Database Management System Project

Faculty - Dr. Arup Ghosh
Slot   - ETH-D2

TOPIC

College Management System

Team Member Details

Name	Registration Number
Karunesh Tripathi	19BCE0880
Neelesh Sharma	19BCE0768
Rishabh Raj	19BCE0777

TABLE OF CONTENT

ABSTRACT

The project entitled as "COLLEGE MANAGEMENT SYSTEM" is to computerize the Front Office Management of College to develop software which is user friendly simple, fast, and cost – effective. It deals with the collection of student's information, details about all the faculties as well as non teaching staff. Traditionally, it was done manually. The main function of the system is register and store all the details about student and faculties and retrieve these details as and when required, and also to manipulate these details meaningfully System input contains details such as student's name, living in hostel or not, which department he/she belongs to etc. and same goes with the faculty, while system output is to get these details on to the screen. The data are well protected for personal use and makes the data processing very fast.

INTRODUCTION

The project "College management system"includes registration of students, faculties storing their details into the system. The software has the facility to give a unique id for every student and stores the details of every student,faculty, administrative staff and the maintenance staff . It includes a search facility to know the current status of any student or faculty i.e. to which department he/she belongs to and their respective building be it academic or hostel. The stored details can be accessed by an higher authority of college who is allowed to ,comfortably without much complications . Only they can add data into the database. The data can be retrieved easily. The interface is very user-friendly. The data are well protected for personal use and makes the data processing very fast. College management System is powerful, flexible, and easy to use and is designed and developed to deliver real conceivable benefits to institutions using it. This project is designed to cover a wide range of college administration and management processes. It is an integrated end-to-end Smart College management system that provides relevant information about the college to support effective decision making for students and faculty care, in a seamless flow. College Management System is a software product suite designed to improve the quality and management of the Colleges.

Logical View of this project
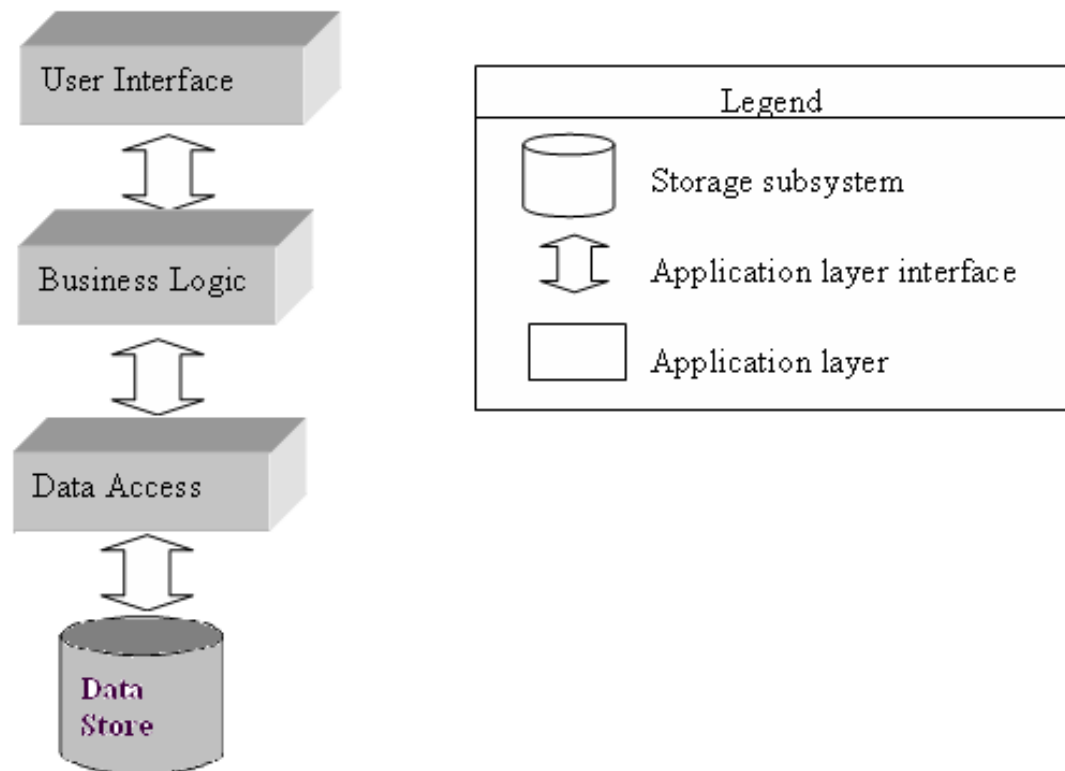
Fig. Abstract view of the project for the user.

Project Resource Requirements

Software Requirements: -

- JavaFX (front end)
- MS Word 97 or later.
- Web Browser: Microsoft Internet Explorer, Mozilla, Google Chrome or any other.
- MySQL Server (back-end).

Hardware requirements:-

- Compatible to windows 8 and above.
- Ubuntu and macOS i3 processor or above.
- 4gb ram or above.
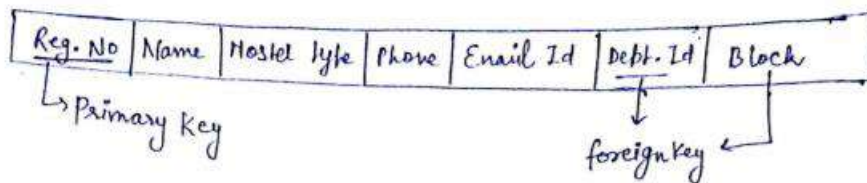
PROJECT MODEL AND APPLICATION

This Project Model system can be described as follows:-

1. It is a standalone module, available for desktop application used by computer Administrator to store student's and faculties data.

2. The storing and retrieval of data is quick It is an offline application.

3. Access is given to only authorize person and no other person can access this software.

4. Data security is maintained properly by authentication of users.

5. Large amount of data can be processed quickly with ease.

6. It is economically feasible.

ENTITY RELATION DIAGRAM
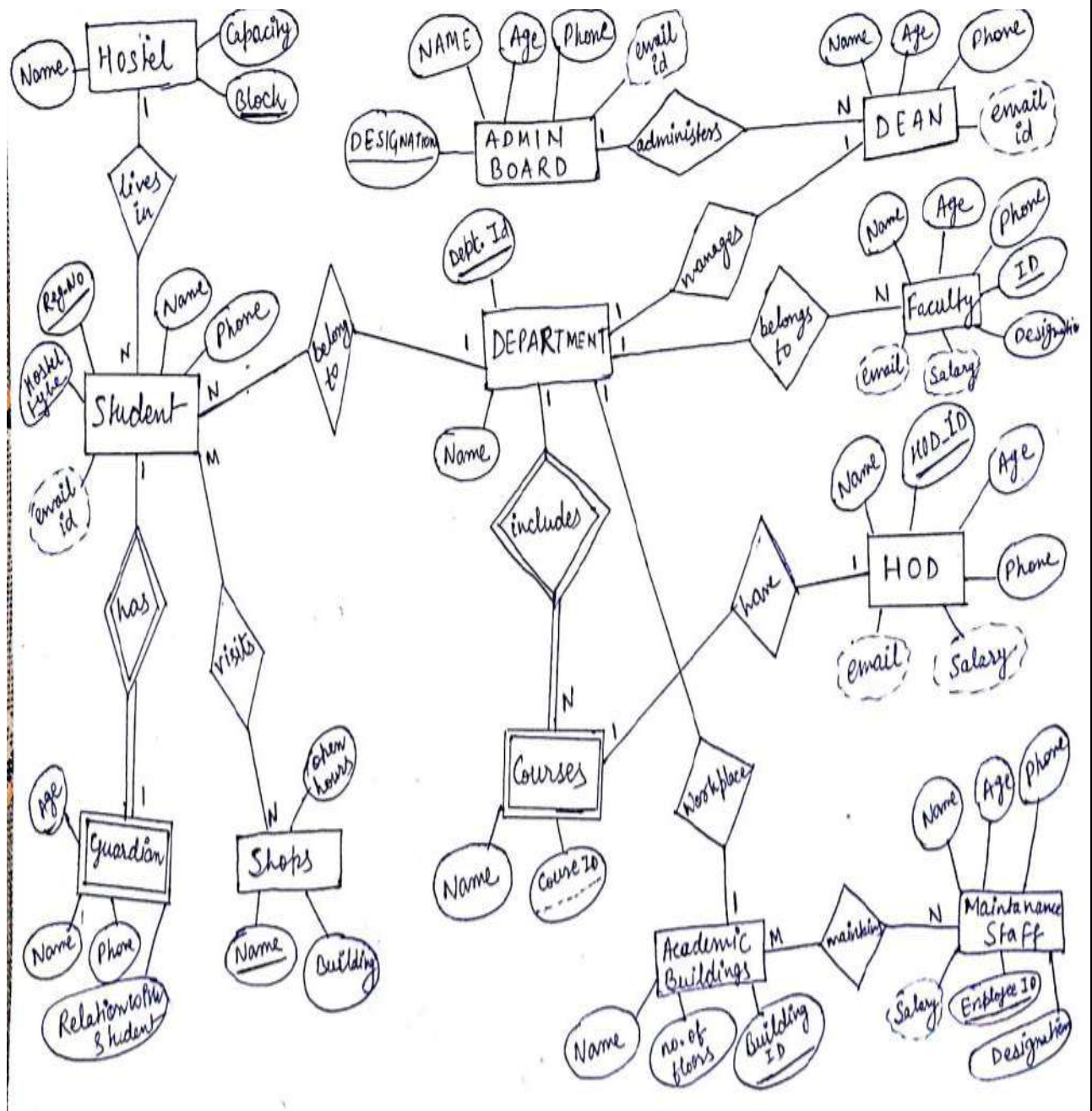
Relational Schema

STUDENT:

Reg. No	Name	Hostel Type	Phone	Email Id	Dept. Id	Block

Reg. No → Primary Key

Dept. Id, Block → foreign key

HOSTEL:

Name	Block	Capacity

Block → Primary Key.

Shops:

Name	Building	Open hours

Name → Primary Key

Visits:

Reg. No	Shop Name

Reg. No, Shop Name → foreign key

Department:

Dept. Name	Dept. Id
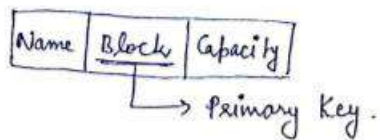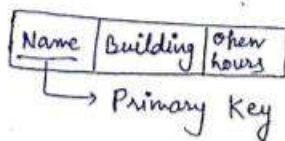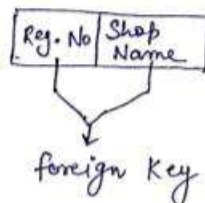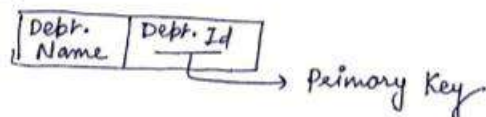
Dept. Id → Primary Key

Dean

Name	Age	Designation"	Phone	email

→ foreign key

Faculty :

Faculty Id	Name	Age	Designation	Salary	Phone	email	Dept. Id

→ Primary Key

foreign key

Academic Building :

Name	Building Id	no. of floors	Dept. Id

→ Primary Key

→ foreign key

Maintenance Staff :

Name	Age	Employee Id	Phone	Designation	Salary

→ Primary Key.

Maintains :

Building Id	Employee Id

→ foreign key

→ foreign key

HOD:

Name	HOD ID	Age	Phone	Email	Salary

→ Primary Key.

Guardian:

Name	Age	Phone	Reg.No.	Relation With Student

→ foreign Key

Courses:

Course Id	HOD Id	Name

→ Partial Key   → foreign Key

Admin Board:

Name	Age	Phone	Email	Designation

→ Primary Key.

SQL TABLE CREATION

Tables that we have created:

collegemanagementsystem
- Tables
  - academic building
  - administration board
  - course
  - dean
  - department
  - faculty
  - guardian
  - hod
  - hostel
  - maintainence staff
  - maintains
  - shop
  - studentguardian
  - visit
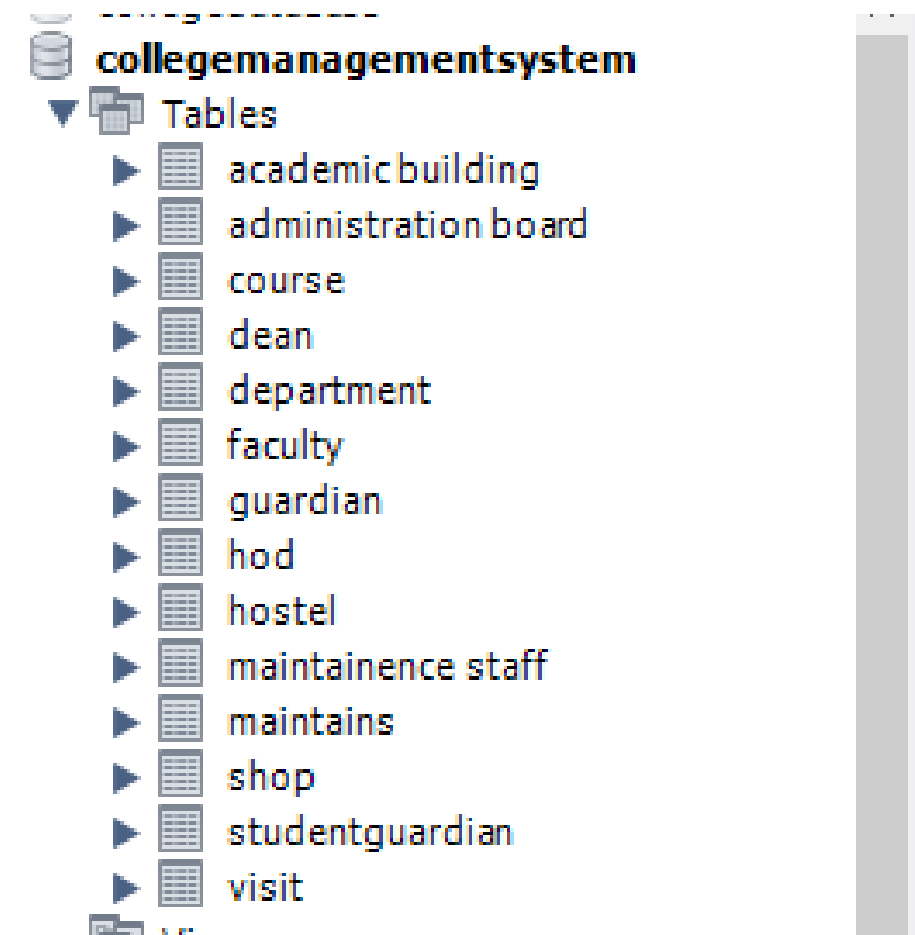
Table creation code for each table:

1. <u>Academic building:</u>

```
CREATE TABLE `academic building` (
  `BuildingName` varchar(50) NOT NULL,
  `BuildingID` varchar(45) NOT NULL,
  `Departmentallotted` varchar(45) NOT NULL,
  `No.of_floors` varchar(45) NOT NULL,
  PRIMARY KEY (`BuildingID`),
  KEY `department ID` (`Departmentallotted`)
)
```

2. <u>Administration Board:</u>

```
CREATE TABLE `administration board` (
  `Name` varchar(50) NOT NULL,
  `Age` varchar(45) NOT NULL,
  `PhoneNumber` varchar(45) NOT NULL,
  `Designation` varchar(45) NOT NULL,
  `Email_ID` varchar(45) NOT NULL,
  PRIMARY KEY (`Designation`)
)
```

3. <u>Course</u>

```
CREATE TABLE `course` (
  `CourseName` varchar(50) NOT NULL,
  `CourseID` varchar(45) NOT NULL,
  `Credits` varchar(45) NOT NULL,
  `HOD_ID` varchar(45) NOT NULL,
  PRIMARY KEY (`CourseID`),
  KEY `hod id` (`HOD_ID`) /*!80000 INVISIBLE */,
  CONSTRAINT `HOD ID` FOREIGN KEY (`HOD_ID`) REFERENCES
`hod` (`HOD_ID`) ON DELETE CASCADE ON UPDATE CASCADE
)
```

4. <u>Dean</u>

```sql
CREATE TABLE `dean` (
  `Name` varchar(50) NOT NULL,
  `Age` varchar(45) NOT NULL,
  `Department` varchar(45) NOT NULL,
  `PhoneNumber` varchar(45) NOT NULL,
  `Email_ID` varchar(45) NOT NULL,
  `AdimnBoardDesignation` varchar(45) NOT NULL,
  PRIMARY KEY (`Department`),
  KEY `AB_designation` (`AdimnBoardDesignation`),
  CONSTRAINT `Admin Board Designation` FOREIGN KEY
(`AdimnBoardDesignation`) REFERENCES `administration board`
(`Designation`) ON DELETE CASCADE ON UPDATE CASCADE
)
```

5. <u>Department</u>

```sql
CREATE TABLE `department` (
  `DepartmentName` varchar(50) NOT NULL,
  `DepartmentID` varchar(45) NOT NULL,
  `DeanDepartment` varchar(45) NOT NULL,
  PRIMARY KEY (`DepartmentID`),
  KEY `dean department` (`DeanDepartment`) /*!80000 INVISIBLE */,
  CONSTRAINT `dean department` FOREIGN KEY
(`DeanDepartment`) REFERENCES `dean` (`Department`) ON DELETE
CASCADE ON UPDATE CASCADE
)
```

6. <u>Faculty</u>

```sql
CREATE TABLE `faculty` (
  `Name` varchar(50) NOT NULL,
  `FacultiID` varchar(45) NOT NULL,
  `Age` varchar(45) NOT NULL,
  `PhoneNumber` varchar(45) NOT NULL,
  `Designation` varchar(45) NOT NULL,
  `Email_ID` varchar(45) NOT NULL,
  `Salary` varchar(45) NOT NULL,
  `Department_ID` varchar(45) NOT NULL,
  PRIMARY KEY (`FacultiID`),
  KEY `Department ID` (`Department_ID`) /*!80000 INVISIBLE */
)
```

7. <u>Guardian</u>

```
CREATE TABLE `guardian` (
    `Guardian Name` varchar(45) NOT NULL,
    `Age` varchar(45) NOT NULL,
    `Phone Number` varchar(45) NOT NULL,
    `Relation with Student` varchar(45) NOT NULL,
    `Student registration number` varchar(45) NOT NULL,
    PRIMARY KEY (`Phone Number`),
    KEY `student reg no.` (`Student registration number`)
)
```

8. <u>HOD</u>

```
CREATE TABLE `hod` (
    `Name` varchar(50) NOT NULL,
    `HOD_ID` varchar(45) NOT NULL,
    `Age` varchar(45) NOT NULL,
    `Course_ID` varchar(45) NOT NULL,
    `PhoneNumber` varchar(45) NOT NULL,
    `Email_ID` varchar(45) NOT NULL,
    PRIMARY KEY (`HOD_ID`)
)
```

9. <u>Hostel</u>

```
CREATE TABLE `hostel` (
    `Name` varchar(50) NOT NULL,
    `Block` varchar(45) NOT NULL,
    `Capacity` varchar(45) NOT NULL,
    PRIMARY KEY (`Block`)
)
```

10. <u>Maintainance Staff</u>

```
CREATE TABLE `maintainence staff` (
    `Name` varchar(50) NOT NULL,
    `Employee_ID` varchar(45) NOT NULL,
    `Age` varchar(45) NOT NULL,
    `Workplace` varchar(45) NOT NULL,
    `PhoneNumber` varchar(45) NOT NULL,
    `Designation` varchar(45) NOT NULL,
    `Salary` varchar(45) NOT NULL,
    PRIMARY KEY (`Employee_ID`)
)
```

11.   <u>Maintains</u>

```
CREATE TABLE `maintains` (
  `AcademicBulidingID` varchar(50) NOT NULL,
  `MaintainenceStaffID` varchar(45) NOT NULL,
  KEY `AB ID` (`AcademicBulidingID`),
  KEY `MS ID` (`MaintainenceStaffID`),
  CONSTRAINT `AB ID` FOREIGN KEY (`AcademicBulidingID`)
REFERENCES `academic building` (`BuildingID`),
  CONSTRAINT `MS ID` FOREIGN KEY (`MaintainenceStaffID`)
REFERENCES `maintainence staff` (`Employee_ID`)
)
```

12.   <u>Shop</u>

```
CREATE TABLE `shop` (
  `ShopName` varchar(50) NOT NULL,
  `Building` varchar(45) NOT NULL,
  `OpenHours` varchar(45) NOT NULL,
  PRIMARY KEY (`ShopName`)
)
```

13.   <u>Student Guardian</u>

```
CREATE TABLE `studentguardian` (
  `Name` varchar(50) NOT NULL,
  `RegistrationNumber` varchar(45) NOT NULL,
  `Age` varchar(45) NOT NULL,
  `HostelType` varchar(45) NOT NULL,
  `MobileNumber` varchar(45) NOT NULL,
  `Department` varchar(45) NOT NULL,
  `HostelBlock` varchar(45) NOT NULL,
  `Email_ID` varchar(45) NOT NULL,
  `GuardianName` varchar(45) NOT NULL,
  `Guardian Age` varchar(45) NOT NULL,
  `Mobile` varchar(45) NOT NULL,
  `GUardianRelationWithStudent` varchar(45) NOT NULL,
  PRIMARY KEY (`RegistrationNumber`)
)
```

14.   <u>Visits</u>
```
CREATE TABLE `visit` (
  `StudentRegno.` varchar(50) NOT NULL,
  `ShopName` varchar(45) NOT NULL,
  KEY `reg no.` (`StudentRegno.`),
  KEY `Shop name` (`ShopName`),
  CONSTRAINT `Shop name` FOREIGN KEY (`ShopName`)
REFERENCES `shop` (`ShopName`),
  CONSTRAINT `Student Reg no.` FOREIGN KEY (`StudentRegno.`)
REFERENCES `studentguardian` (`RegistrationNumber`))
```

Entity description

1. Admin Board administers all the deans.
2. Every dean manages a single department.
3. Many faculties belong to a single department.
4. Every dept. Has several Courses (weak entity).
5. Each course has a HOD.
6. Every department has a workplace i.e. each department is allotted a building.
7. There are many staff to maintain all those buildings.
8. Every student belong to a single department.
9. Students are also provided with an option to stay in hostels.
10. Every student has a local guardian who can visit them in case of any emergency.
11. There are several shops available to students for meeting their demands.

