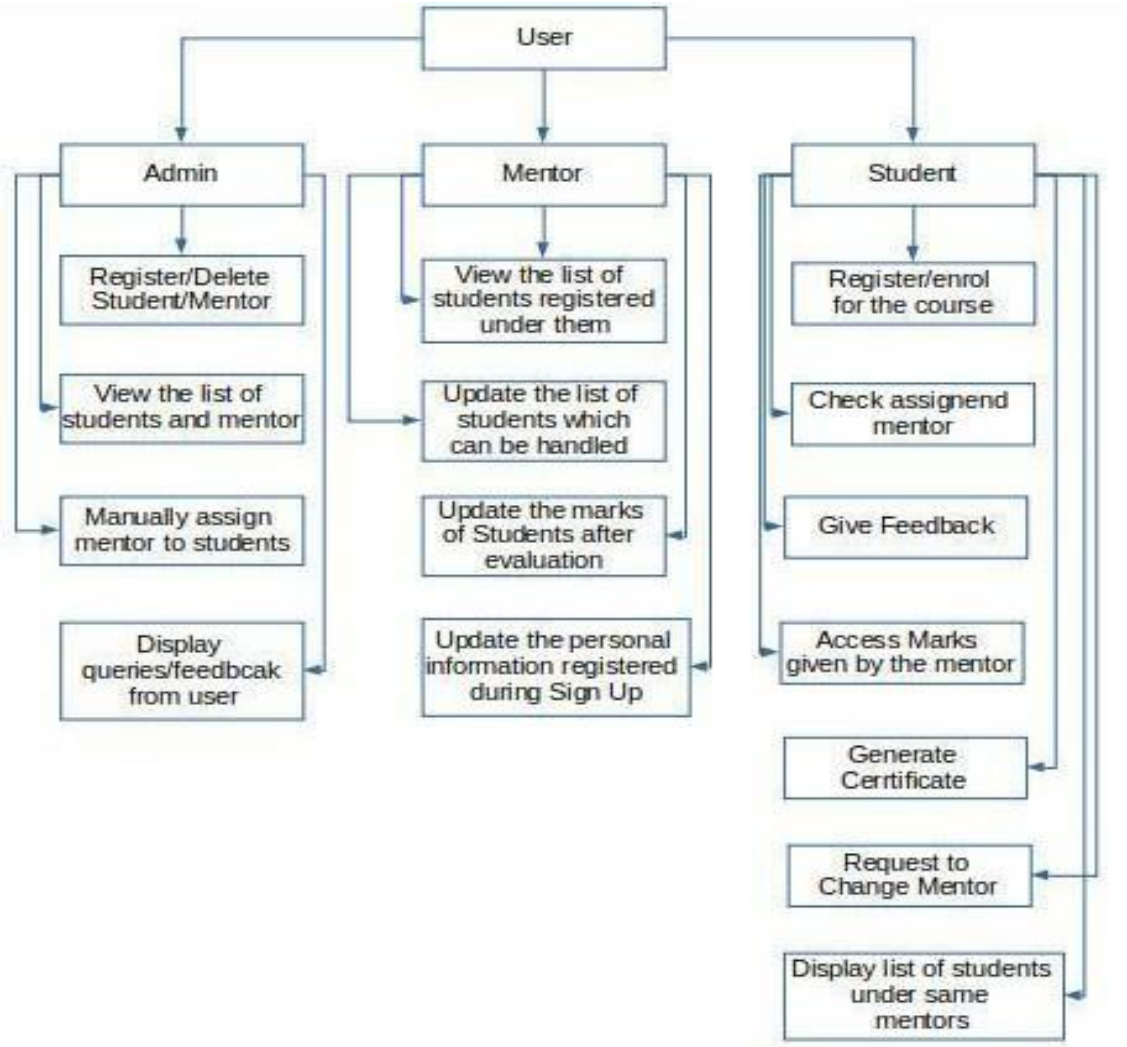


Figure 1: Use case diagram

There is an entry interface that is intended to facilitate the actors [Admin|Mentor|Student] to login into the system provided they have their user account, i.e., already registered with the system. If not then contact Admin for registration as Mentor or Student. The user has to enter the login credentials i.e. Email-Id and Password information for login.

◆ Scenario 1: Mainline Sequence

1. **Admin:** Enter Admin Email and Password.
2. **System:**

    a) Display the Admin dashboard where the admin can handle students and mentors.
    b) View student and mentor information and activities (progress).

◆ Scenario 2: Mainline Sequence

1. **Mentor:** Once registered, enter Mentor Email and Password.
2. **System:**
    a) Display the mentor dashboard where the mentor is responsible for guiding the students through their course phase.

    b) View each student's progress and provide necessary guidance.

◆ Scenario 3: Mainline Sequence

1. **Student:** Once registered, enter Student Email and Password.
2. **System:**

    a) Sign up for registration

    b) Display the Student dashboard where a student can view activities provided by the mentor

6.1 Sequence

◆ ADMIN

Mainline Sequence:

1. Admin**:** Admin logs in.
2. System**:** Opens admin home page.
3. Admin**:** Admin Clicks on Profile.
4. System**:** Opens page and shows admin information.
5. Admin: admin click on delete account.
6. System: admin account will delete.
7. Admin: Admin Clicks on dropdown Course management and clicks on Add course.
8. System: Opens Fill course details form for adding a course.
9. Admin: Clicks on Manage Courses.
10. System: Opens Course List and admin can delete the course.
11. Admin: Admin Clicks on the dropdown Student management and clicks on Student Registration.
12. System: Opens Student Registration from.
13. Admin: Clicks on Manage Student.
14. System: Opens Student List and admin can delete the Student.
15. Admin: Admin Clicks on the dropdown Mentor management and clicks on Mentor Registration.
16. System: Opens Mentor Registration from.
17. Admin: Clicks on Manage Mentor.

18. System: Opens Mentor List and admin can delete the Mentor.
19. Admin: Admin Clicks on dropdown Admin and clicks on Admin Registration.
20. System: Opens Admin Registration from.
21. Admin: Clicks on Manage Admin.
22. System: Opens Admin List and admin can delete the Admin.

◆ Mentor

Mainline Sequence:

1. Mentor**:** Mentor logs in.
2. System**:** Opens mentor home page and shows details of the mentor.
3. Mentor**:** Mentor Clicks on Dropdown profile and clicks on view profile.
4. System**:** Opens page which shows information about login mentor.
5. Mentor**:** Mentor clicks on Dropdown profile and clicks on update profile.
6. System**:** Opens Update information form, mentor can update their information.
7. Mentor**:** Clicks on Dropdown profile and click on delete account.
8. System**:** Delete mentor's account and redirect to the login page.
9. Mentor: Clicks on dropdown Student and click on view batch details.
10. System: Opens list of students under mentor and mentor can delete the student.

11. Mentor: Clicks on dropdown Student and click on update marks of students.

12. System: Opens list of students and mentors can give marks to students and can update it.

13. Mentor: Clicks on dropdown Course and click on Course details.

14. System: Opens course details.

◆ STUDENT

Mainline Sequence:

1. Student**:** Student logs in.
2. System**:** Opens student home page and shows details of the student.
3. Student**:** Student Clicks on Dropdown profile and clicks on view profile.
4. System**:** Opens page which shows information about login student.
5. Student**:** Student clicks Dropdown Profile and clicks on update profile.
6. System**:** Opens Update information form student can update their information.
7. Student**:** Clicks on Dropdown Profile and click on delete account.
8. System**:** Delete student account and redirect to the login page.
9. Student: Clicks on dropdown Mentor and click on Get mentor.
10. System: Click on the get information butxton and the student will get the mentor.
11. Student: Clicks on dropdown Mentor and click on mentor information.
12. System: Opens the page which shows the mentor information.
13. Student: Clicks on dropdown Course and click on Course details.