GUI Implementation

CODE LINKS-

GitHub Link- https://github.com/karuneshtripathi13/DBMS-Project.git

Or

Google Drive Link-

https://drive.google.com/drive/folders/1uEXObpVoFvNadHbU_qw5SsrYNZ6bTXr6?usp=sharing

GUI Implementation Output

HOME SCREEN
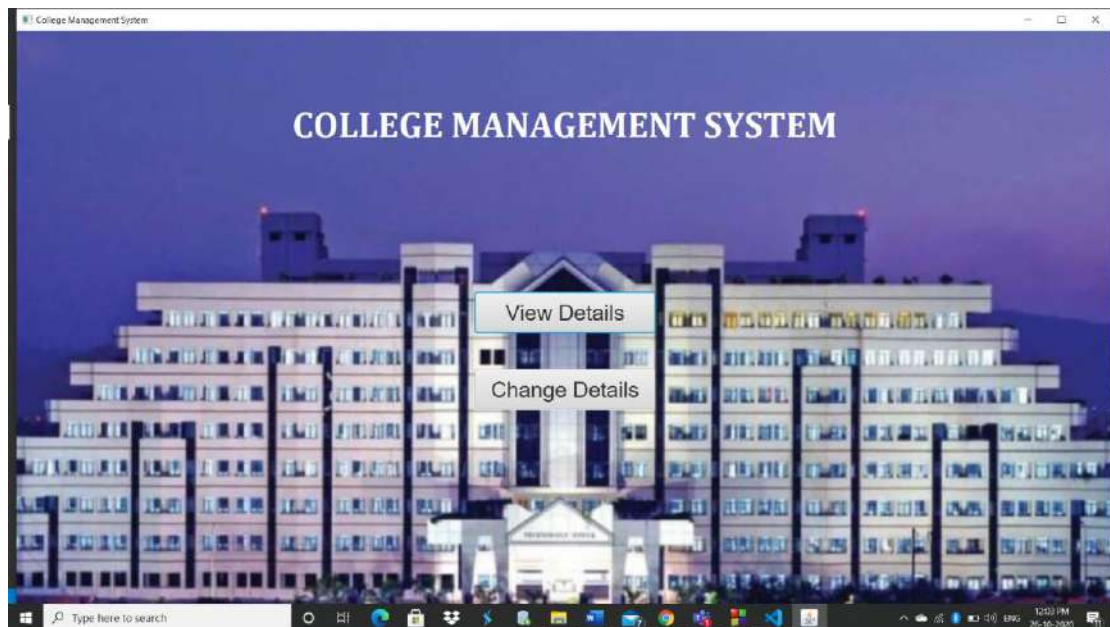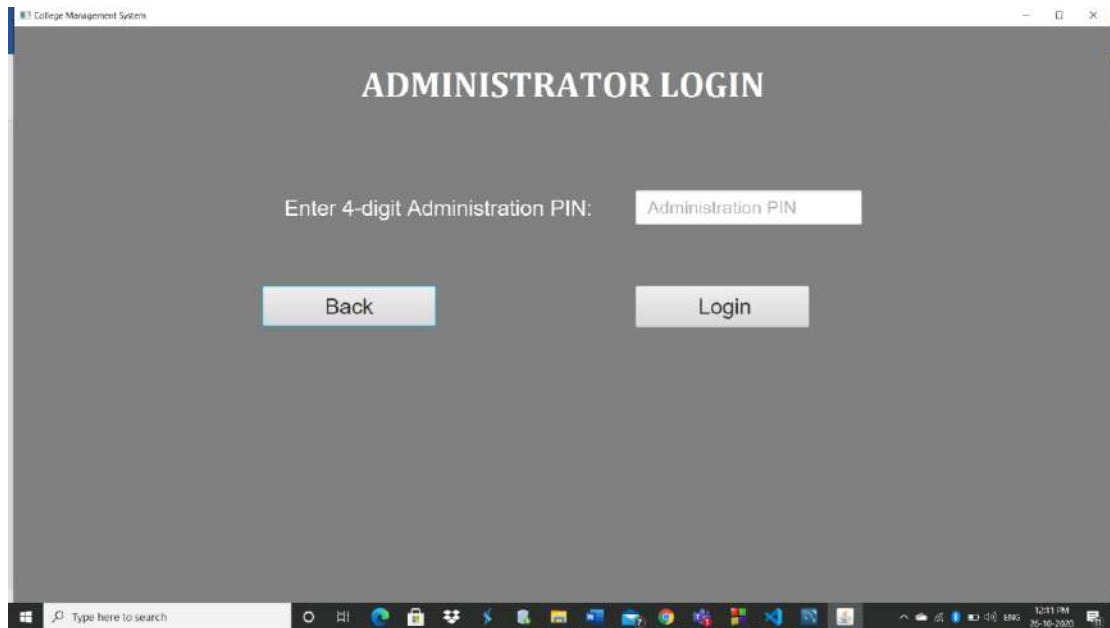
- To Change Details in Data Base Click on Change Details. It can only be done by Administrator.
- To View Details in Data Base Click on View Details. Both Student and Administrator can view details.

CHANGE DETAILS

Details can only be changed by Administrator.
- This ensures that data is not manipulated by an unauthorised person.
- Ensures security of Information in Data Base.
- In our project the Administration PIN is 0880
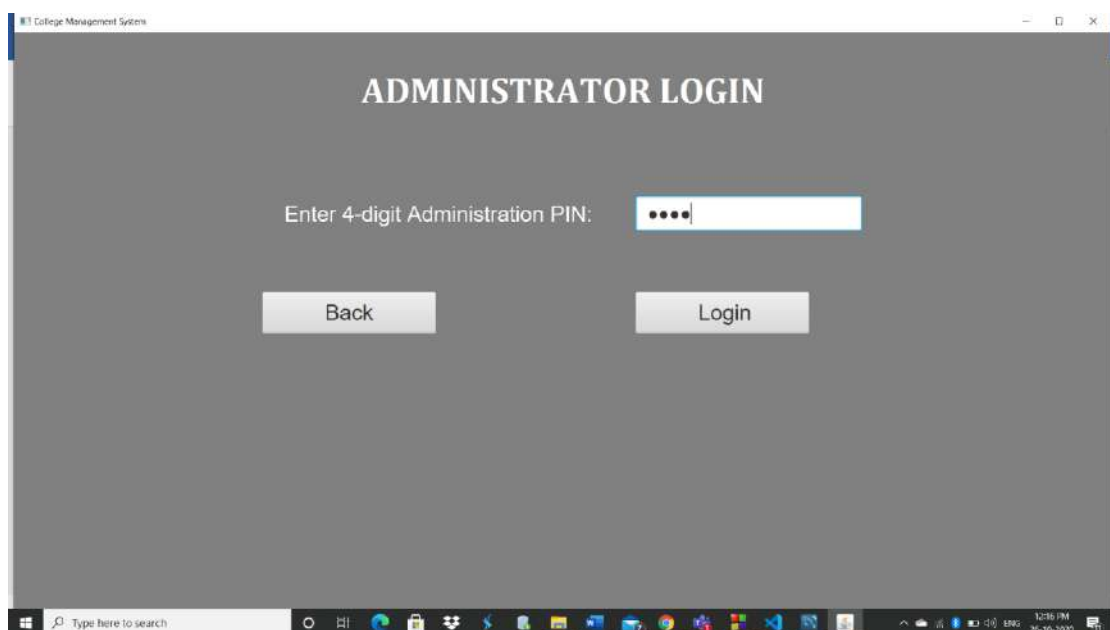
- We can add (Register), Update or Delete details from data base as administrator.
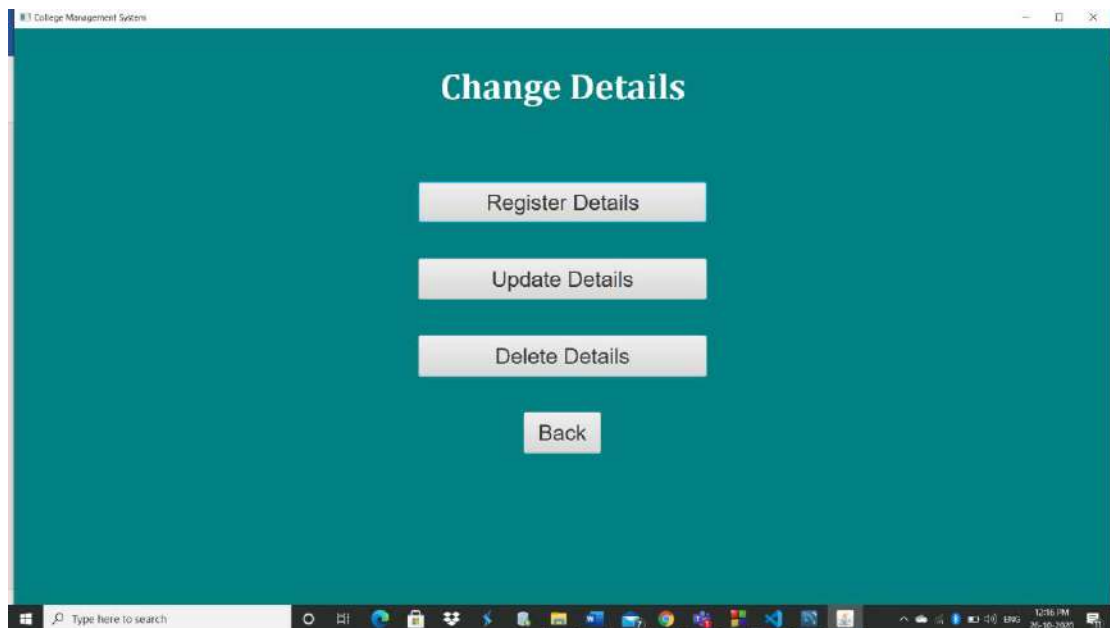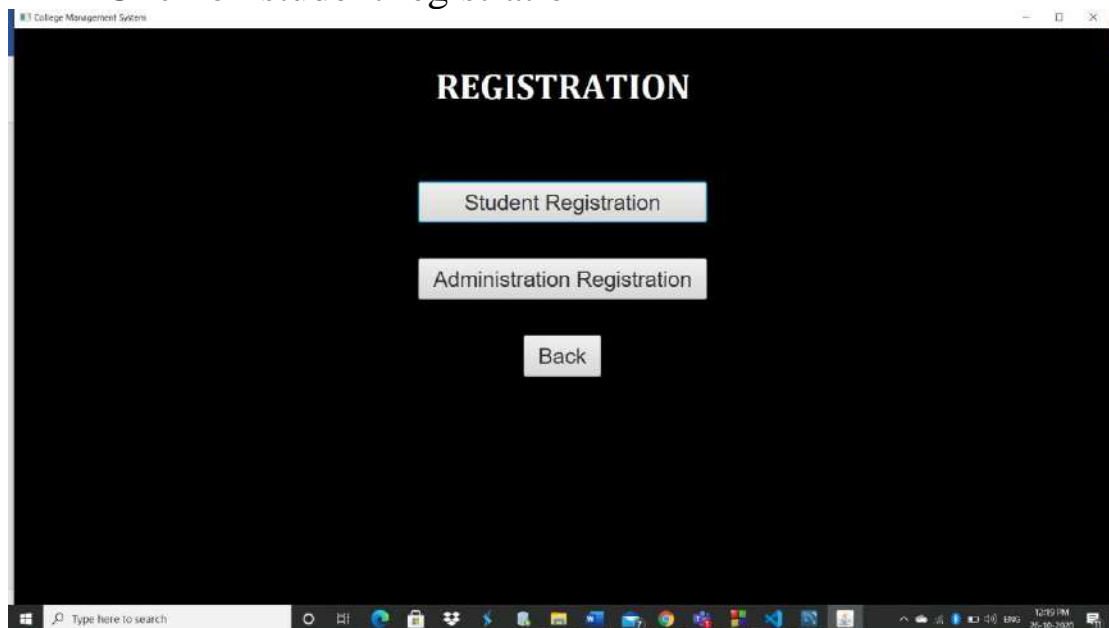
REGISTER DETAILS

- We can Register Details of Students
- We can Register Details of Administration
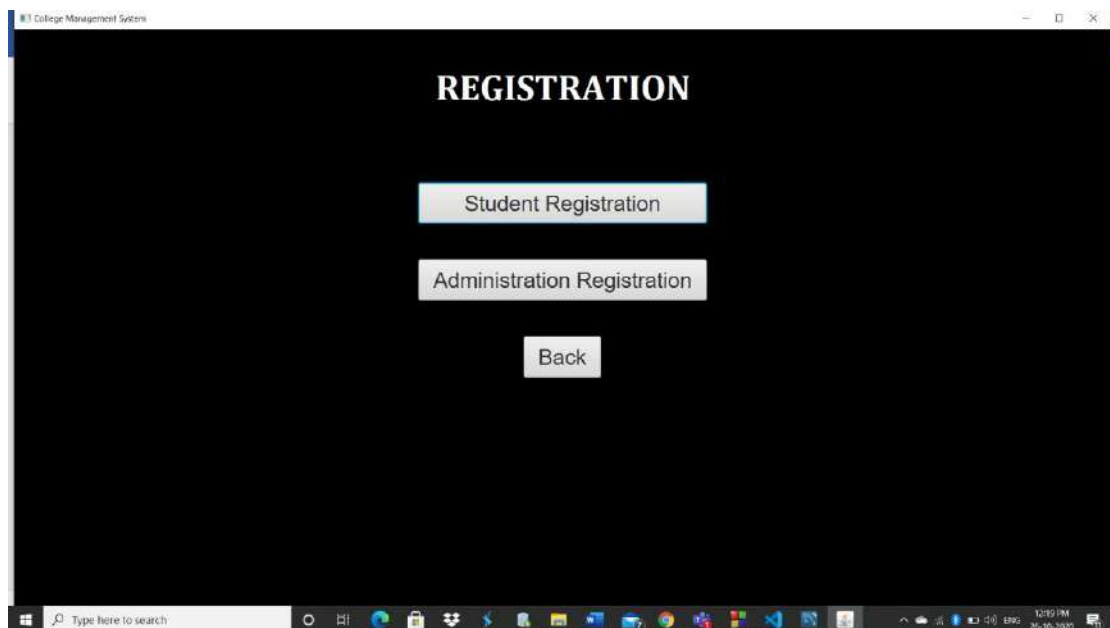
STUDENT REGISTRATION-

- Click on student registration

- Fill the details
- Email will automatically get generated is our system
- Click on Register

- Pop up window shows successful registration if all details are valid

- If we write Invalid registration number then pop up window will show message as shown below

- If we try to register student which is already registered then pop up window will show message as shown below

- The email id automatically appear on successful registration in Student Email ID column.

ADMINISTRATION REGISTRATION-

- Click on Administration Registration

- Choose the field you want to register.
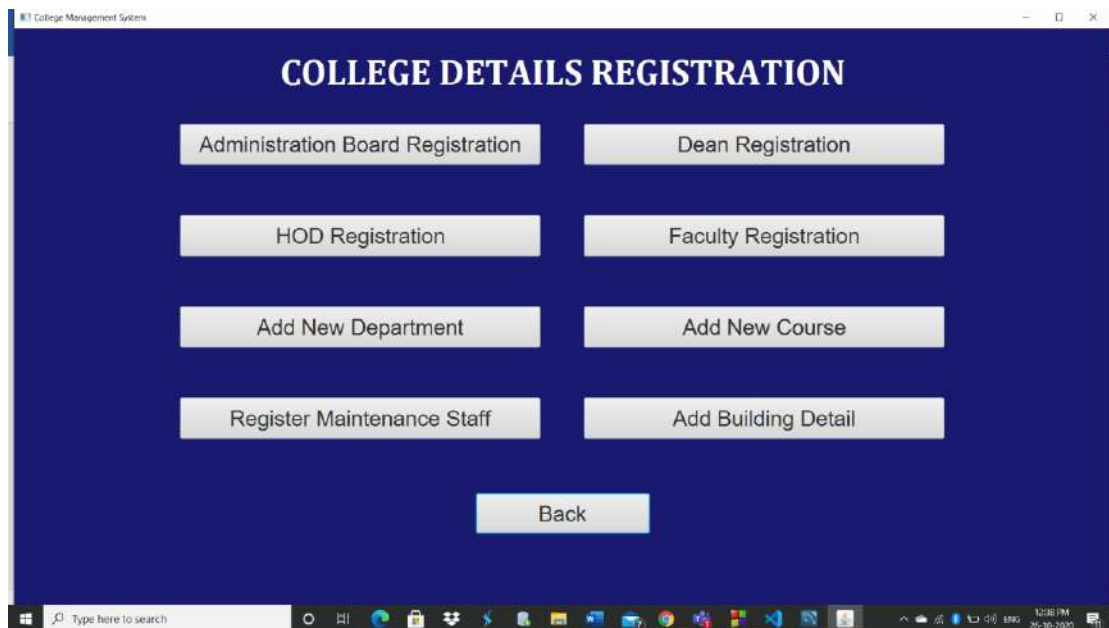
1. ADMINISTRATION BOARD MEMBER REGISTRATION

- Click on Administration Board Registration

- Fill the details
- Email will automatically get generated is our system
- Click on Register
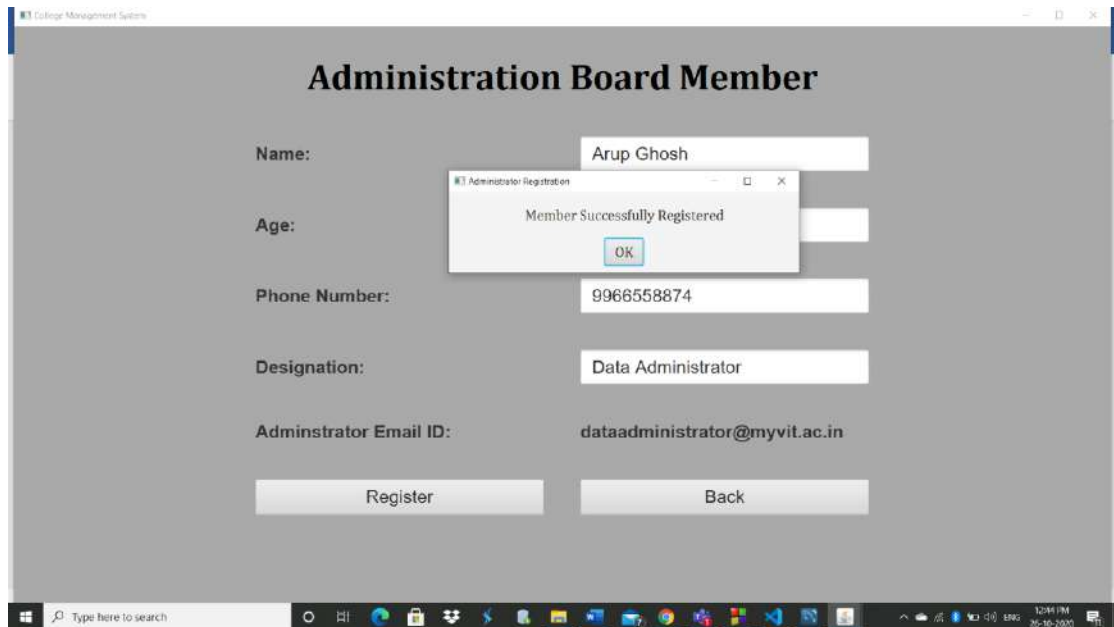
- Pop up window shows successful registration if all details are valid. Email ID will appear in Administrator Email ID column.
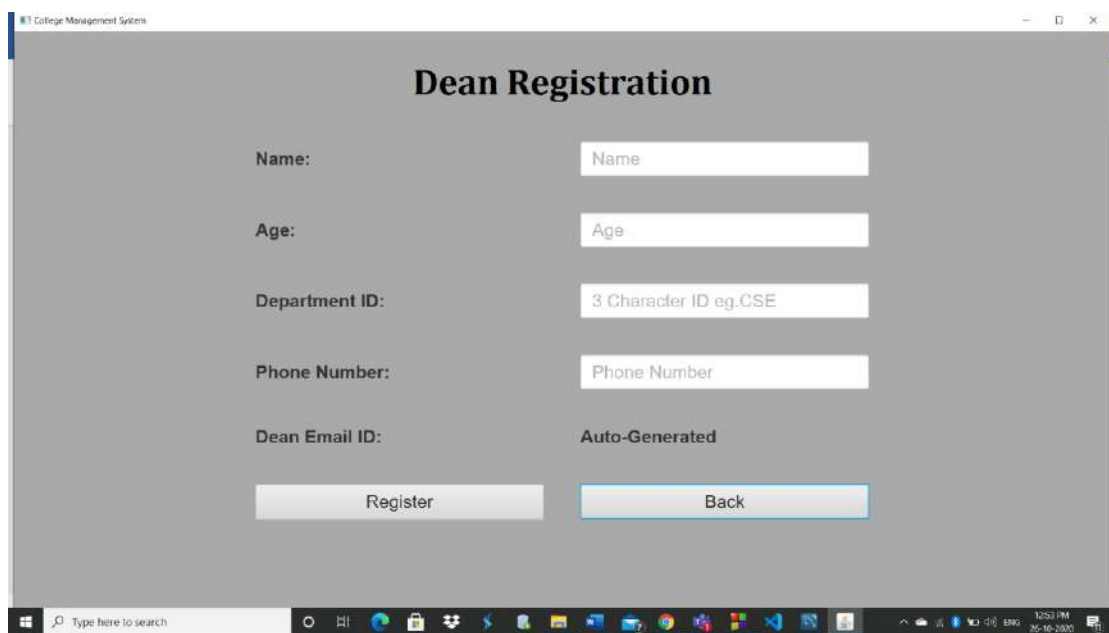
- If we try to register board member which is already registered then pop up window will show message as shown below

2. DEAN REGISTRATION

- Click on Dean Registration

- Fill the details
- Email will automatically get generated is our system
- Click on Register

- Pop up window shows successful registration if all details are valid

- Email ID will appear when we click on OK button

- If we try to register dean which is already registered then pop up window will show message as shown below

- If we enter invalid Department ID then pop up window will show message as shown below

3. HOD REGISTRATION

- Click on HOD Registration

- Fill the details
- HOD ID , HOD Email ID , Salary will automatically get generated is our system
- Click on Register

- Pop up window shows successful registration if all details are valid

- Email ID , HOD ID and Salary will appear when we click on OK button

- If we try to register HOD which is already registered then pop up window will show message as shown below

4. FACULTY REGISTRATION

- Click on Faculty Registration

- Fill the details
- Email and Salary will automatically get generated is our system
- Click on Register

- Pop up window shows successful registration if all details are valid

- Email ID and Salary will appear when we click on OK button

- If we try to register faculty which is already registered then pop up window will show message as shown below

- If we enter invalid Faculty ID then pop up window will show message as shown below

5. ADD NEW DEPARTMENT IN COLLEGE

- Click on Add New Department

- Fill the details
- Click on Register

- Pop up window shows successful registration if all details are valid

- If we try to add department which is already registered then pop up window will show message as shown below
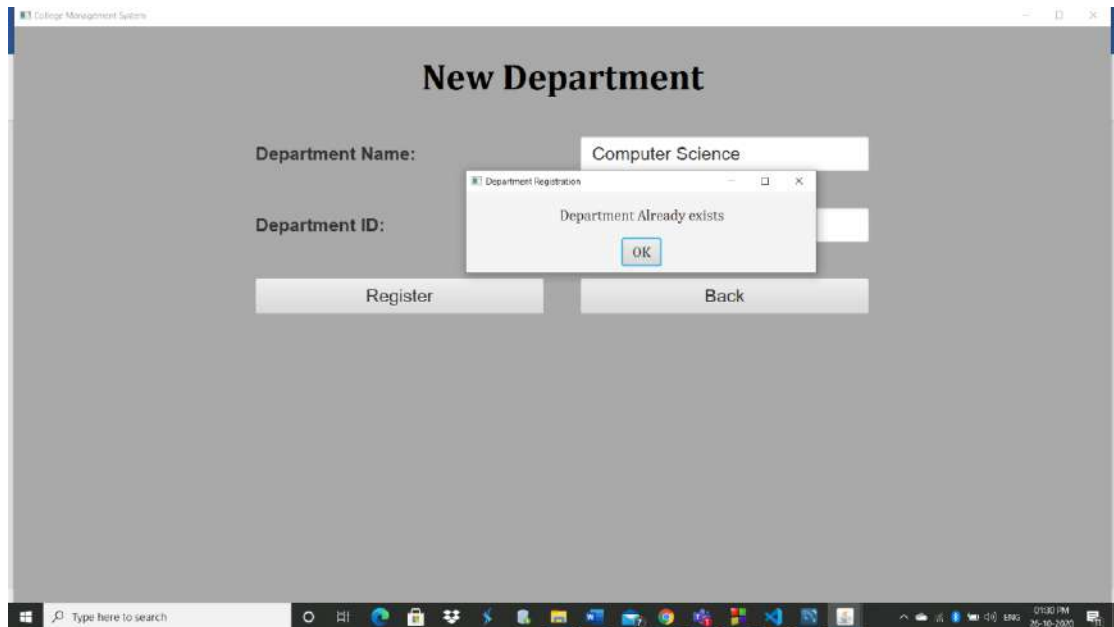

- Dean acts as Foreign key in Department
- If we try to add Department which has no Dean registered then pop up window will show message as shown below
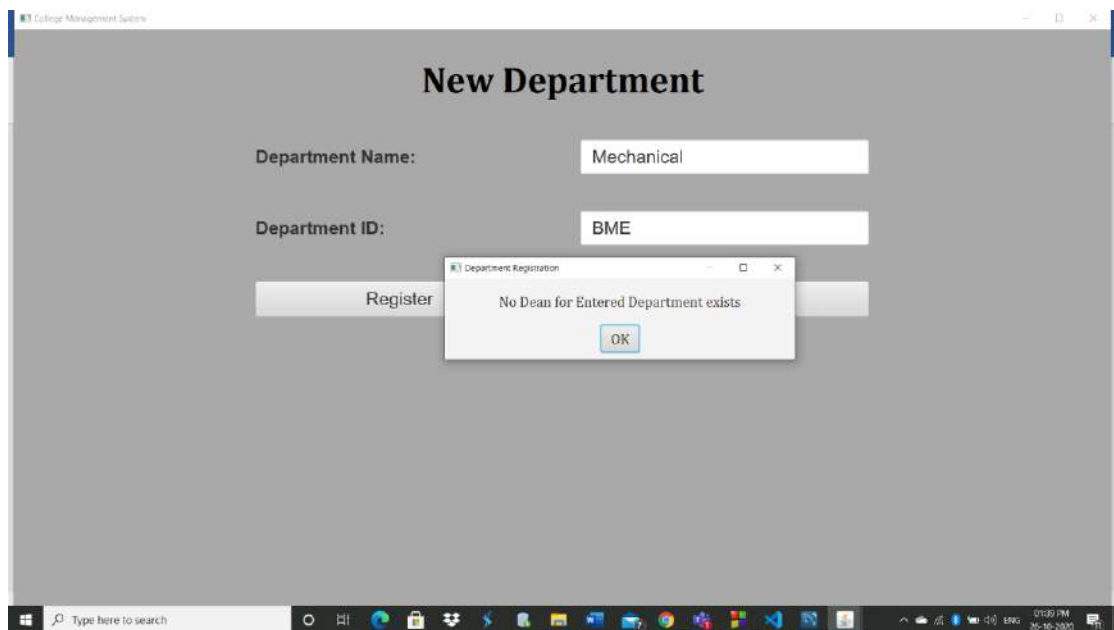
- If we enter invalid Department ID then pop up window will show message as shown below

6. ADD NEW COURSE IN COLLEGE

- Click on Add New Course

- Fill the details
- Click on Register

- Pop up window shows successful registration if all
  details are valid
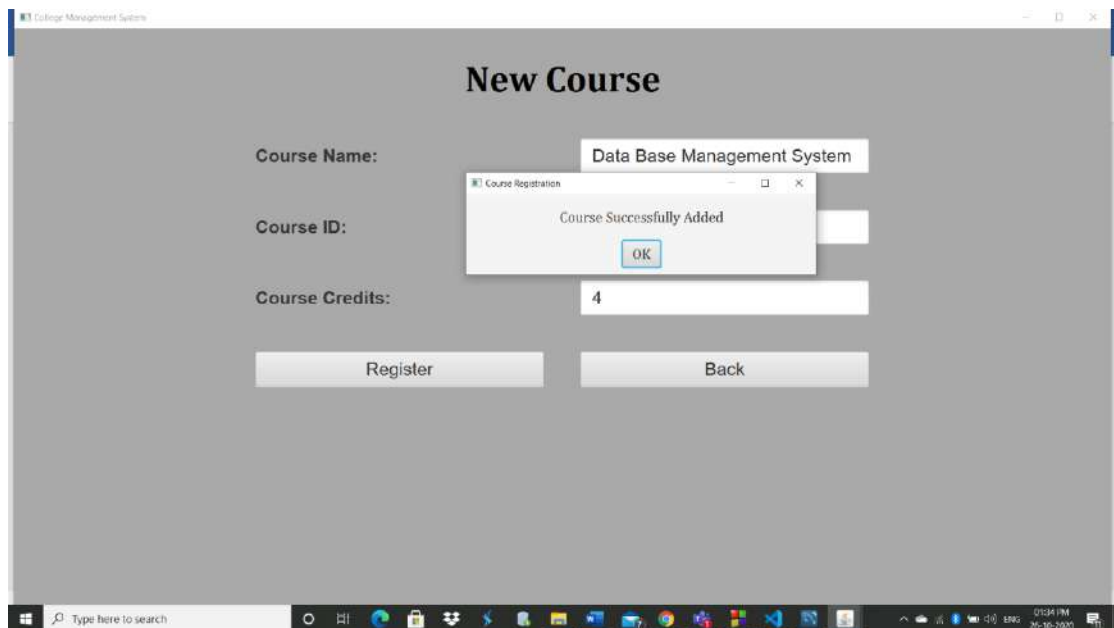
- If we try to add Course which is already registered then
  pop up window will show message as shown below
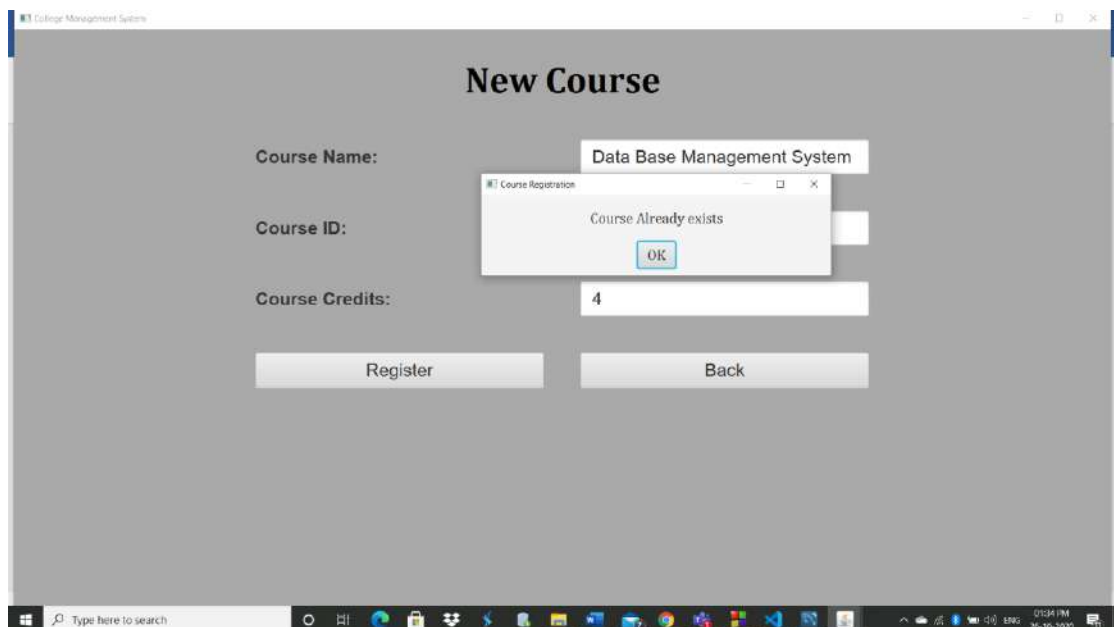
- HOD acts as Foreign key in Course
- If we try to add Course which has no HOD registered then pop up window will show message as shown below
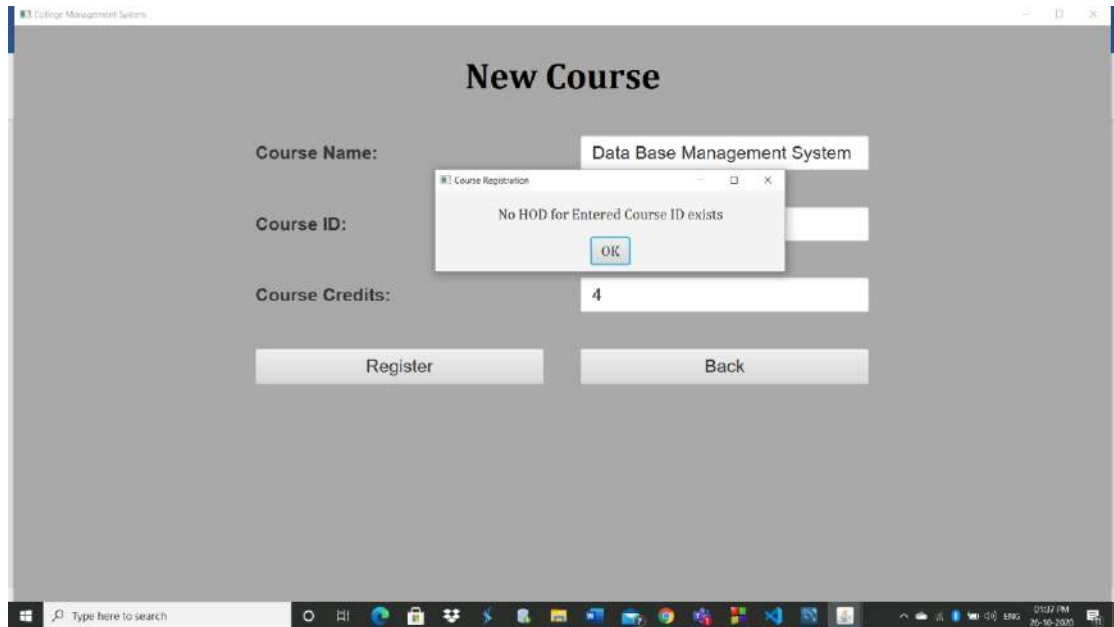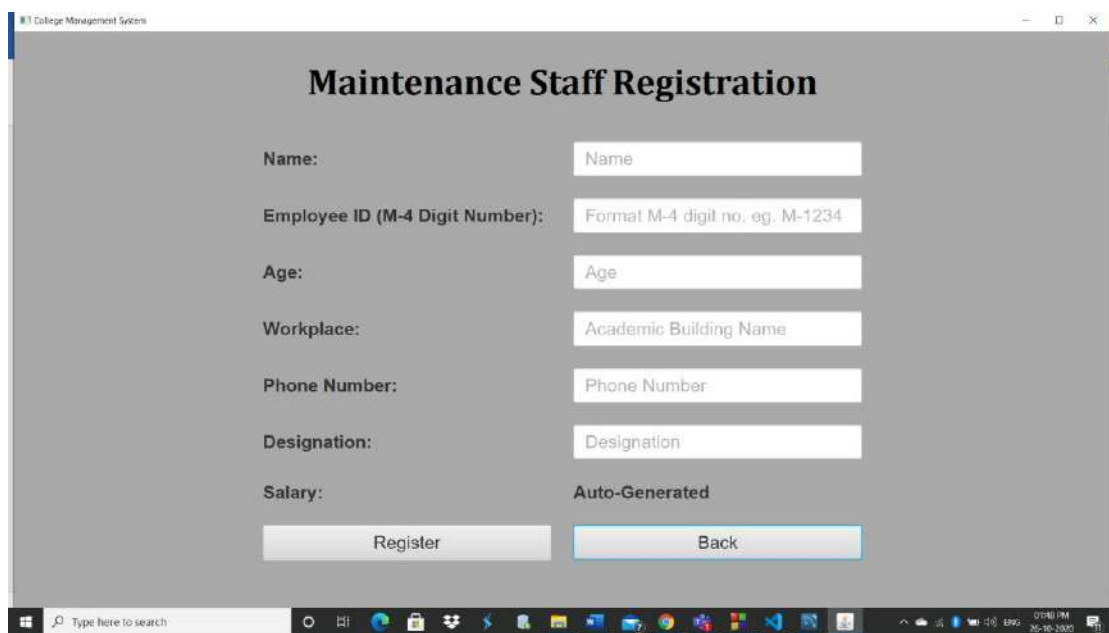
- If we enter invalid Course ID then pop up window will show message as shown below
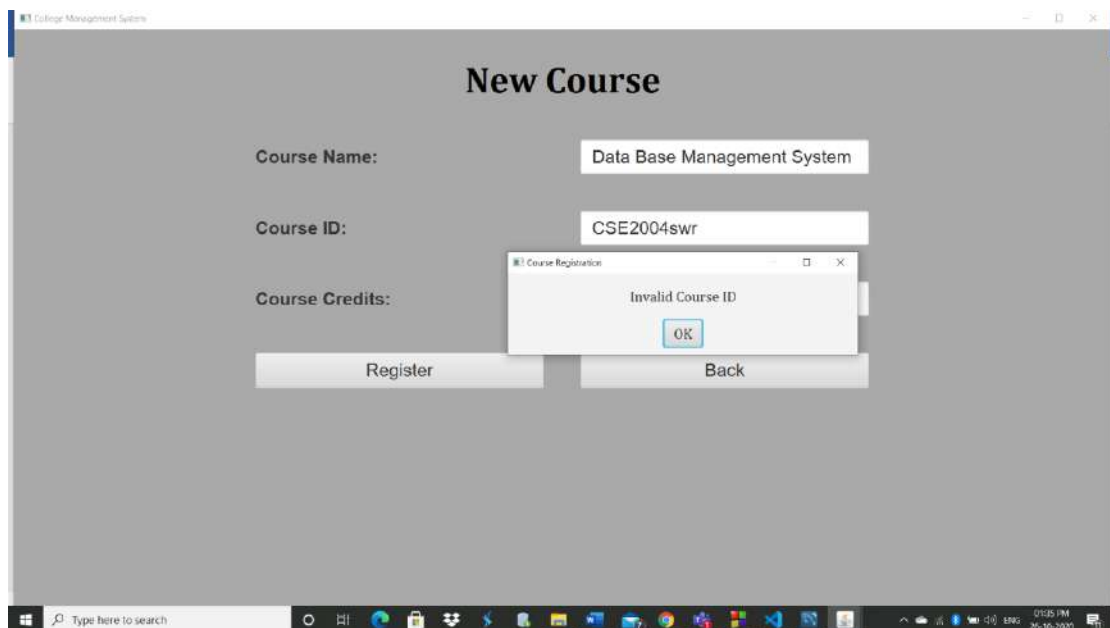
7. MAINTENANCE STAFF REGISTRATION