14. System: Opens course details.

7. Non-Functional Requirement

➢ **Performance Requirement**

1. The time between request and response should be less.
2. Minimum time should be taken by the application to display the result.
3. In case of power failure, the data should be stored in the state that was last saved by the user.

➢ **Security Requirement**

1. Only one active session per user
2. Session timeouts after a specified time using the JWT token.
3. Authorization based on roles on the application and have these roles apply to the specific URL accessed dynamically at run time.
4. Passwords shall never be viewable at the point of entry or at any other time.
5. Students can not update their marks.

8. Software Quality Attributes

Various software quality attributes were taken into considerations while designing the system:

1. **Availability:** As our Student Mentor Guidance System is a web-based service provided to the users, it will be available as long as the server is up.

2. **Interoperability:** Student Mentor Guidance System is interoperable on various operating systems, hence, increasing the application's usability and flexibility.

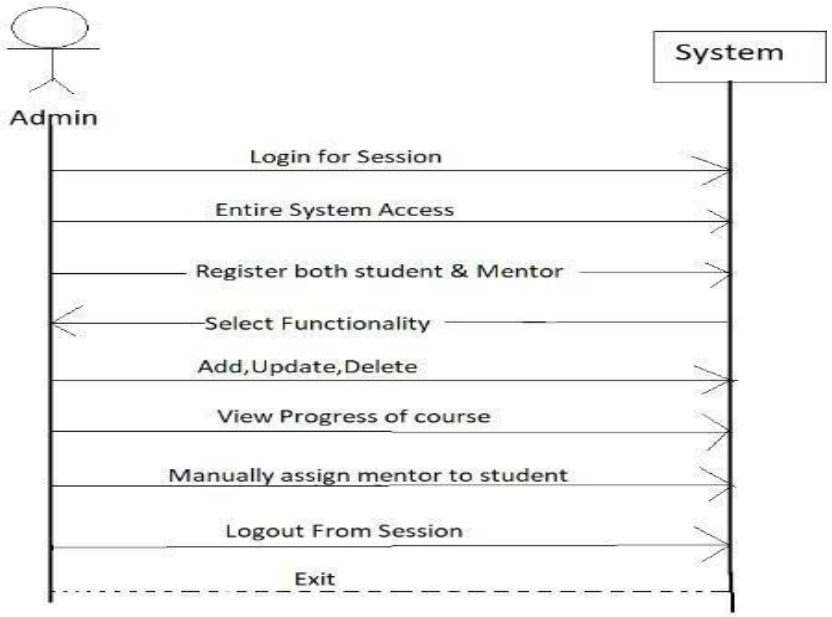
3. **Usability:** The main purpose of developing the Student Mentor Guidance System is to create a system for CDAC students so that,

    a. The entire project life cycle can be tracked, managed, and have accountable to the amount of work being accomplished by each team.
    b. Track the assistance being provided by the mentor
    c. Admin can see the progress of all the students in real-time.

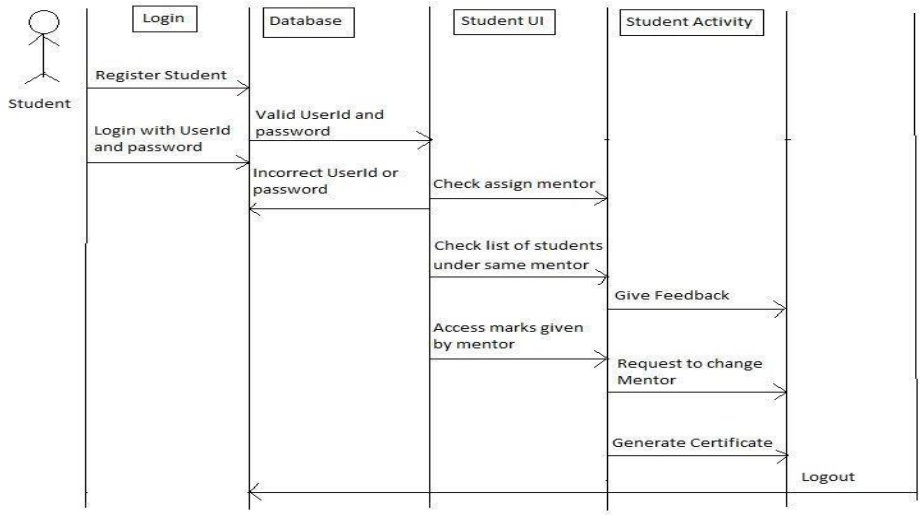
9. Sequence Diagram:

A sequence diagram simply depicts an interaction between objects in a sequential order in which these interactions take place.

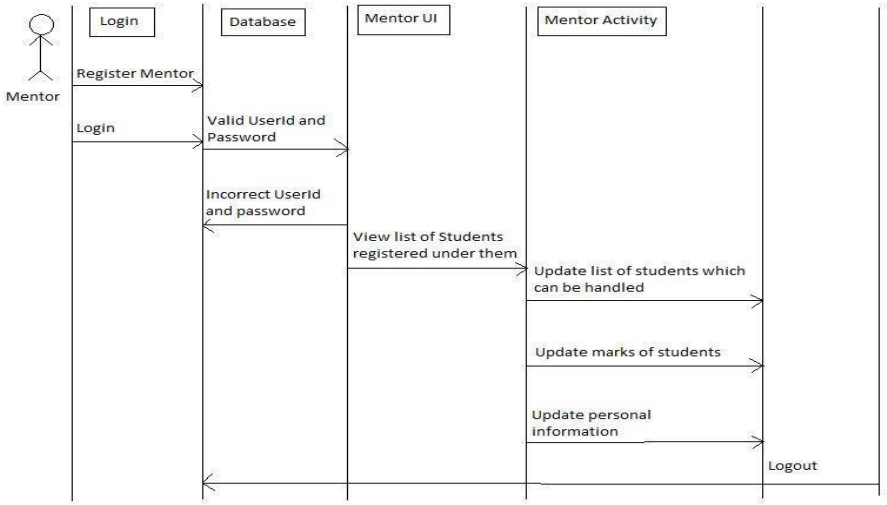
Admin Model:



Student Model:



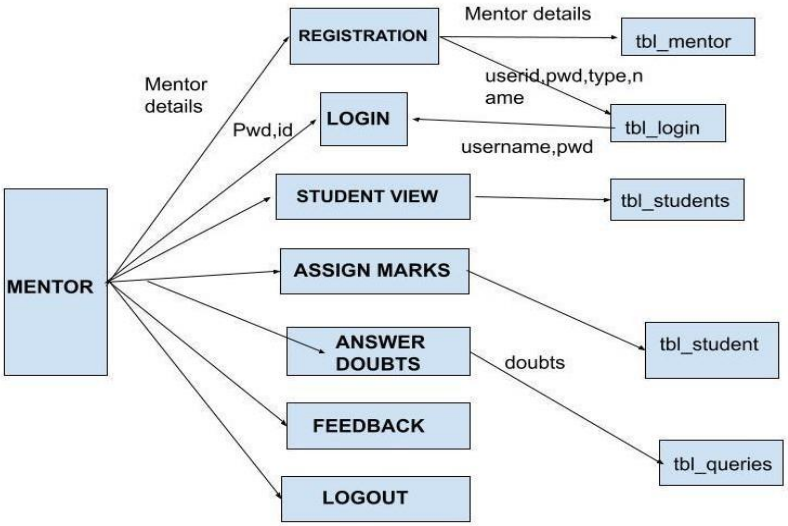
Mentor Model:



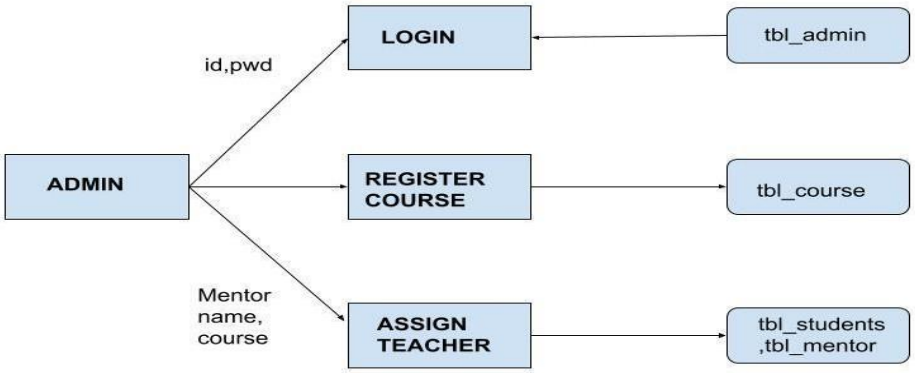
10. Data Flow Diagram:

Data Flow Diagram represents a detailed and well-explained diagram of system components.

Level 1 - DFD Mentor Model:



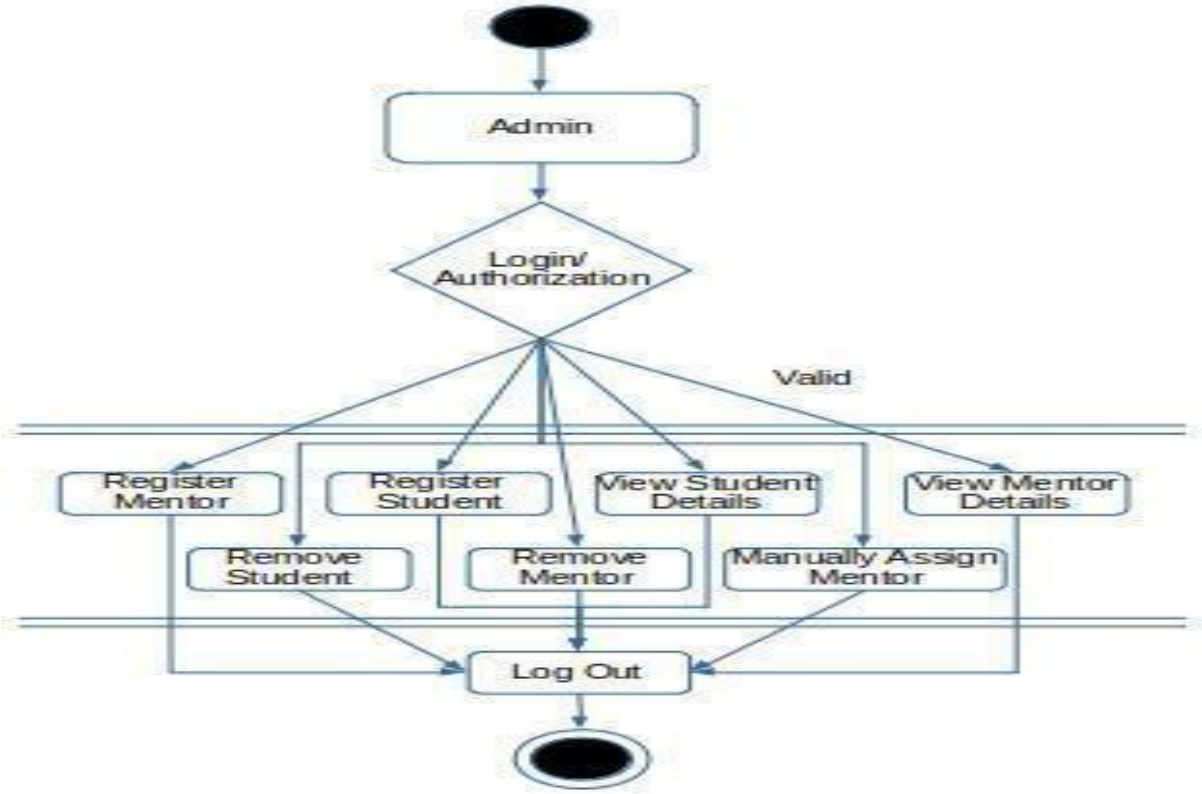
Level 1 - DFD Admin Model



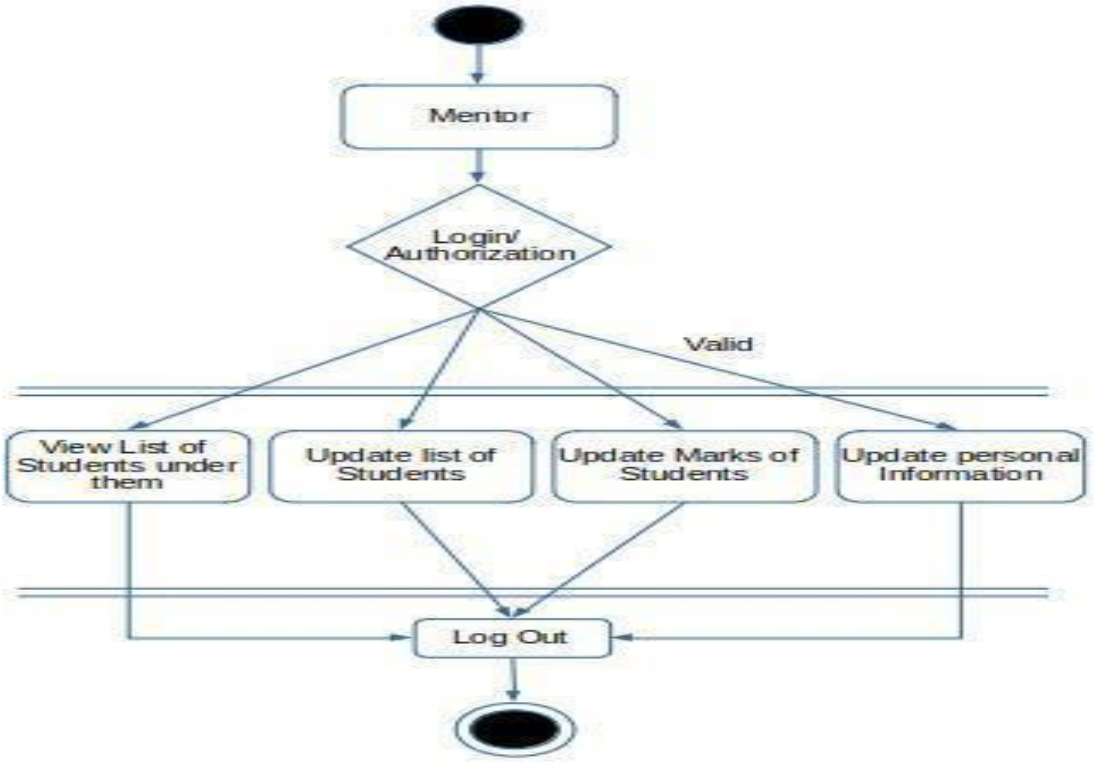
11. Activity Diagram

An activity diagram portrays the control flow of SMGS from a start point to a finish point showing the various decision paths that exist while the activity is being executed.