- Click on Register Maintenance Staff

- Fill the details
- Salary will automatically get generated is our system
- Click on Register

- Pop up window shows successful registration if all details are valid

- Salary will appear when we click on OK button

- If we try to register faculty which is already registered then pop up window will show message as shown below
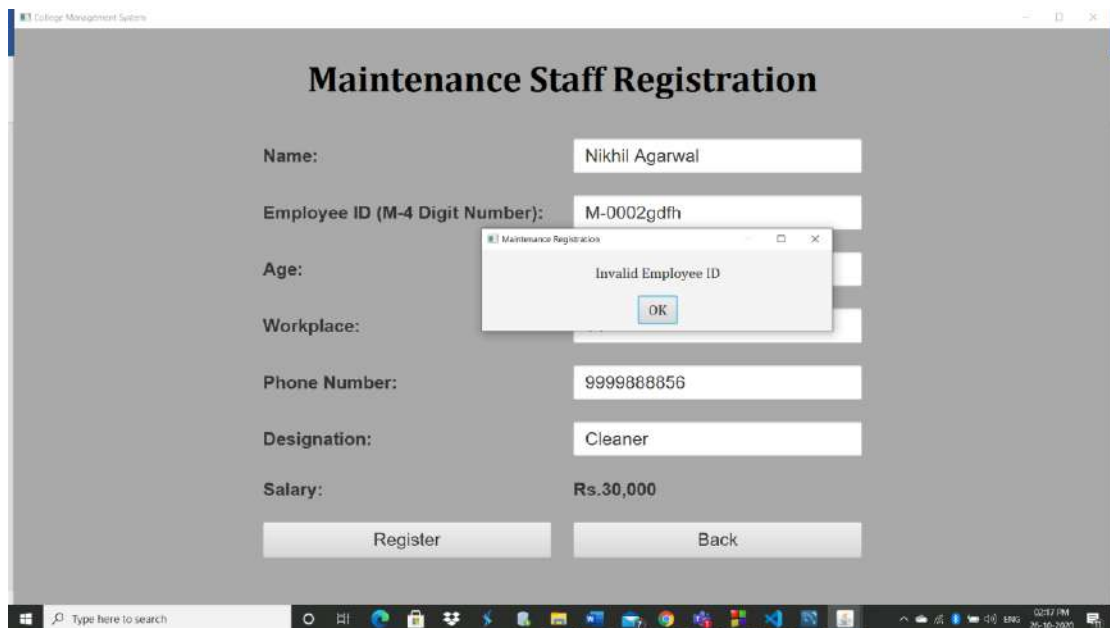
- If we enter invalid Employee ID then pop up window will show message as shown below

8. ADD NEW ACADMIC BUILDING IN COLLEGE

- Click on Add Building Detail

- Click on Add New Academic Building
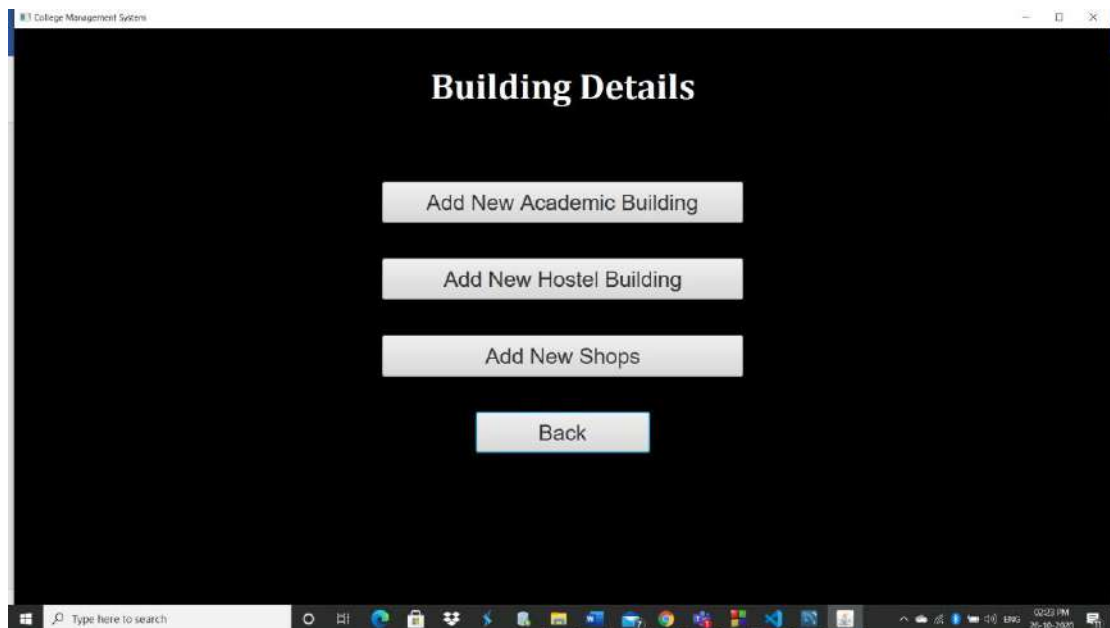
- Fill the details
- Building ID will be automatically generated by our system
- Click on Add Building

- Pop up window shows successful registration if all details are valid

- Building ID will appear when we click on OK button

- If we try to add Building which is already registered then pop up window will show message as shown below

9. ADD NEW HOSTEL BUILDING IN COLLEGE

- Click on Add Building Detail
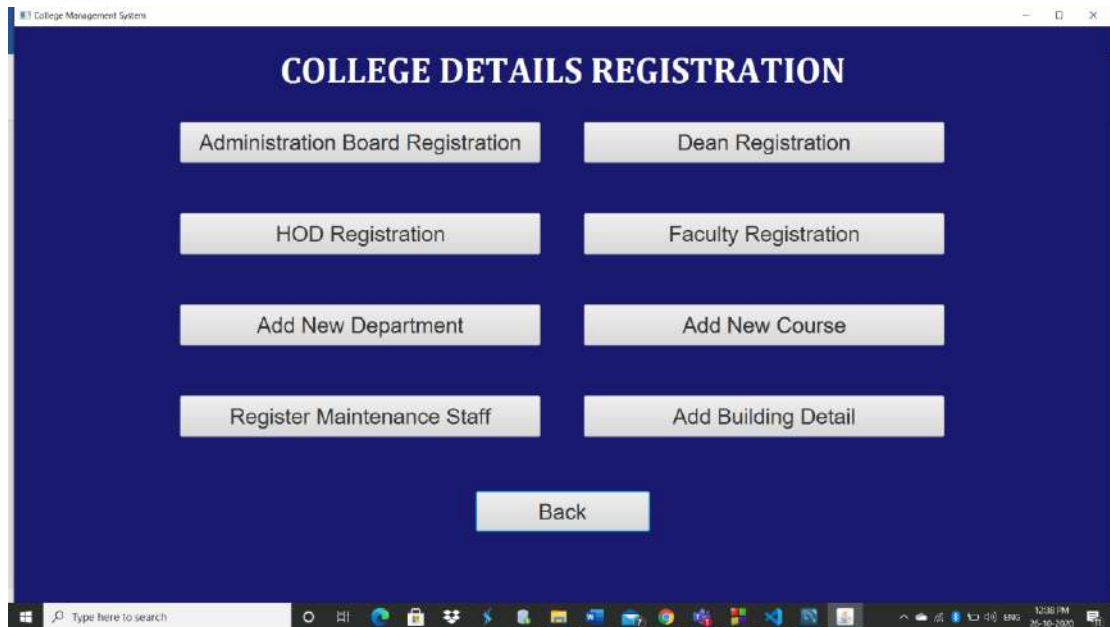
- Click on Add New Hostel Building
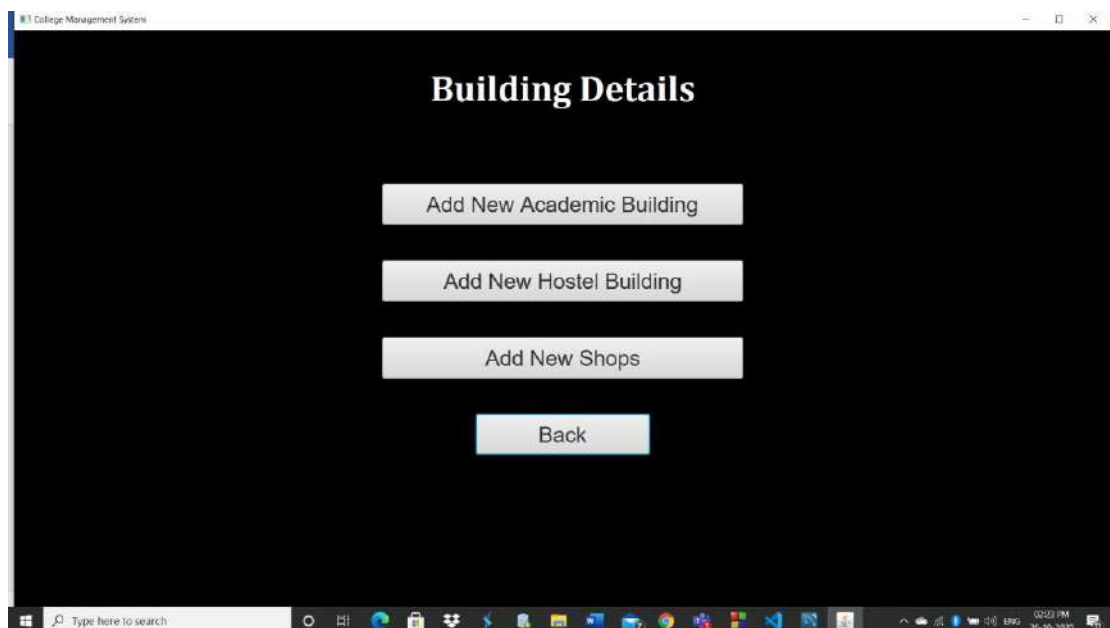
- Fill the details
- Click on Add Hostel

- Pop up window shows successful registration if all details are valid
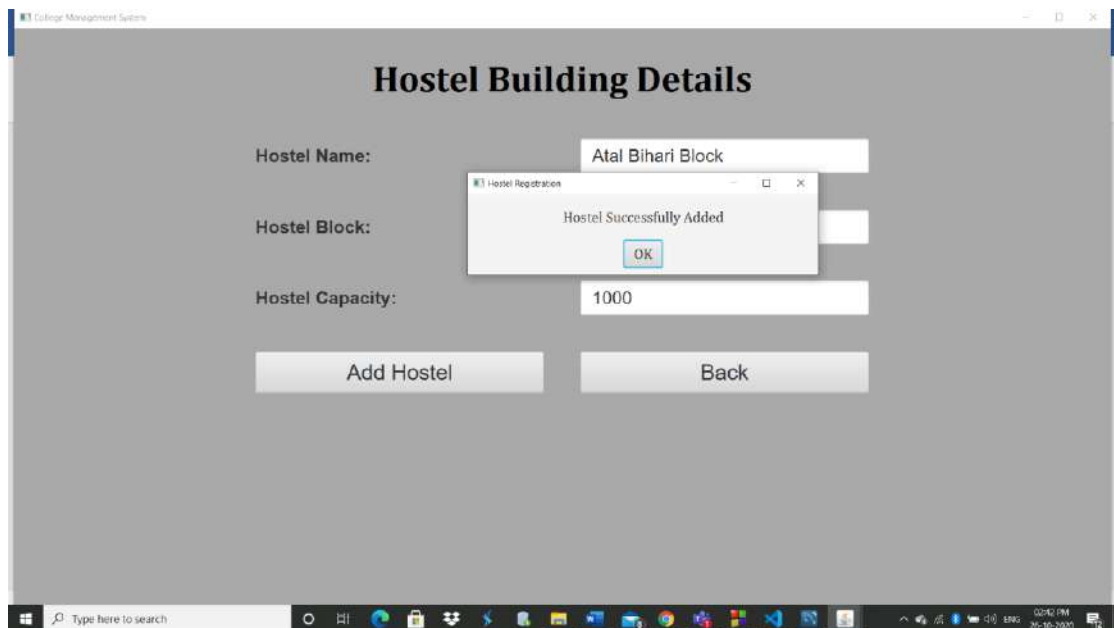
- If we try to add Hostel which is already registered then pop up window will show message as shown below
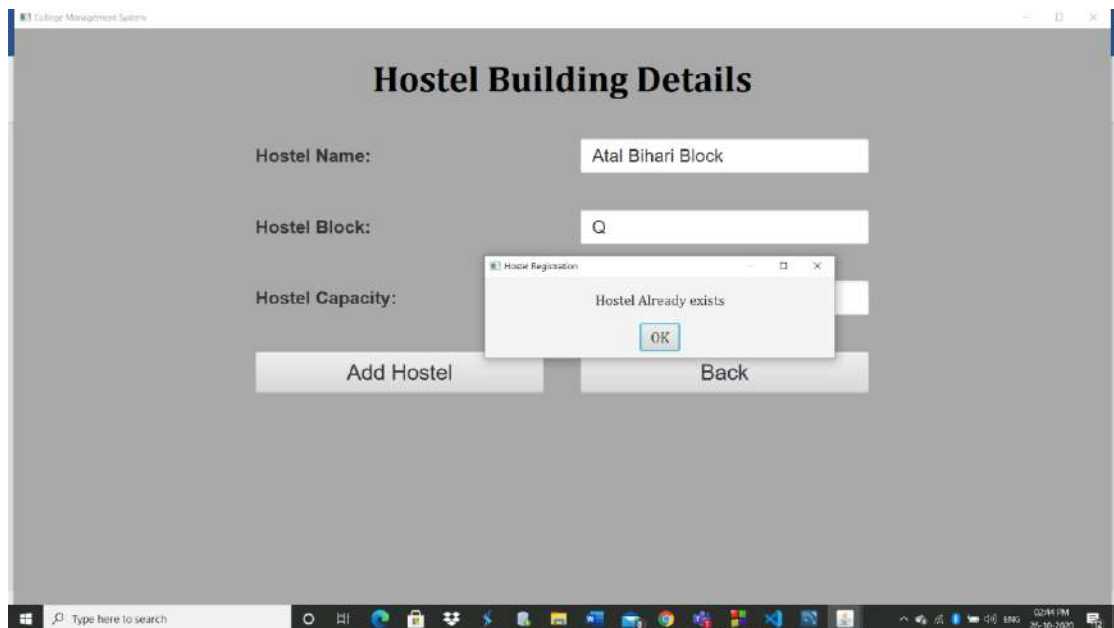
10. ADD NEW SHOP IN COLLEGE

- Click on Add Building Detail
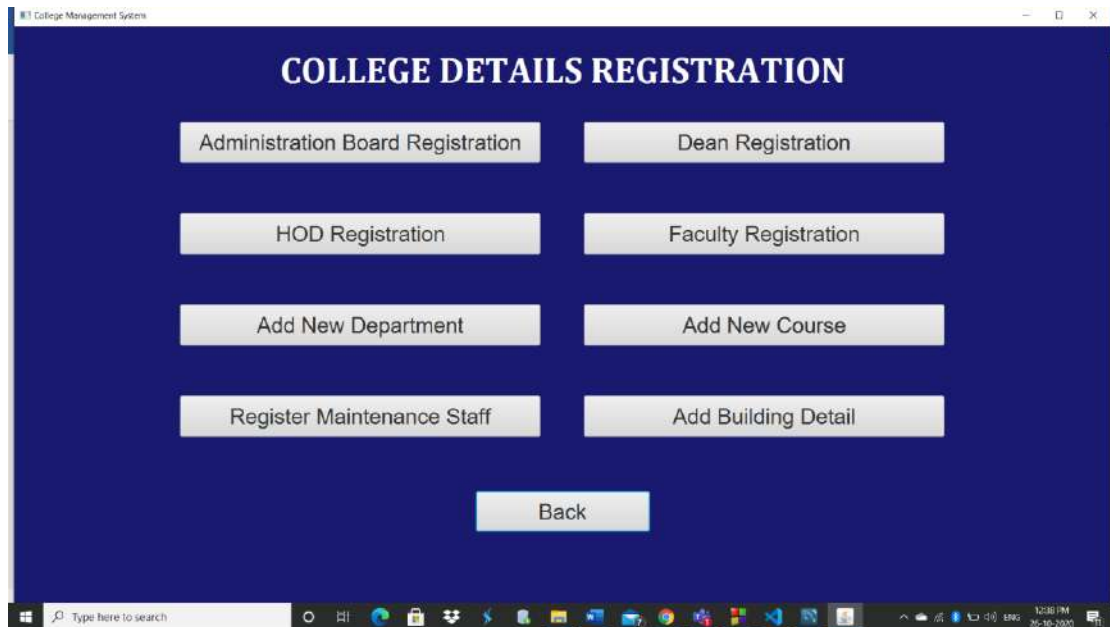
- Click on Add New Shops
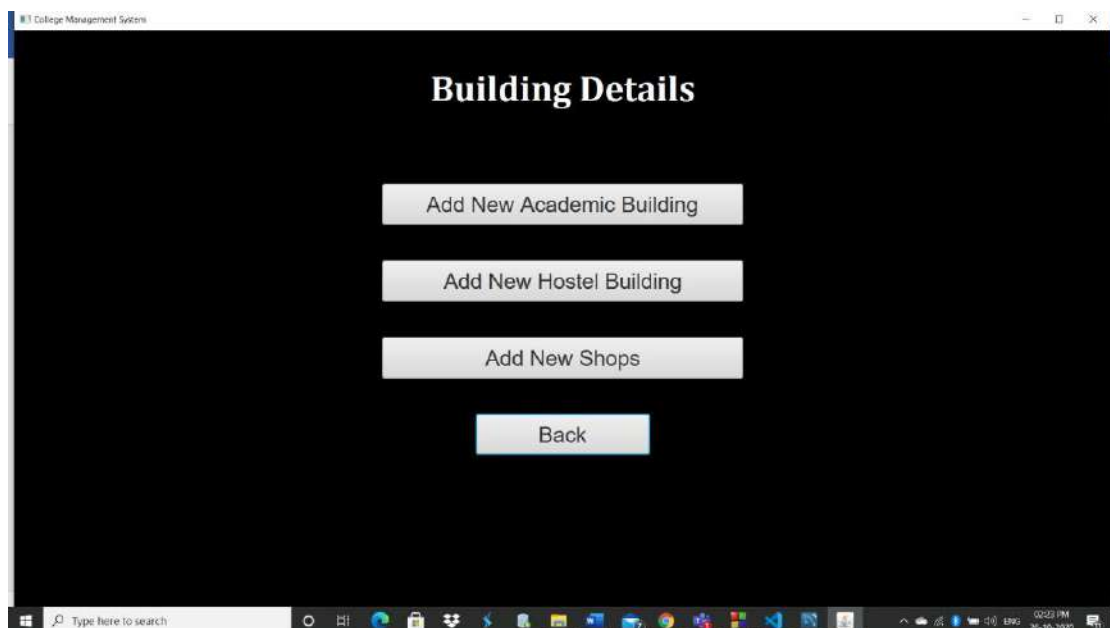
- Fill the details
- Click on Add Shop