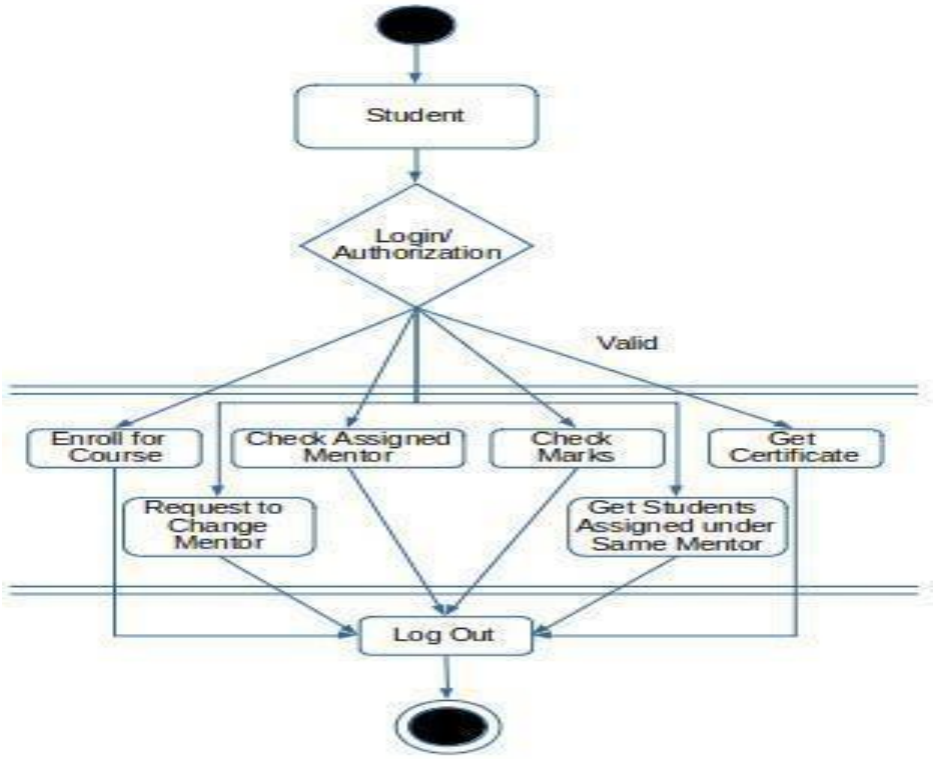
11.1 Admin Activity Diagram



11.2 Mentor Activity Diagram



11.3 Mentor Activity Diagram



12. Table Structure :

1.Table Name: admin

Description: To store the admin credential Information.

Primary Key: admin_id

```
mysql> desc admin;
+----------------+--------------+------+-----+---------+----------------+
| Field          | Type         | Null | Key | Default | Extra          |
+----------------+--------------+------+-----+---------+----------------+
| admin_id       | int          | NO   | PRI | NULL    | auto_increment |
| admin_dob      | date         | YES  |     | NULL    |                |
| admin_email    | varchar(255) | YES  |     | NULL    |                |
| admin_gender   | int          | YES  |     | NULL    |                |
| admin_password | varchar(255) | YES  |     | NULL    |                |
+----------------+--------------+------+-----+---------+----------------+
5 rows in set (0.06 sec)
```

2.Table Name: students

Description: To store the user Personal Information.

Primary Key: student_id

Foreign Key: assigned_mentor_id, student_course_id , student_address_id

```
mysql> desc students;
+--------------------+--------------+------+-----+---------+----------------+
| Field              | Type         | Null | Key | Default | Extra          |
+--------------------+--------------+------+-----+---------+----------------+
| student_id         | int          | NO   | PRI | NULL    | auto_increment |
| student_dob        | date         | YES  |     | NULL    |                |
| student_email      | varchar(255) | YES  |     | NULL    |                |
| student_fname      | varchar(255) | YES  |     | NULL    |                |
| student_gender     | int          | YES  |     | NULL    |                |
| student_lname      | varchar(255) | YES  |     | NULL    |                |
| student_marks      | float        | YES  |     | NULL    |                |
| student_mobile_no  | varchar(255) | YES  |     | NULL    |                |
| student_password   | varchar(255) | YES  |     | NULL    |                |
| assigned_mentor_id | int          | YES  | MUL | NULL    |                |
| student_course_id  | int          | YES  | MUL | NULL    |                |
| student_address_id | int          | YES  | MUL | NULL    |                |
+--------------------+--------------+------+-----+---------+----------------+
12 rows in set (0.00 sec)
```

3.Table Name: courses

Description: To store the courses releated Information.

Primary Key: course_id

```
mysql> desc courses;
+-------------+-------------+------+-----+---------+----------------+
| Field       | Type        | Null | Key | Default | Extra          |
+-------------+-------------+------+-----+---------+----------------+
| course_id   | int         | NO   | PRI | NULL    | auto_increment |
| course_name | varchar(20) | YES  |     | NULL    |                |
| end_date    | date        | YES  |     | NULL    |                |
| start_date  | date        | YES  |     | NULL    |                |
+-------------+-------------+------+-----+---------+----------------+
4 rows in set (0.00 sec)
```

4.Table Name: mentors
Description: To store the Mentors Personal Information.
Primary Key: mentor_id
Foreign Key : mentor_course_id , mentor_address_id

```
mysql> desc mentors;
+--------------------+--------------+------+-----+---------+----------------+
| Field              | Type         | Null | Key | Default | Extra          |
+--------------------+--------------+------+-----+---------+----------------+
| mentor_id          | int          | NO   | PRI | NULL    | auto_increment |
| avg_rating         | float        | YES  |     | NULL    |                |
| batch_size         | int          | YES  |     | NULL    |                |
| current_batch_size | int          | YES  |     | 0       |                |
| mentor_dob         | date         | YES  |     | NULL    |                |
| mentor_email       | varchar(255) | YES  |     | NULL    |                |
| mentor_fname       | varchar(255) | YES  |     | NULL    |                |
| mentor_gender      | int          | YES  |     | NULL    |                |
| mentor_join_year   | date         | YES  |     | NULL    |                |
| mentor_lname       | varchar(255) | YES  |     | NULL    |                |
| mentor_mono        | varchar(255) | YES  |     | NULL    |                |
| mentor_password    | varchar(255) | YES  |     | NULL    |                |
| mentor_course_id   | int          | YES  | MUL | NULL    |                |
| mentor_address_id  | int          | YES  | MUL | NULL    |                |
+--------------------+--------------+------+-----+---------+----------------+
14 rows in set (0.00 sec)
```

5.Table Name: addresses
Description: To store the address releated Information.
Primary Key: address_id

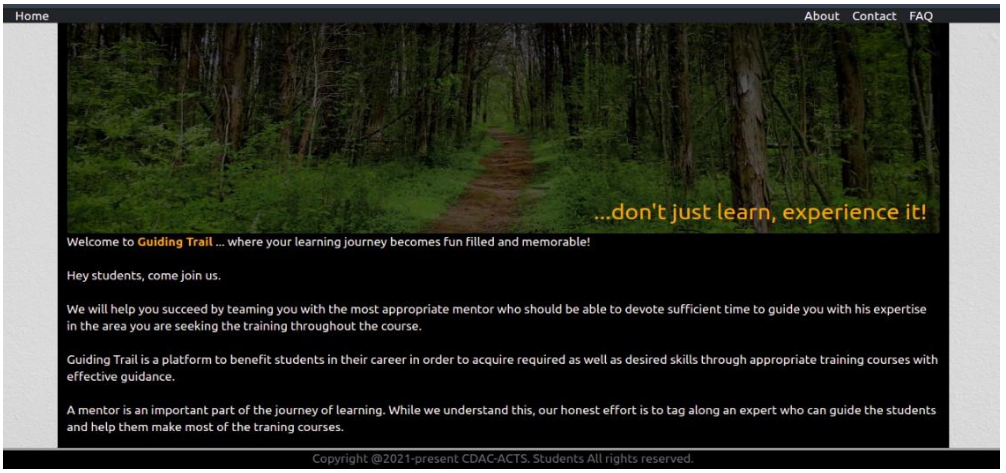
```
mysql> desc addresses;
+---------------+--------------+------+-----+---------+----------------+
| Field         | Type         | Null | Key | Default | Extra          |
+---------------+--------------+------+-----+---------+----------------+
| address_id    | int          | NO   | PRI | NULL    | auto_increment |
| address_line1 | varchar(255) | YES  |     | NULL    |                |
| address_line2 | varchar(255) | YES  |     | NULL    |                |
| area          | varchar(255) | YES  |     | NULL    |                |
| city          | varchar(255) | YES  |     | NULL    |                |
| country       | varchar(255) | YES  |     | NULL    |                |
| pin_code      | int          | YES  |     | NULL    |                |
| state         | varchar(255) | YES  |     | NULL    |                |
+---------------+--------------+------+-----+---------+----------------+
8 rows in set (0.00 sec)
```

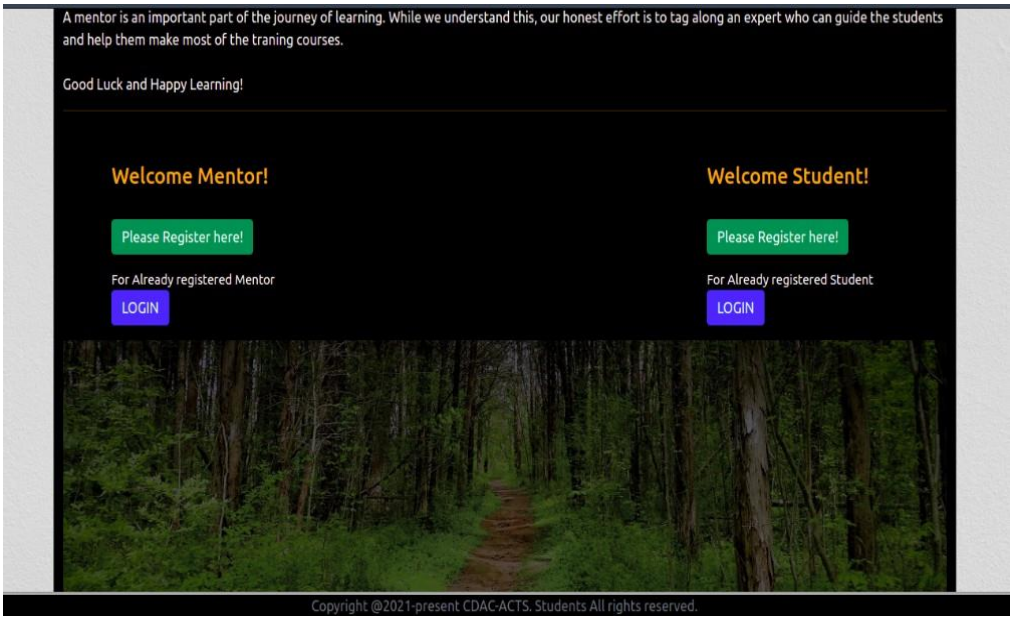
6.Table Name: queries
Description: To store the queries releated Information.
Primary Key: query_id

```
mysql> desc queries;
+----------+--------------+------+-----+---------+----------------+
| Field    | Type         | Null | Key | Default | Extra          |
+----------+--------------+------+-----+---------+----------------+
| query_id | int          | NO   | PRI | NULL    | auto_increment |
| answer   | varchar(255) | YES  |     | NULL    |                |
| question | varchar(255) | YES  |     | NULL    |                |
+----------+--------------+------+-----+---------+----------------+
3 rows in set (0.01 sec)
```

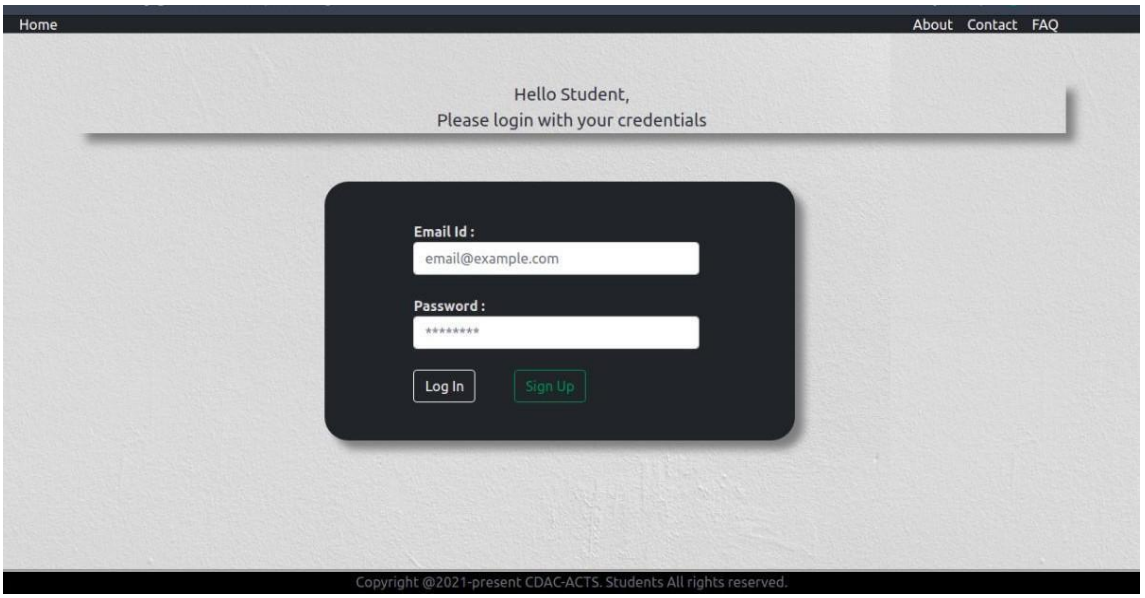
13. User Interface :