- Pop up window shows successful registration if all details are valid
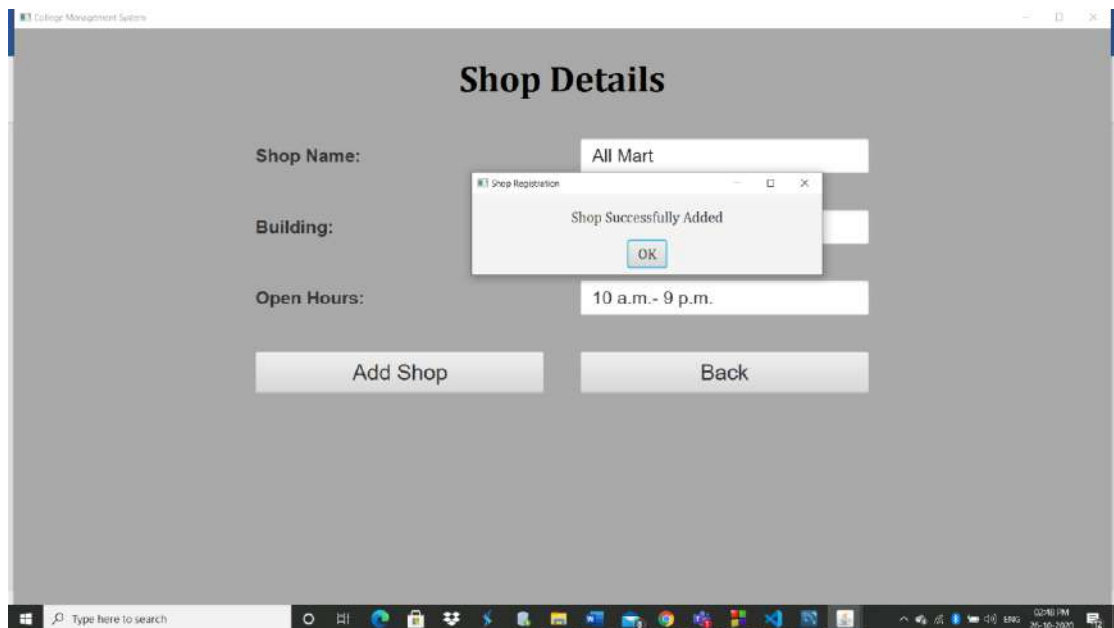
- If we try to add Shop which is already registered then pop up window will show message as shown below
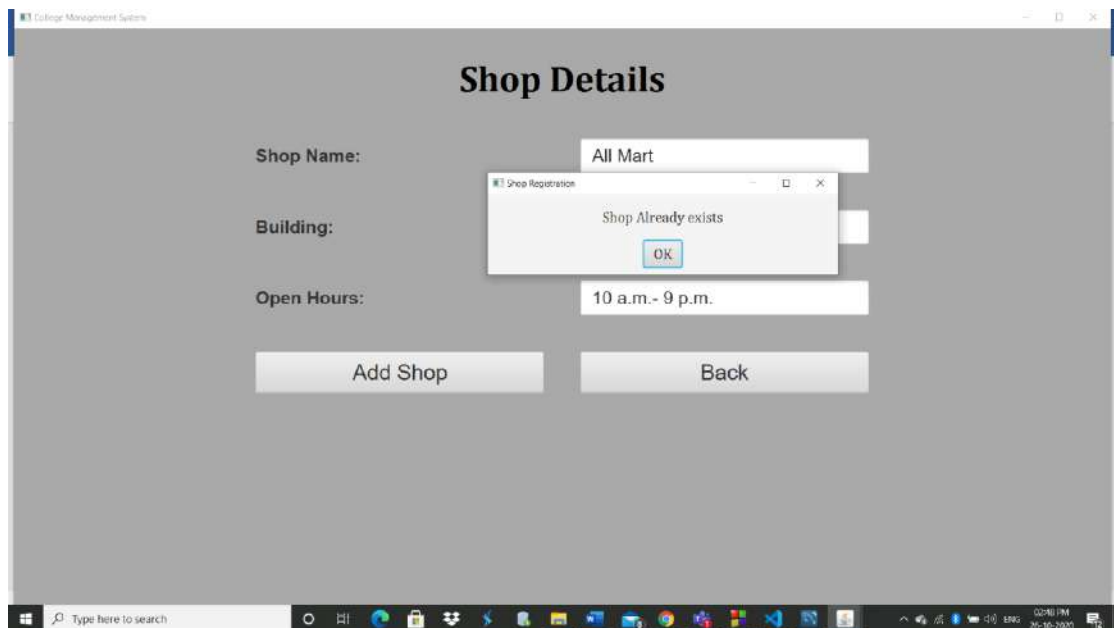
UPDATE DETAILS

- Click on Update Details
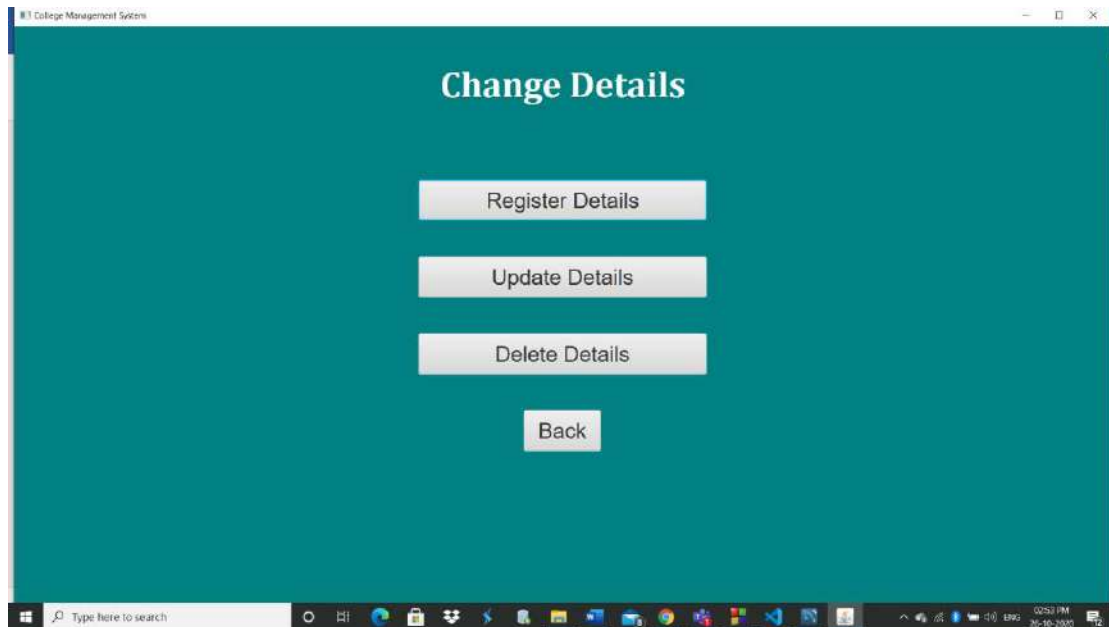
1. UPDATE STUDENT DETAILS

- Click on Update Student
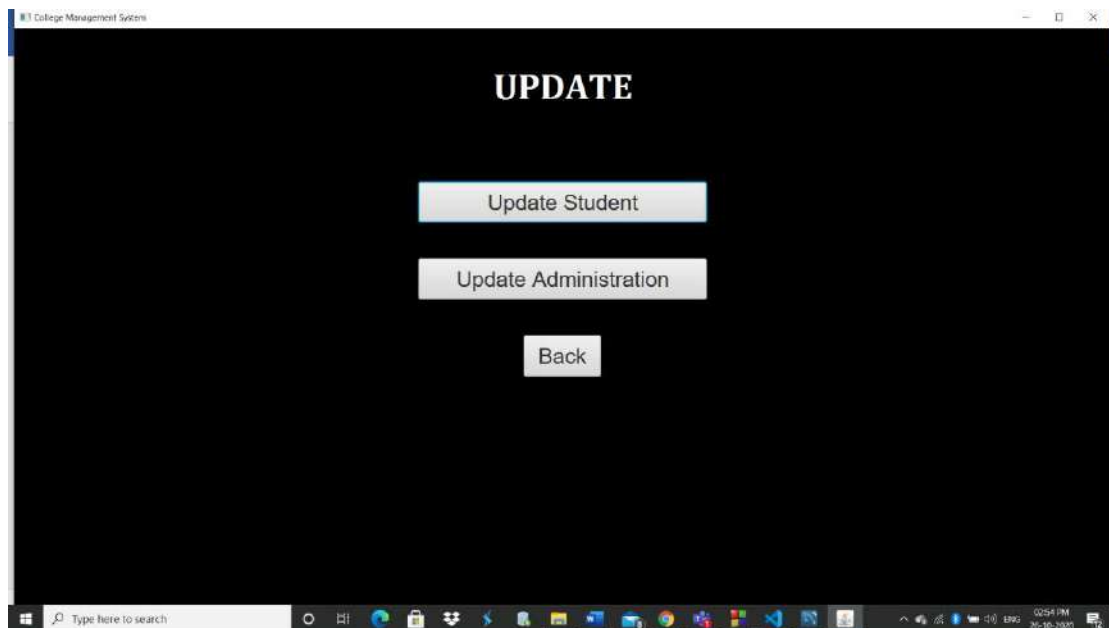
- Enter Registration number of student whose details is to be updated.
- Click on Search

- If we enter wrong Registration Number or Registration number which does not exist in database then pop up window will show message as shown below
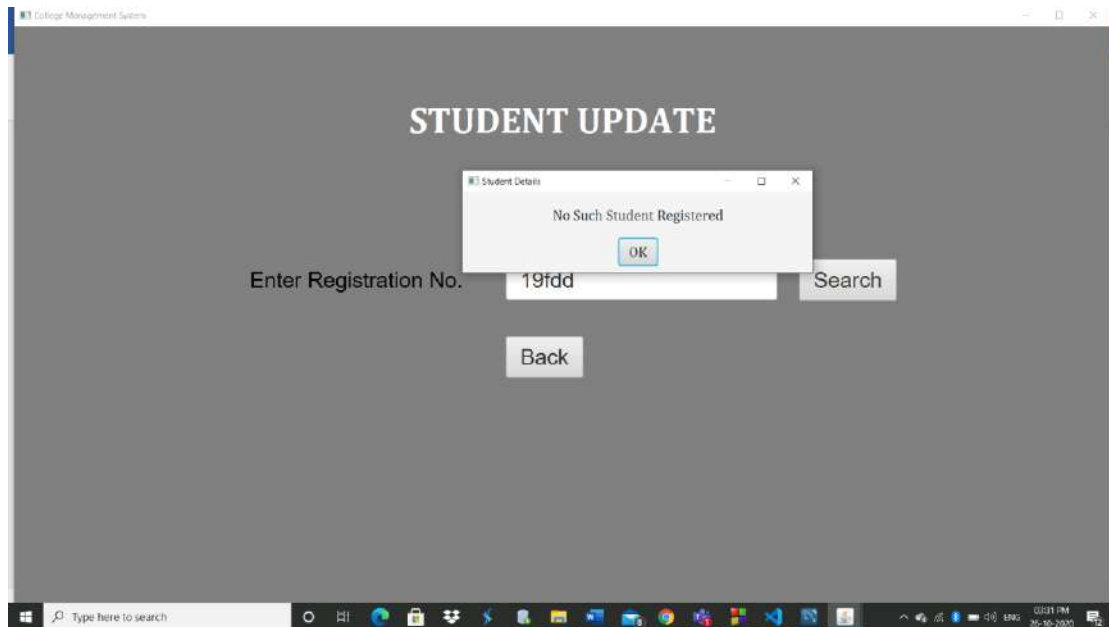
- If correct Registration Number

- The existing details of the searched student will automatically appear ready to be updated.

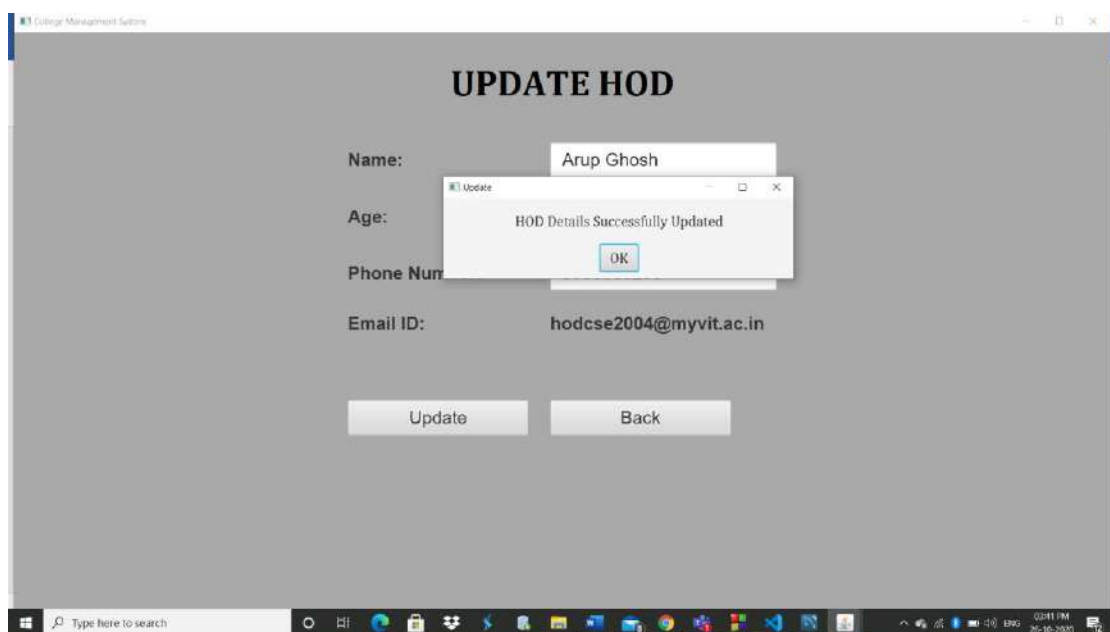- Change the details according to need then click on Update

- Pop up window will notify that the details has been updated as shown below

2. UPDATE ADMINISTRATION BOARD DETAILS

- Click on Update Administration

- Click on Update Administration Board

- Enter Designation of Board Member whose details is to be updated.
- Click on Search

- If we enter wrong designation or designation which does not exist in database then pop up window will show message as shown below

- If correct Designation

- The existing details of the searched Designation will automatically appear ready to be updated.