13.1 Home Page

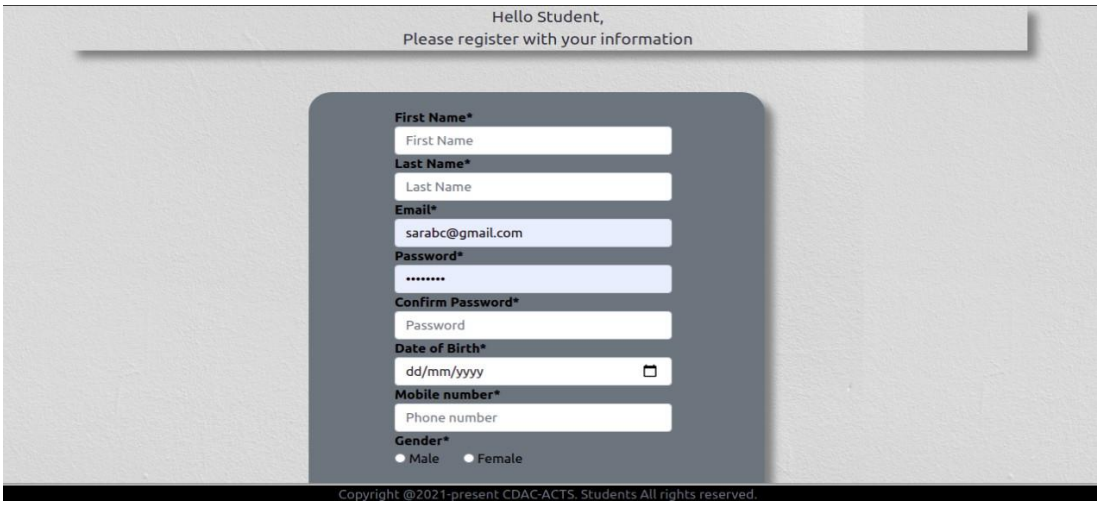




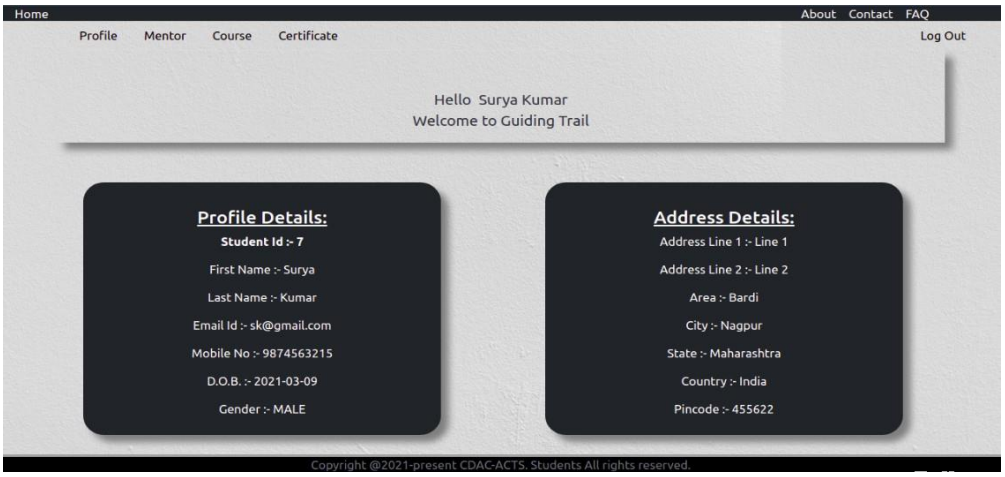
13.2 Login Page



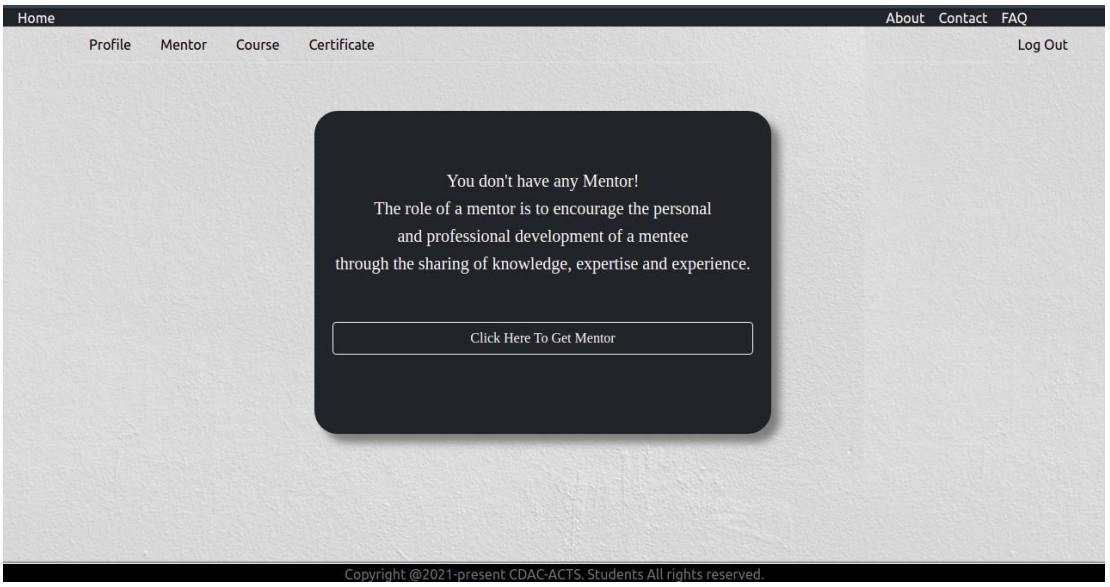
13.3 Student Registration

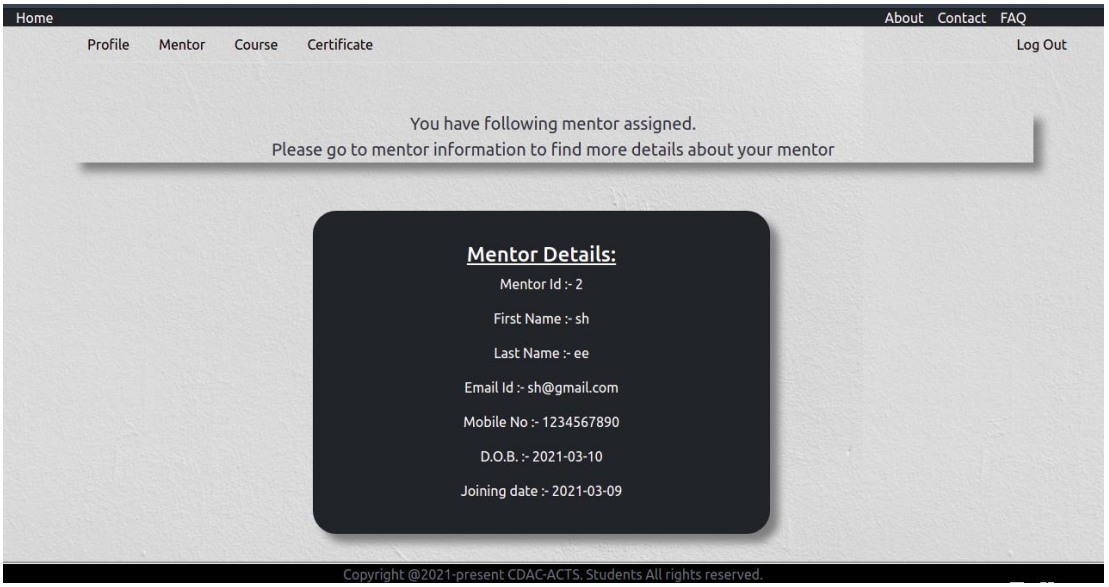


13.4 Student HomePage

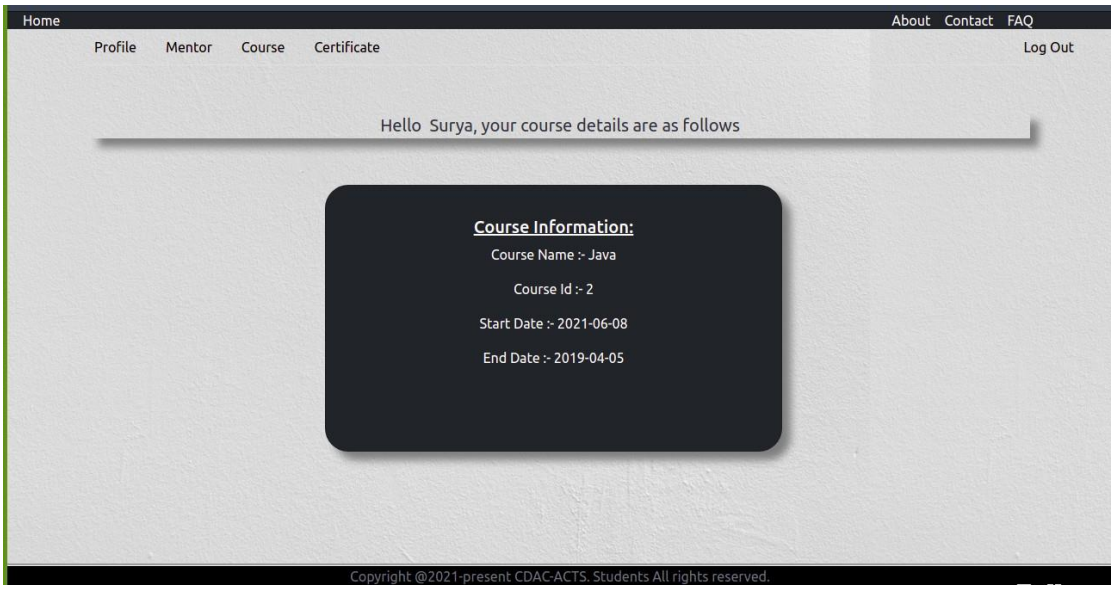


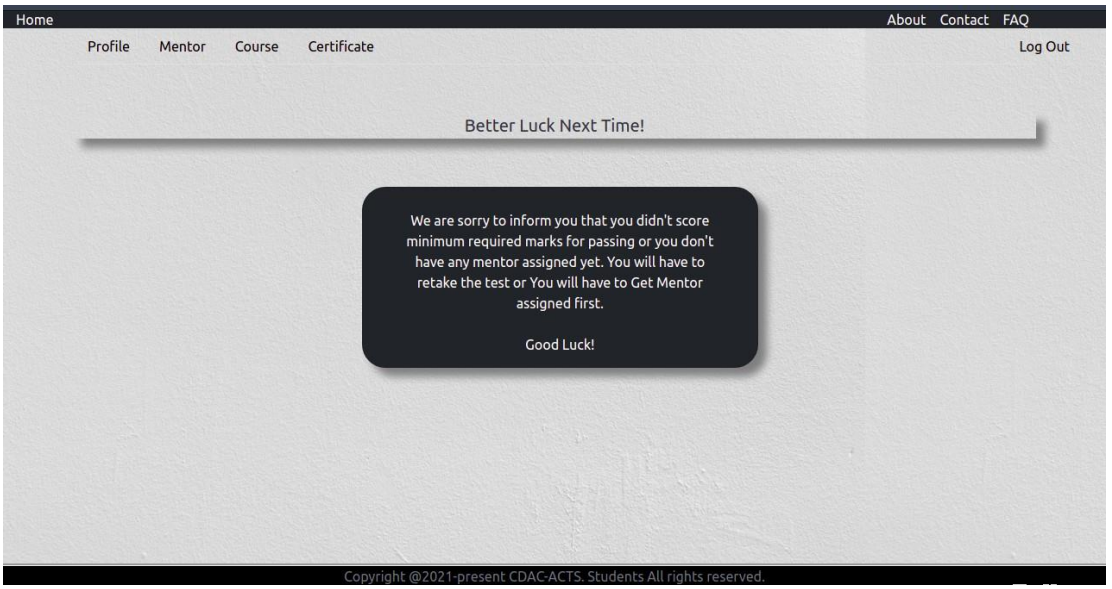
13.5 Mentor Assign



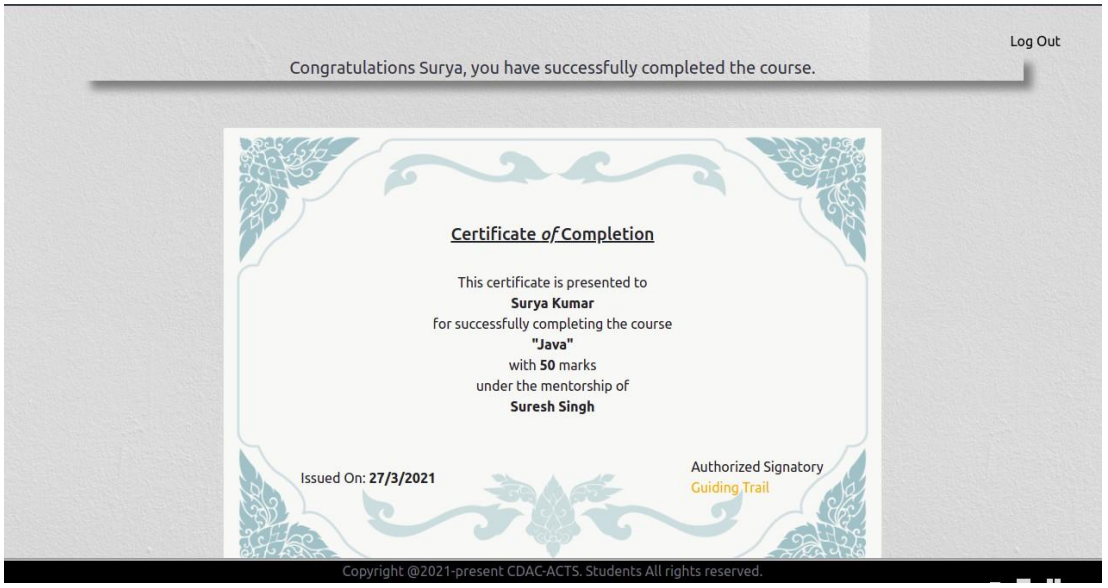


13.6 Course Information

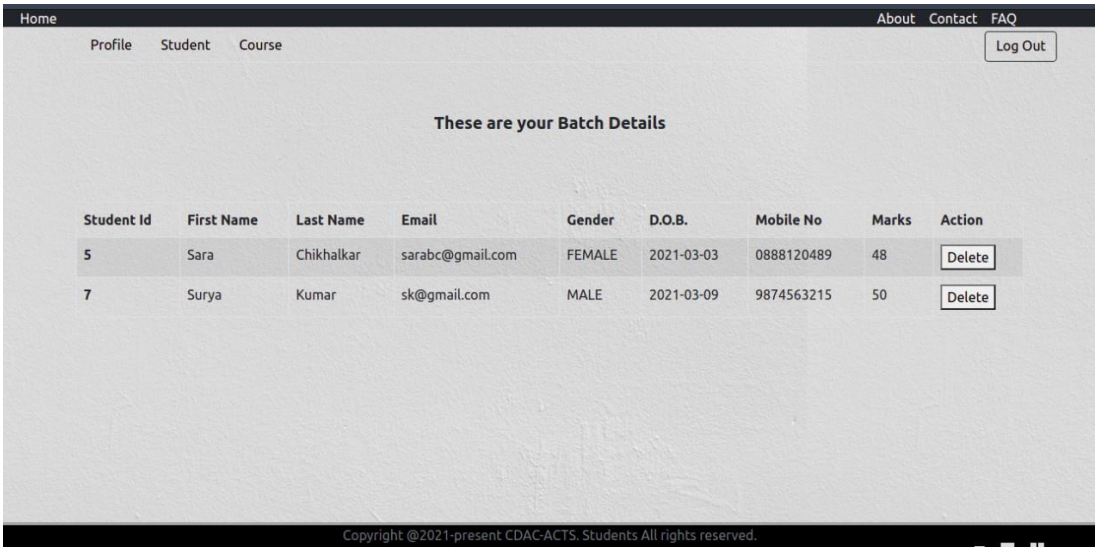




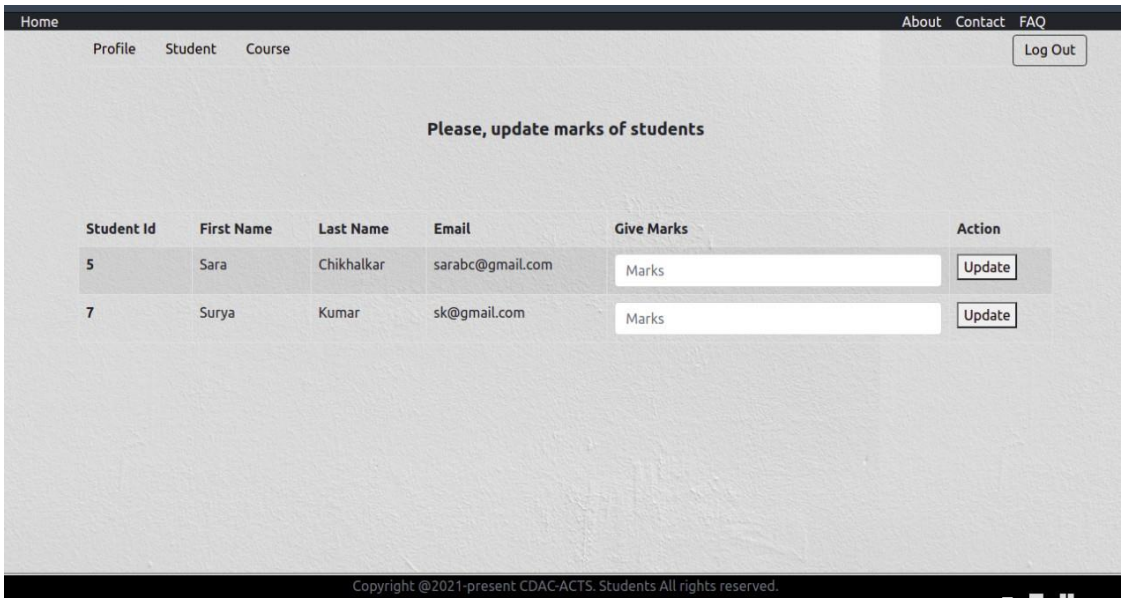
13.7 Certificate



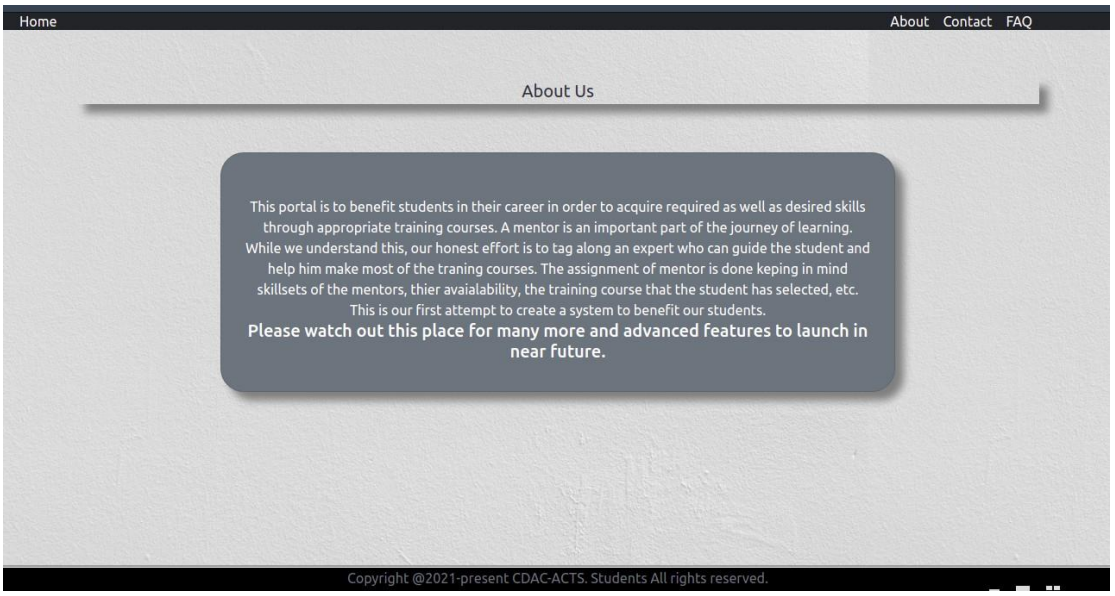
13.8 Batch Details



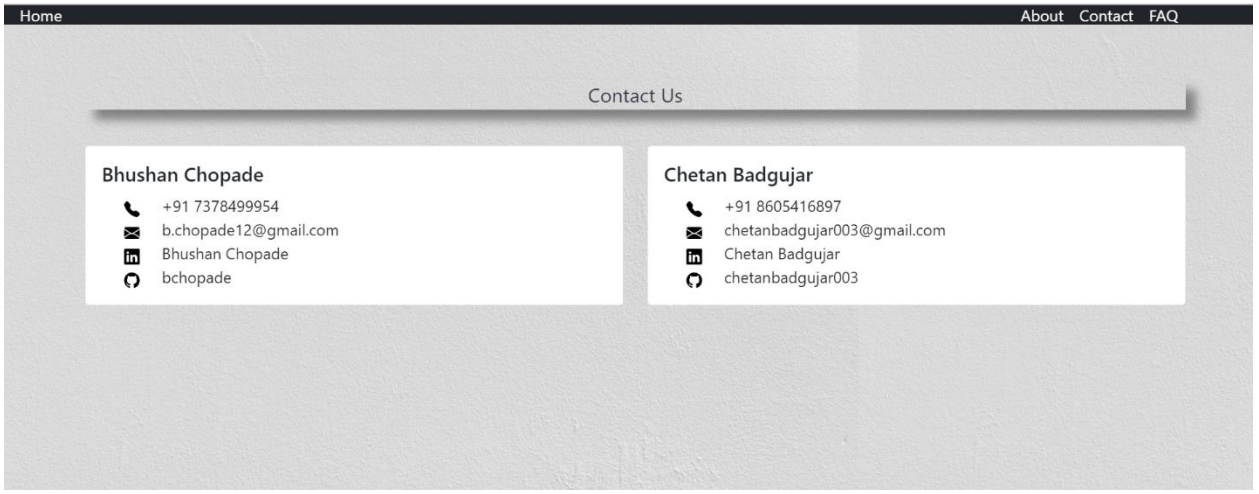
13.7 Update Students Marks



13.8 About Us



13.7 Contact Us



14. Future Scope:

As stated before, this project enhances the communication between student and mentor thereby improving the academic performances of the student. Each student is graded according to their Performances and they receive questions based on these grades. Their grades may improve or fall based on their performances. Hence varying levels