- Change the details according to need then click on Update

- Pop up window will notify that the details has been updated as shown below

3. UPDATE DEAN DETAILS

- Click on Update Administration
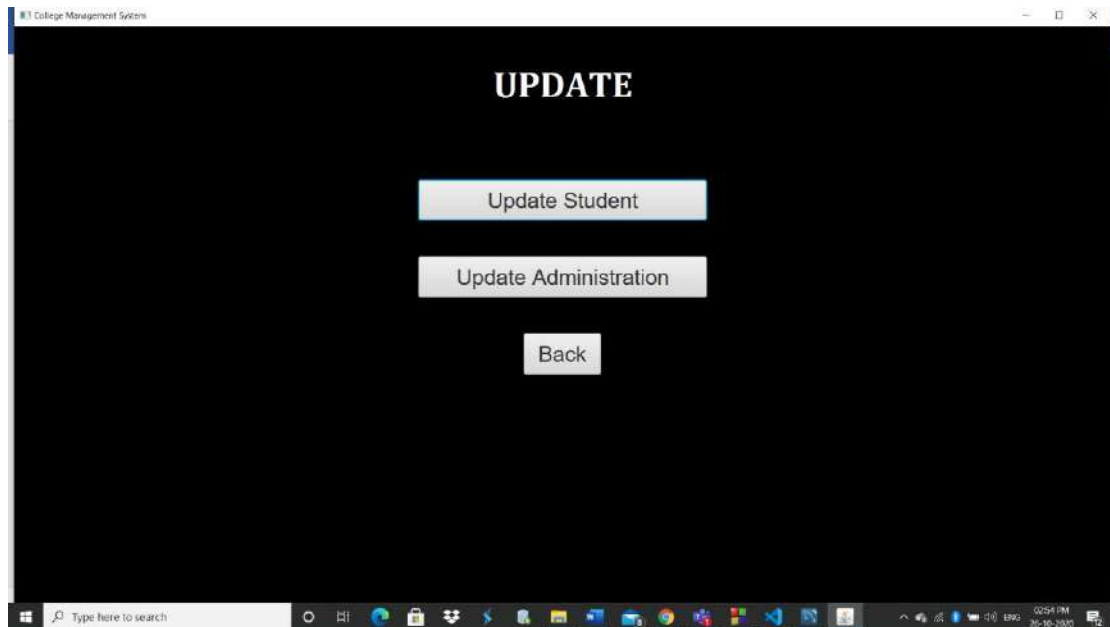
- Click on Update Dean
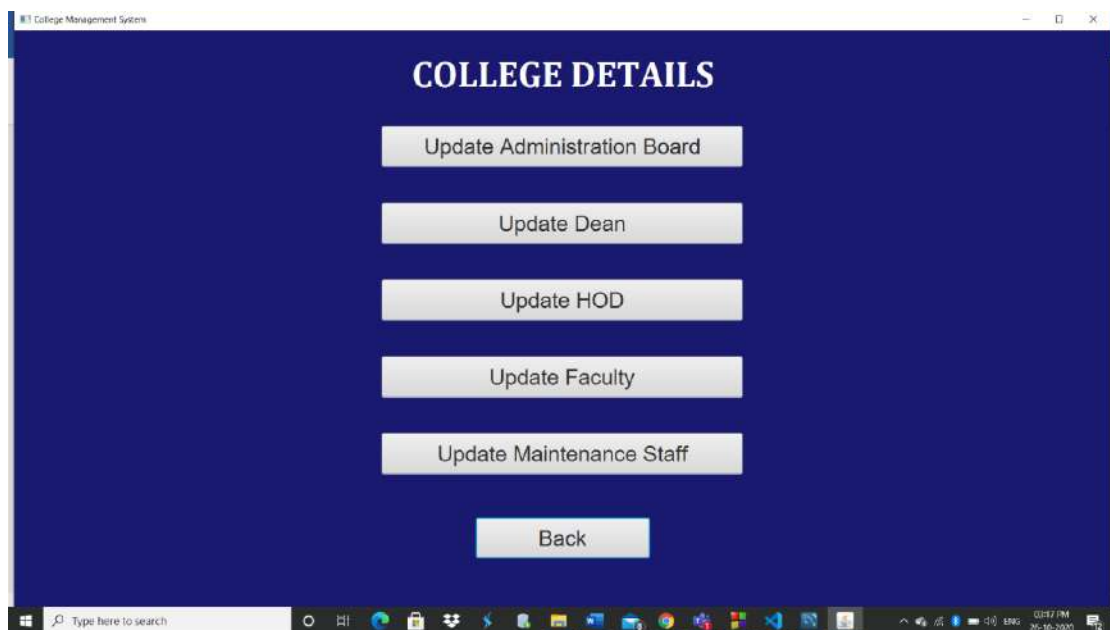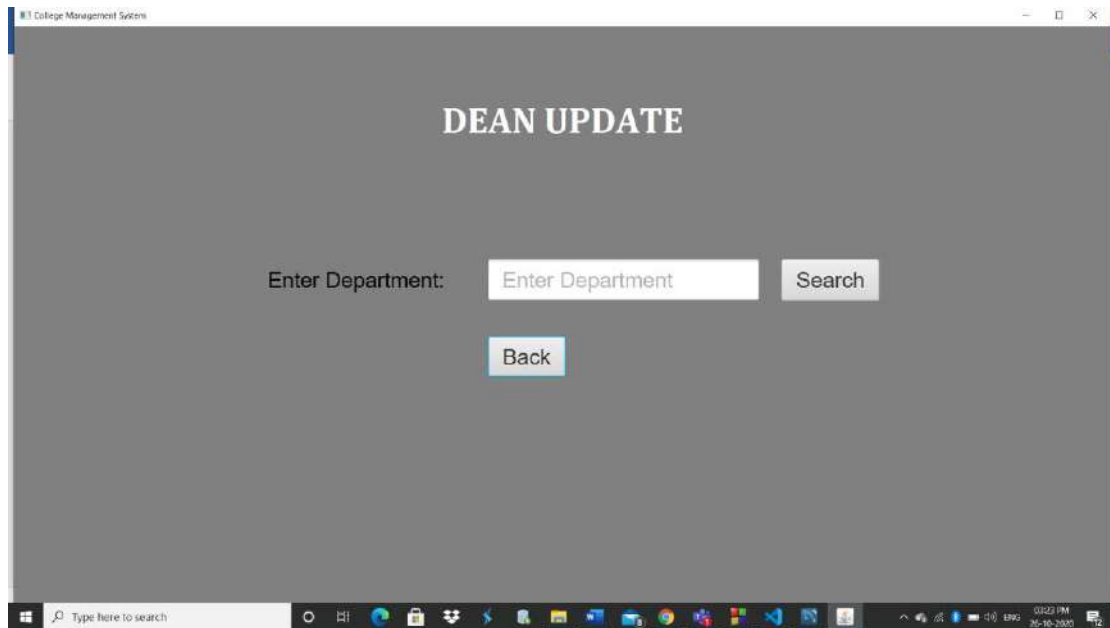
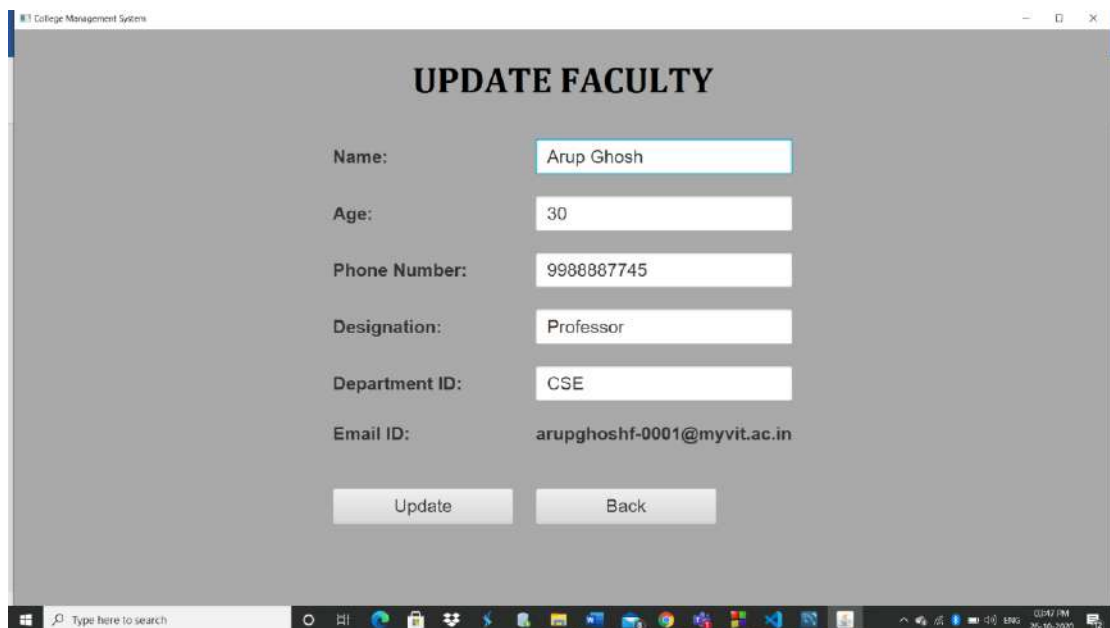- Enter Department of Dean whose details is to be updated.
- Click on Search
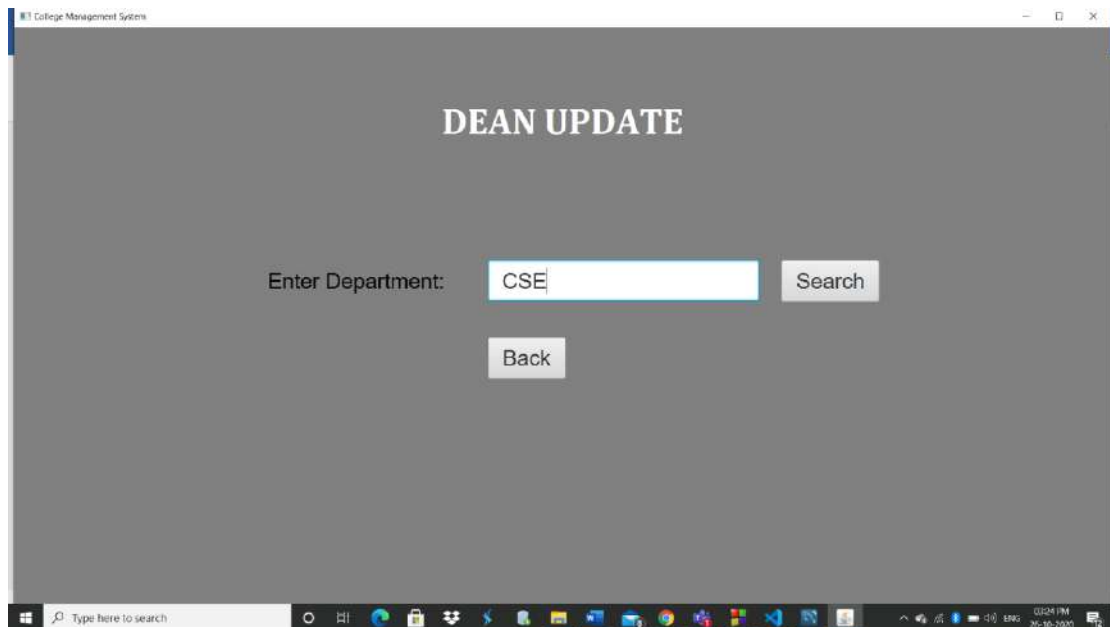
- If we enter wrong Department or Department which does not exist in database then pop up window will show message as shown below
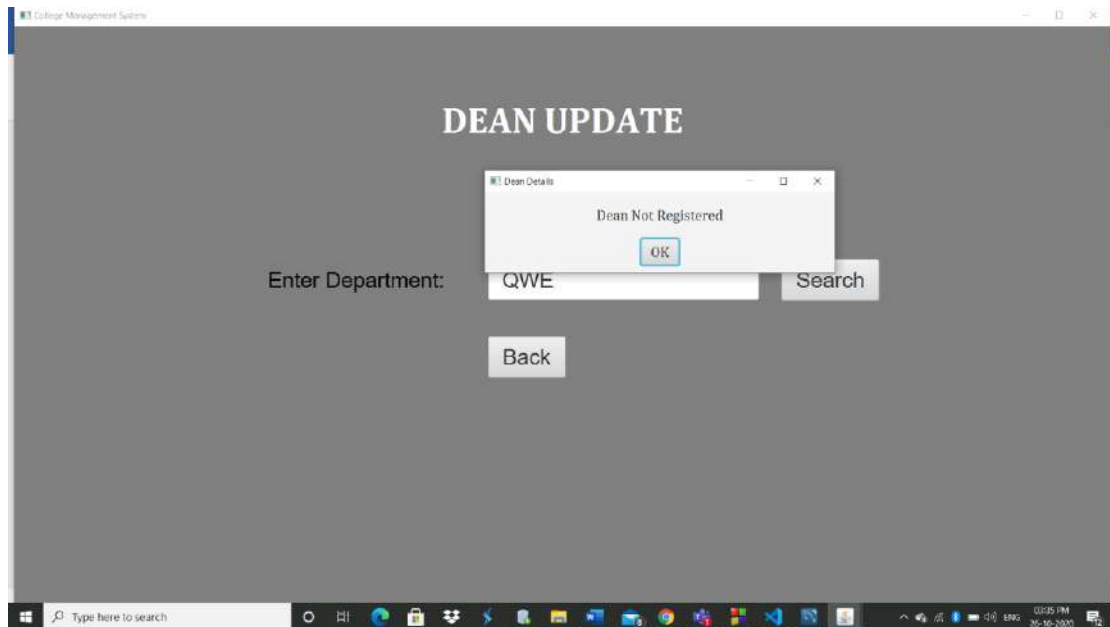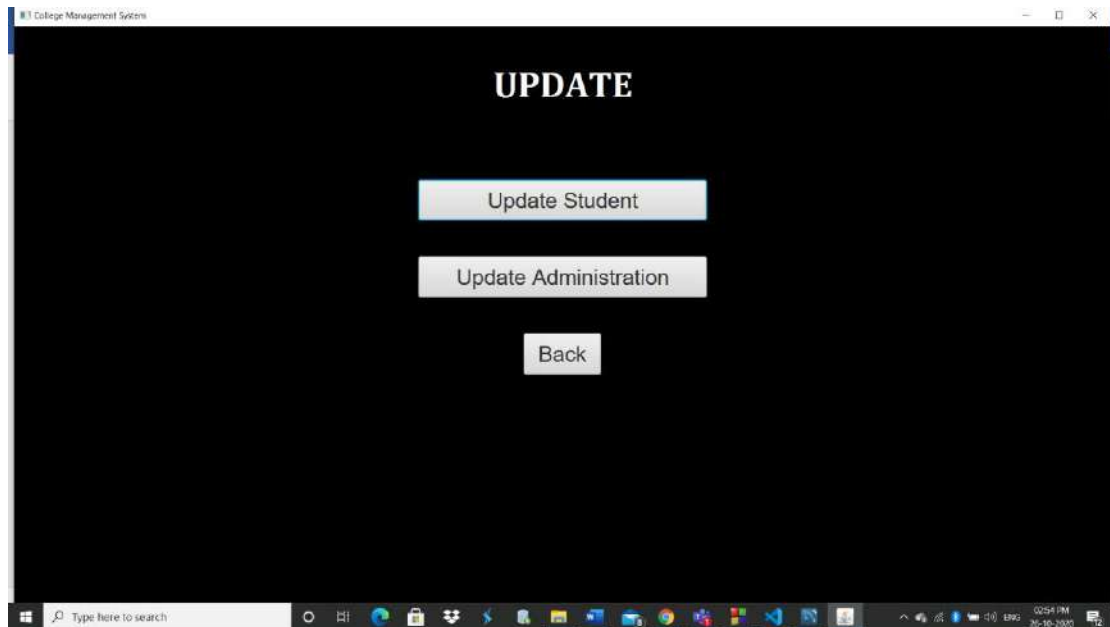
- If correct Department

- The existing details of the searched Dean will automatically appear ready to be updated.
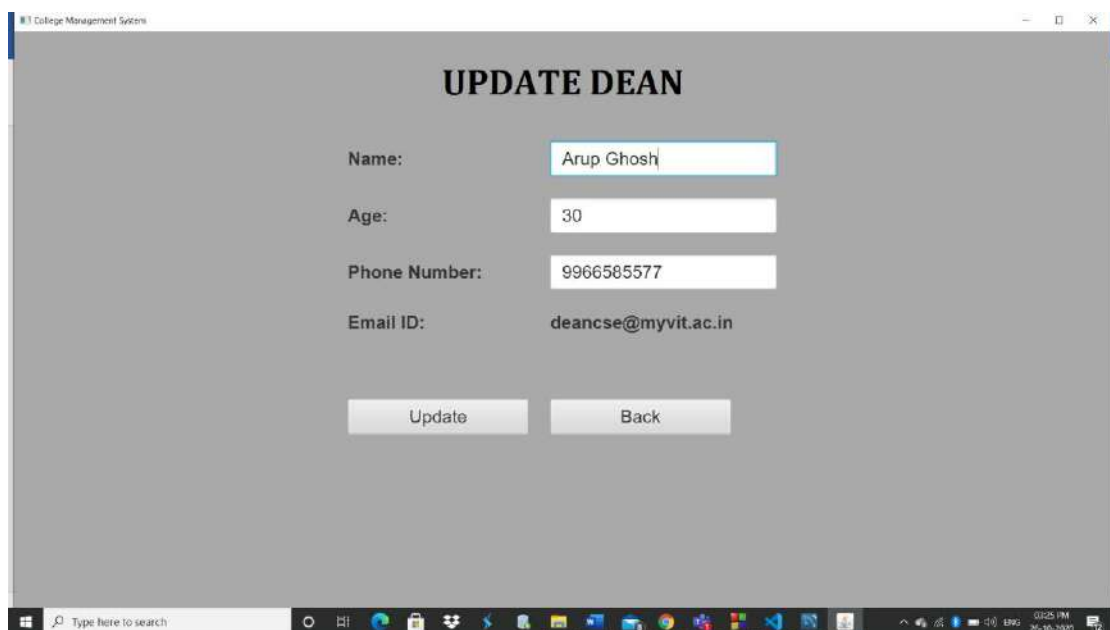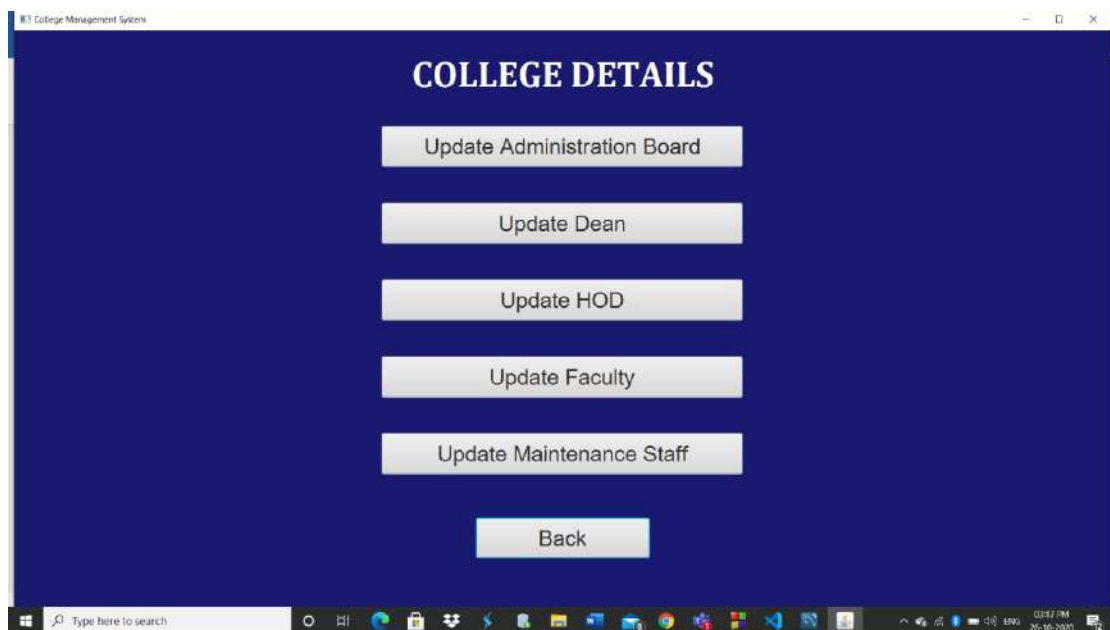
4. UPDATE HOD DETAILS

- Click on Update Administration

- Click on Update HOD
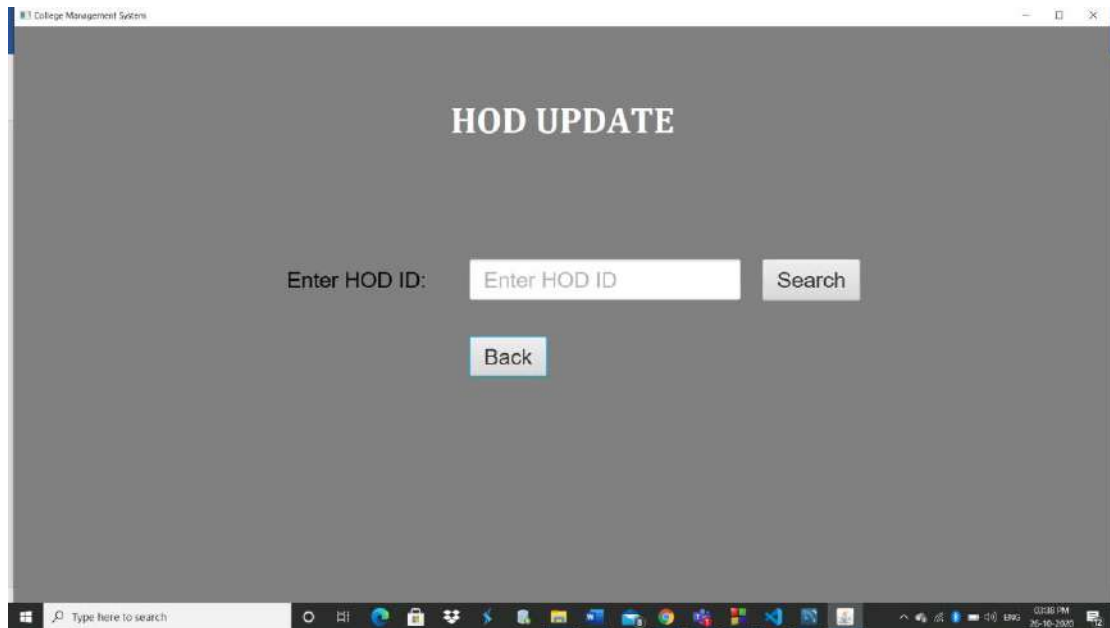
- Enter HOD ID whose details is to be updated.
- Click on Search
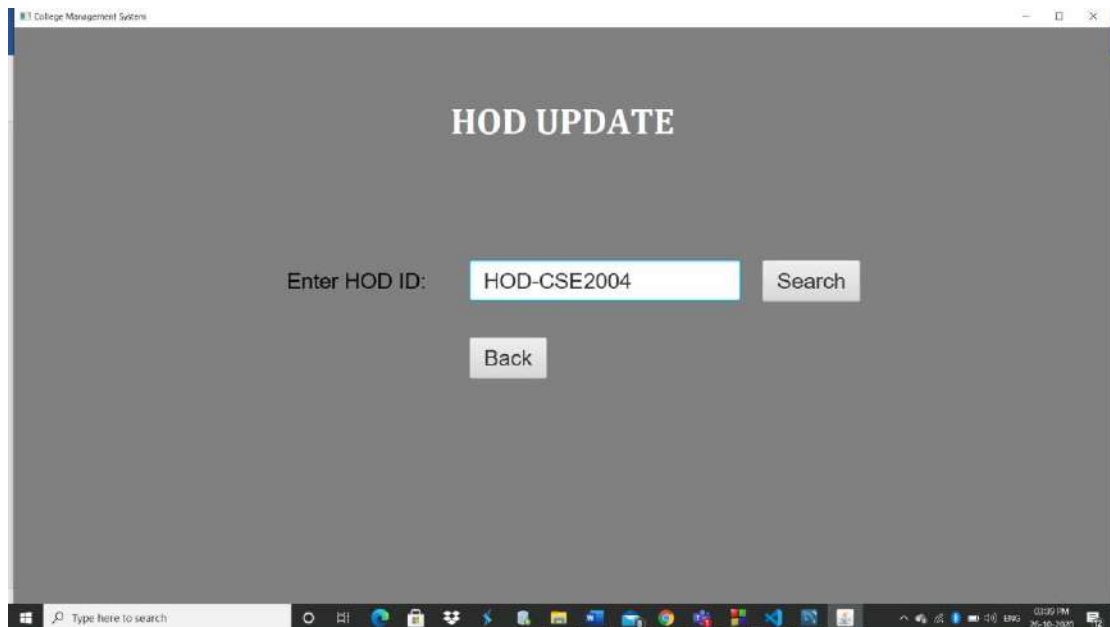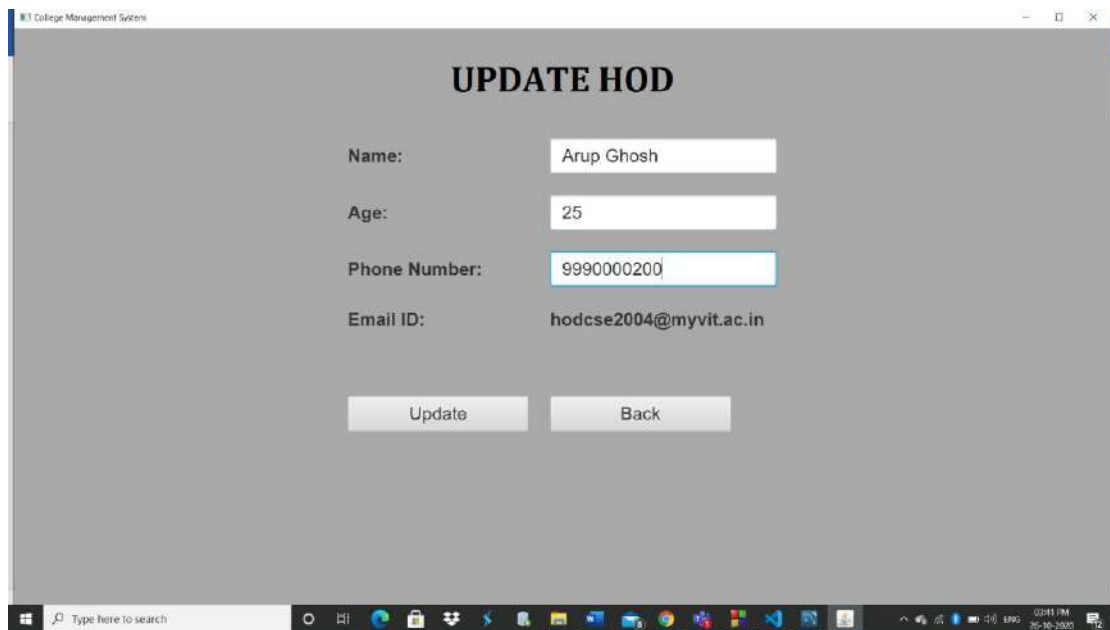
- If we enter wrong HOD ID or HOD ID which does not exist in database then pop up window will show message as shown below
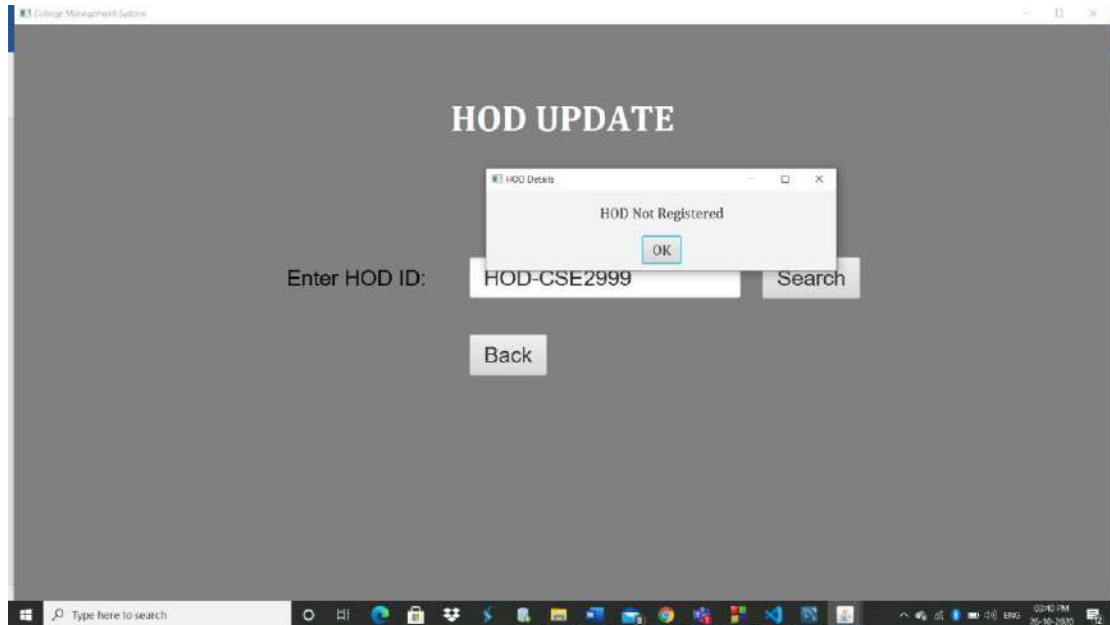
- If correct HOD ID

- The existing details of the searched HOD will automatically appear ready to be updated.

- Change the details according to need then click on Update

- Pop up window will notify that the details has been updated as shown below

5. UPDATE FACULTY DETAILS

- Click on Update Administration
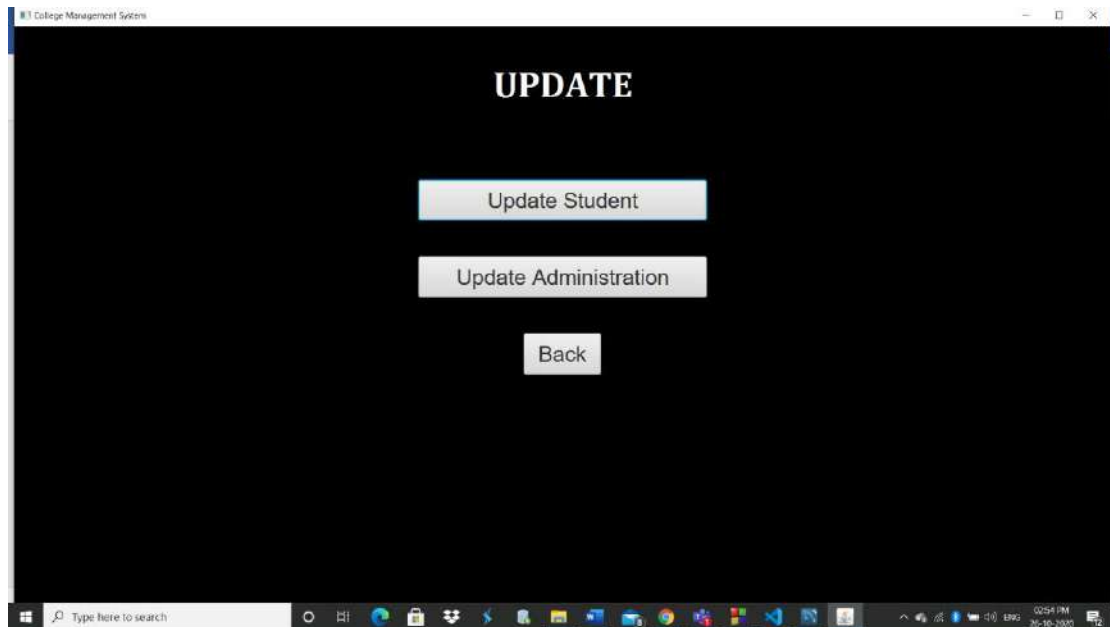
- Click on Update Faculty
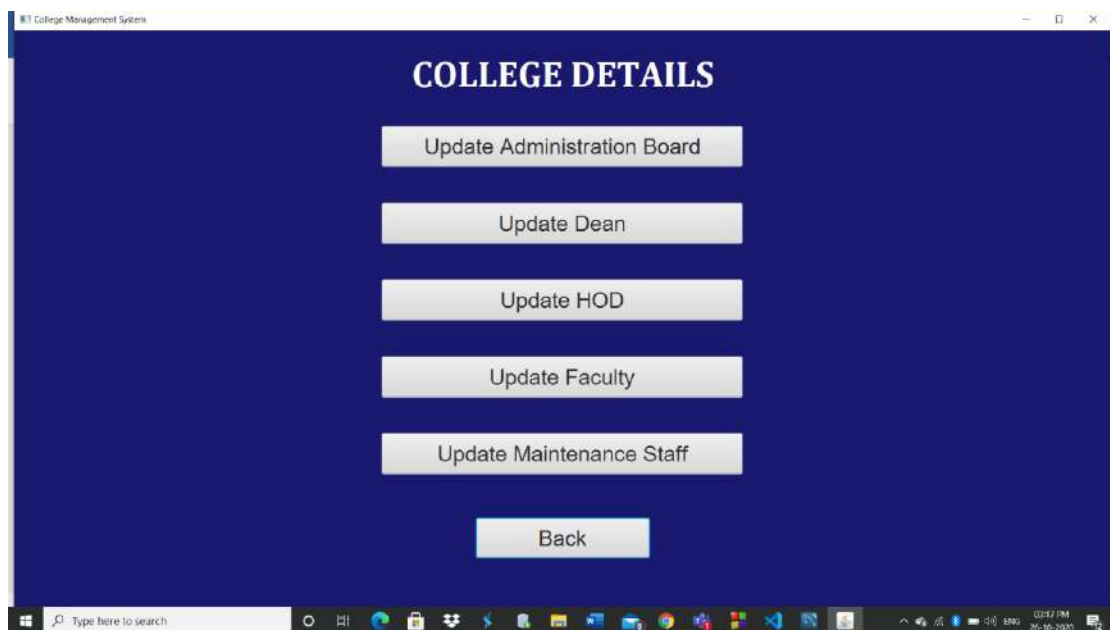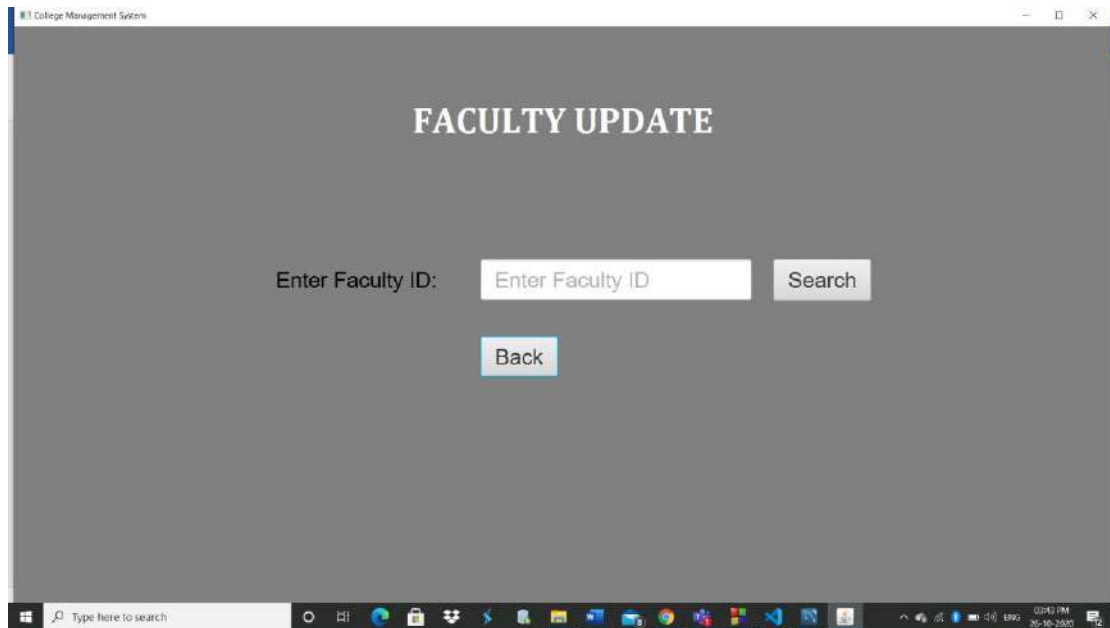
- Enter Faculty ID whose details is to be updated.
- Click on Search
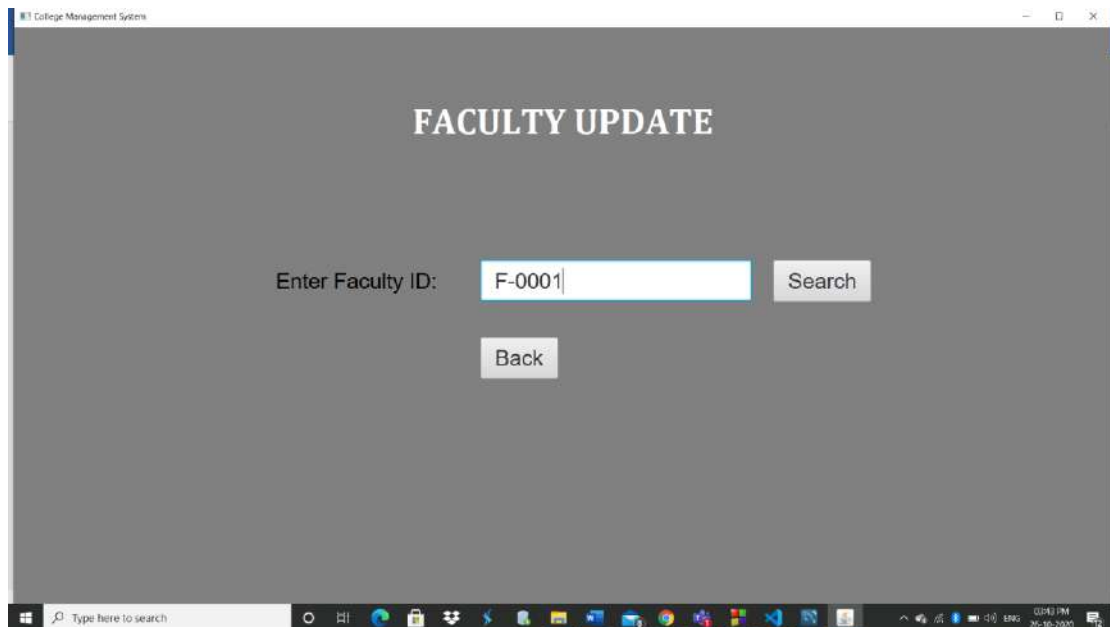
- If we enter wrong Faculty ID or Faculty ID which does not exist in database then pop up window will show message as shown below
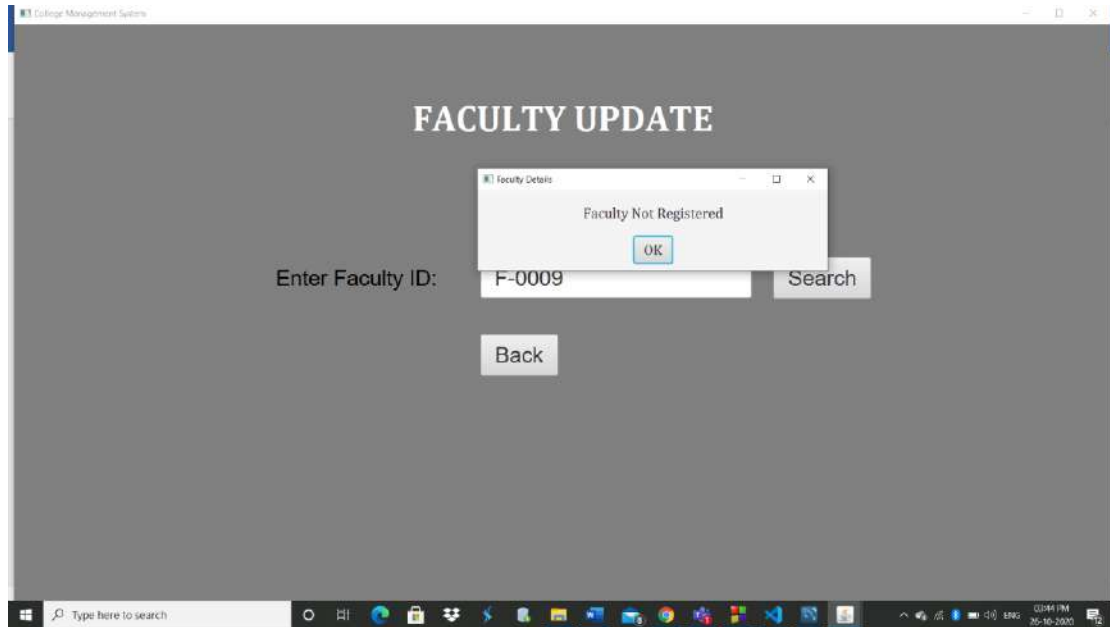
- If correct Faculty ID

- The existing details of the searched Faculty will automatically appear ready to be updated.

- Change the details according to need then click on Update

- Pop up window will notify that the details has been updated as shown below

6. UPDATE MAINTENANCE STAFF DETAILS

- Click on Update Administration

- Click on Update Maintenance Staff

- Enter Employee ID whose details is to be updated.
- Click on Search

- If we enter wrong Employee ID or Employee ID which does not exist in database then pop up window will show message as shown below
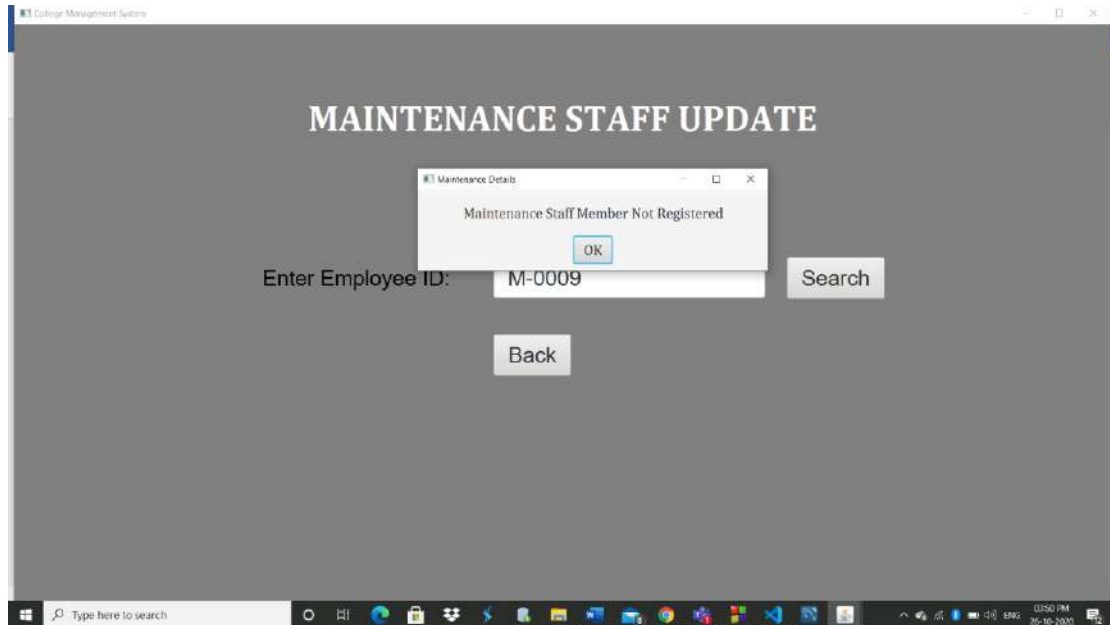
- If correct Employee ID

- The existing details of the searched Employee will automatically appear ready to be updated.

- Change the details according to need then click on Update

- Pop up window will notify that the details has been updated as shown below

DELETE DETAILS

- Click on Delete Details
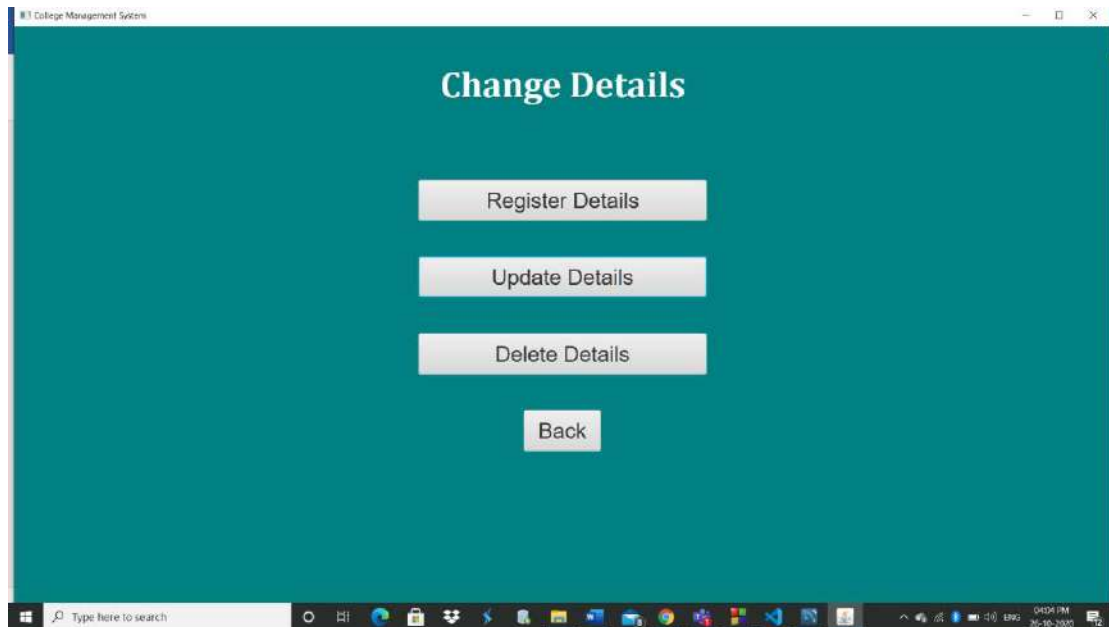
1. DELETE STUDENT

- Click on Delete Student
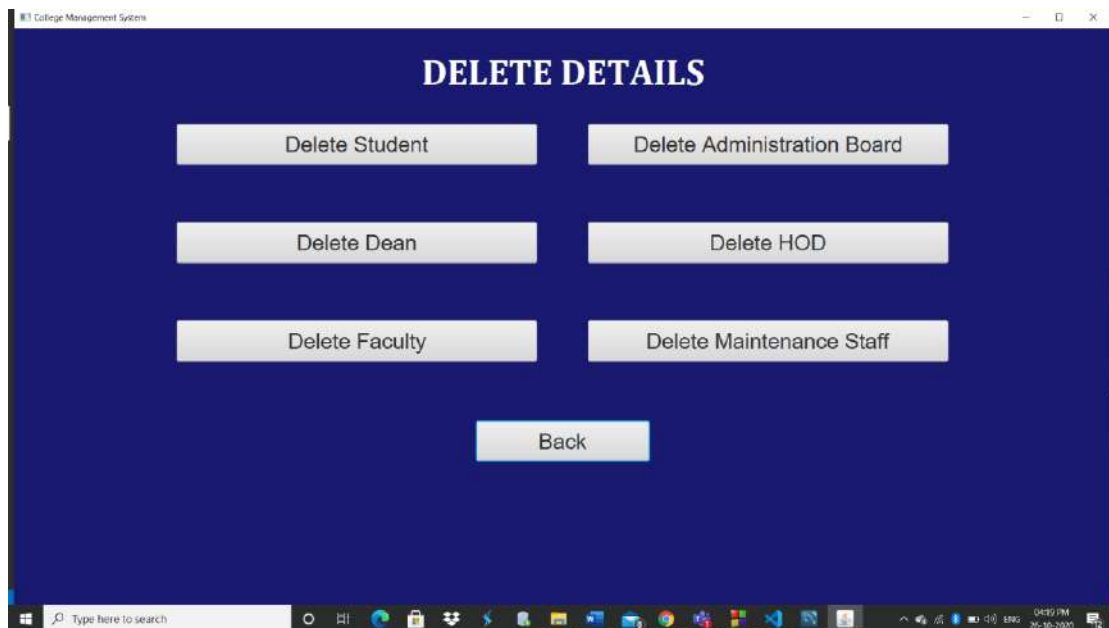
- Enter Registration Number of student whose details is to deleted.