of attention can be given to the students. By this work, we conclude that e-mentoring in an academic institute can be developed and tremendous System which is easily accessible to parents as well as mentors and students. Hence it will allow the mentors to dedicate more time whenever they wish and can give much precise feedback that will give proper guidance and the right solution to the problems of students.

15. References:

1. StackOverflow: https://stackoverflow.com/
2. GitHub: https://github.com/
3. Java Docs:
https://docs.oracle.com/javase/8/docs/technotes/tools/windows/javadoc.html
4. Spring Boot Docs:
https://docs.spring.io/spring-boot/docs/current/reference/htmlsingle/
5. Hibernate Docs: https://hibernate.org/orm/documentation/5.4/
6. ReactJS Docs: https://reactjs.org/docs/getting-started.html

16.Test Reports:

The test of report is given hereunder:

Sr.No.	Test Case Title	Description	Expected Outcome	Error Message	Result
1	Login Page – Admin	If User Email=Admin Email, Password= Admin Password	If Validated allow for Admin Home Page If not redirect to the same page	Username and password required	Passed
2	Login Page – Mentor	If User Email=Mentor Email, Password= Mentor Password	If Validated allow for Admin Home Page If not redirect to the same page	Username and password required	Passed
3	Login Page – Student	If User Email= Student Email, Password= Student Password	If Validated allow for Admin Home Page If not redirect to the same page	Username and password required	Passed
4	Show Details of user	Admin can see the status of given user	User Details	No Error	Passed
5	New User Registration	Admin can register new Admin, Mentor, Student, and Course	If Validated, Success Message with user details registered	Validation Error	Passed
6	Update	Admin can update new Admin, Mentor, Student, and Course	If Validated, Success Message with user details registered	Validation Error	Passed
7	Deletion	Admin can delete new Admin, Mentor, Student, and Course	Success Message	No Error	Passed
8	Log out	User / Admin can log out by using the Logout link	Redirected to Home page	No Error	Passed