- Click on delete button to delete student from data base.
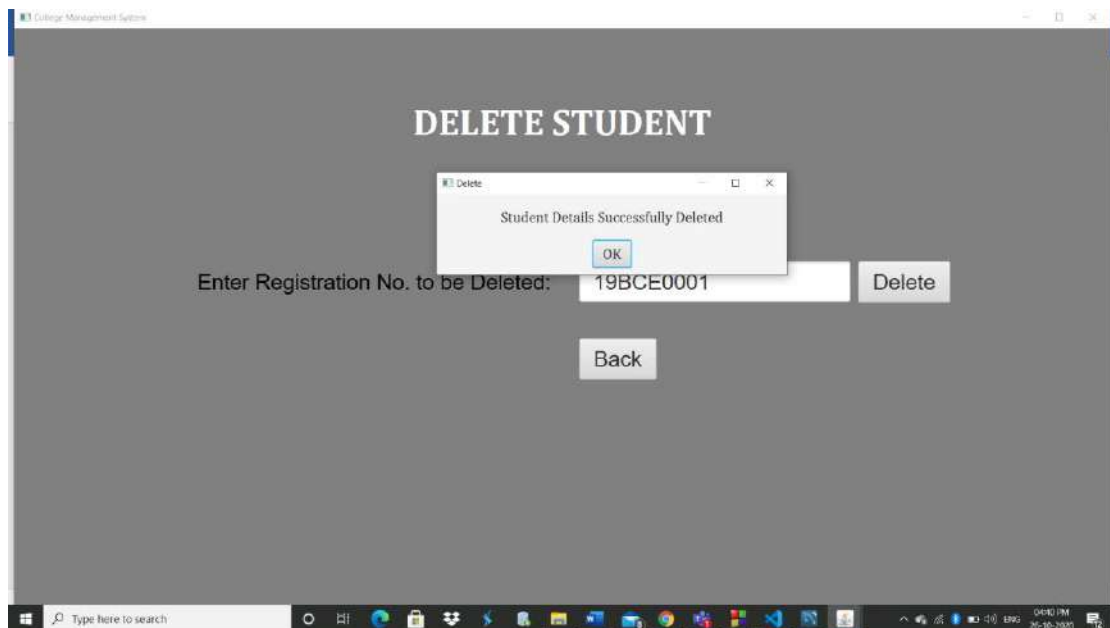
- If we enter wrong Registration Number or Registration Number which does not exist in database then pop up window will show message as shown below
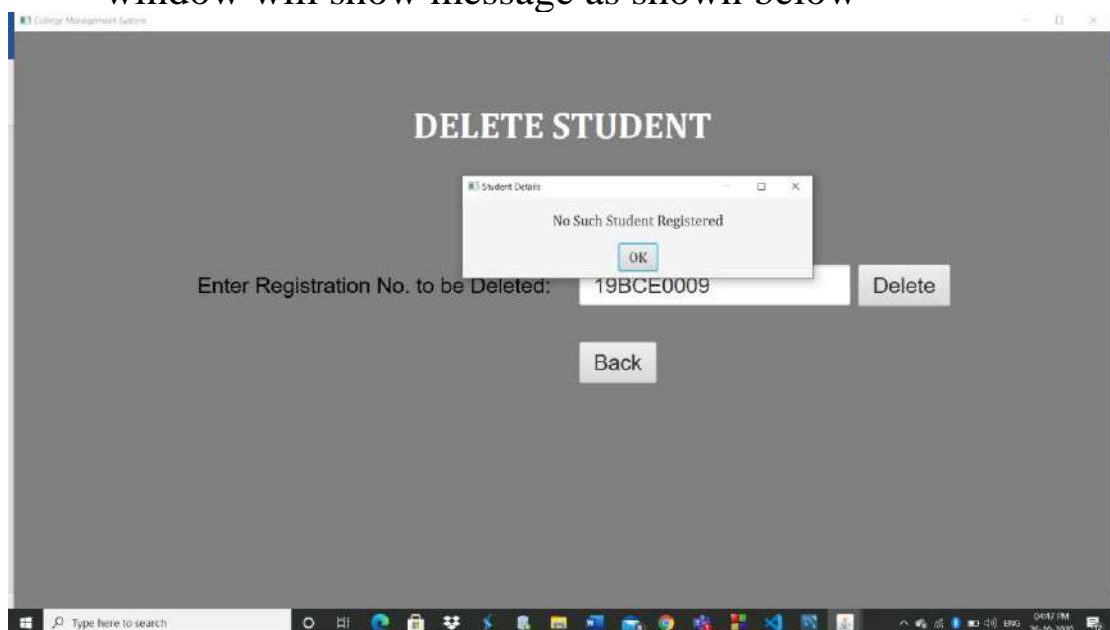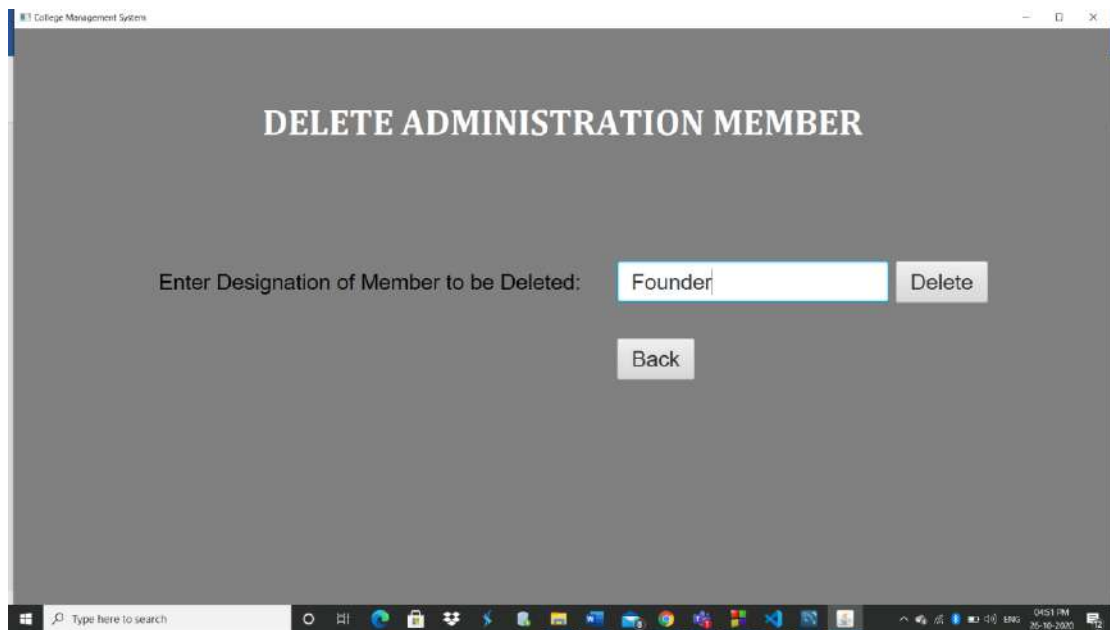
2. DELETE ADMINISTRATION BOARD

- Click on Delete Administration Board
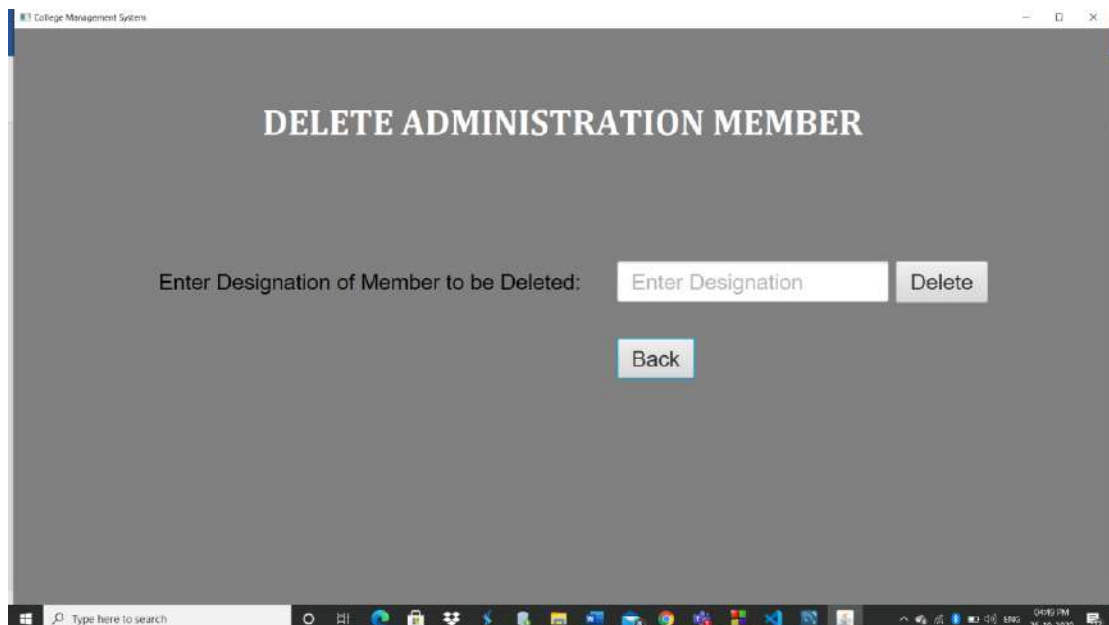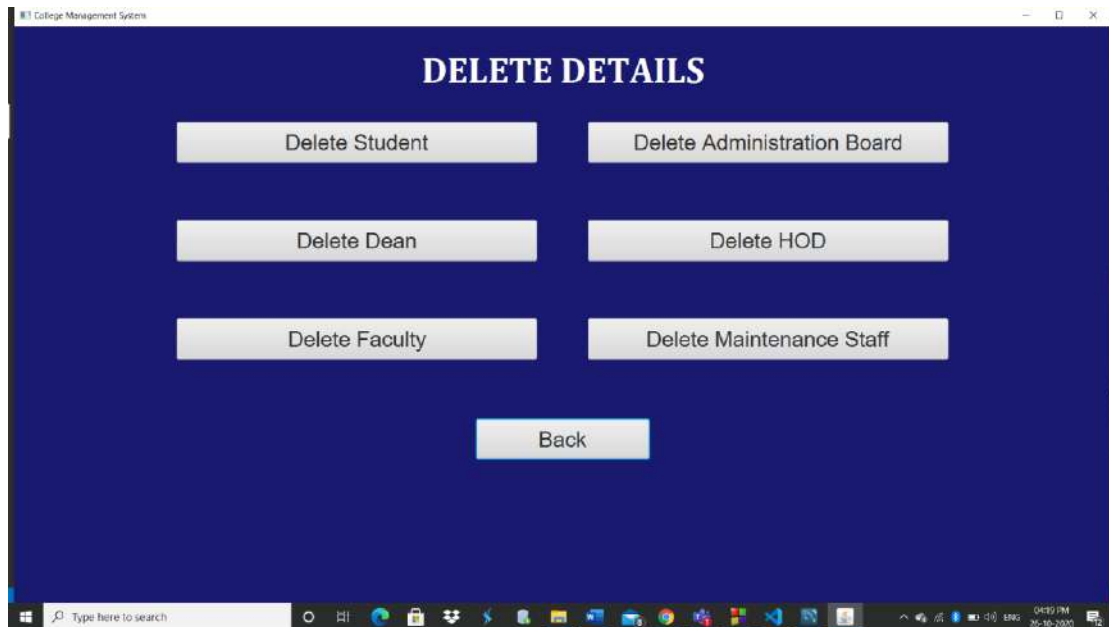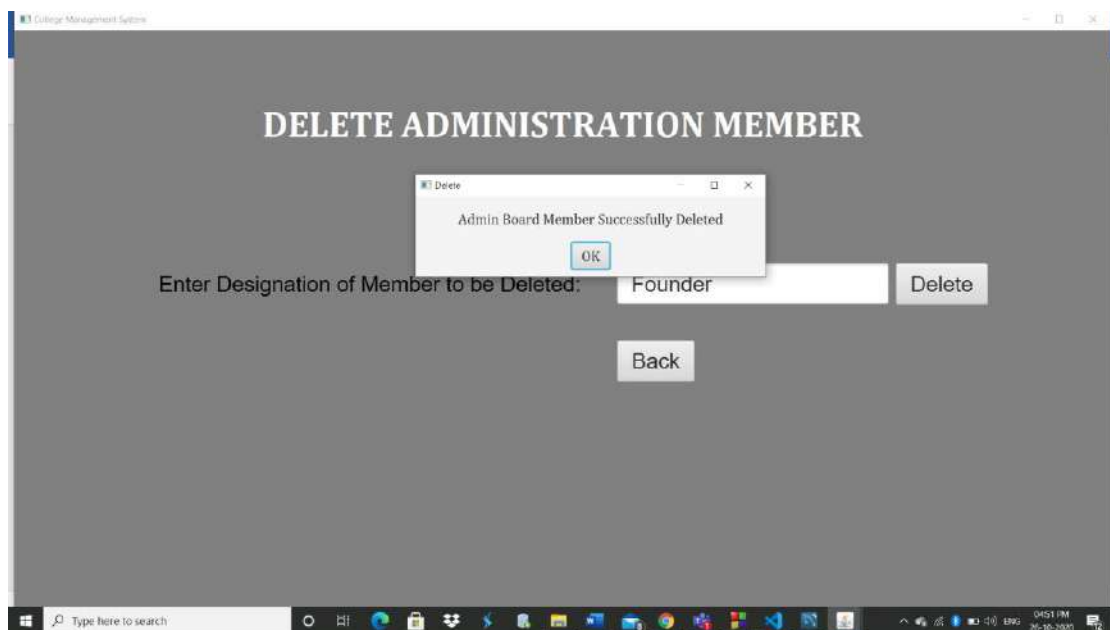
- Enter Designation of Administration Board Member whose details is to deleted.

- Click on delete button to delete Admin Member from data base.

- If we enter wrong Designation or Designation which does not exist in database then pop up window will show message as shown below
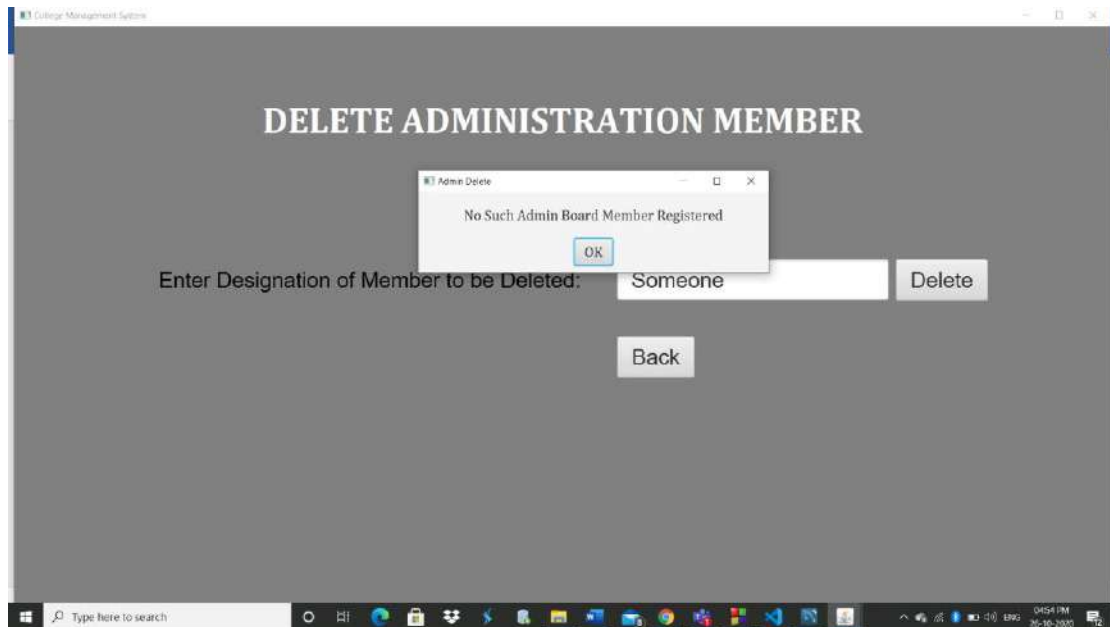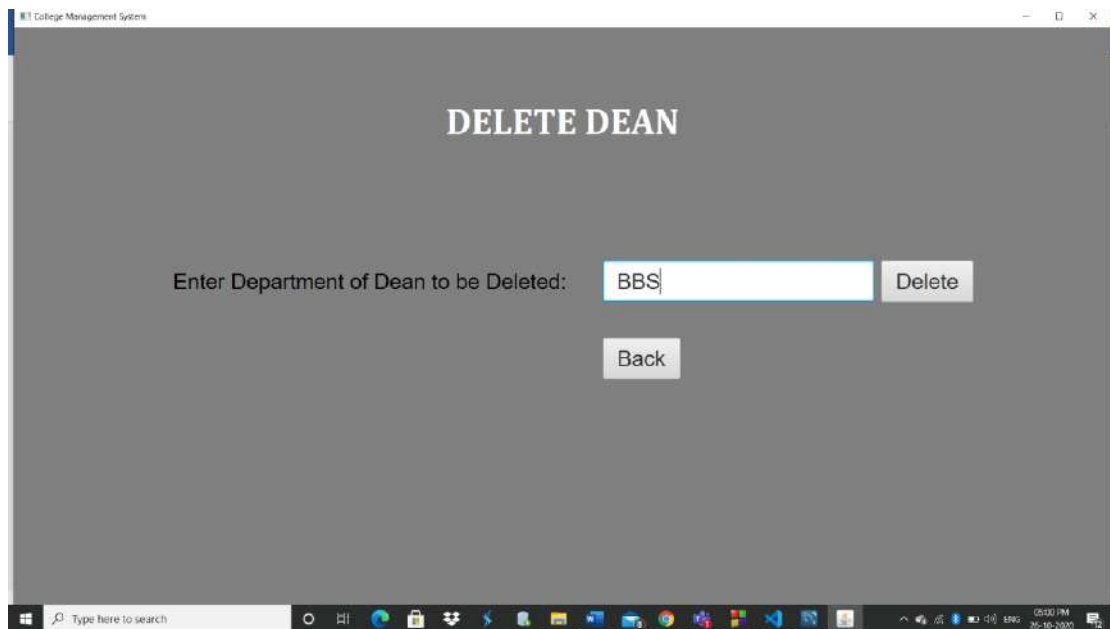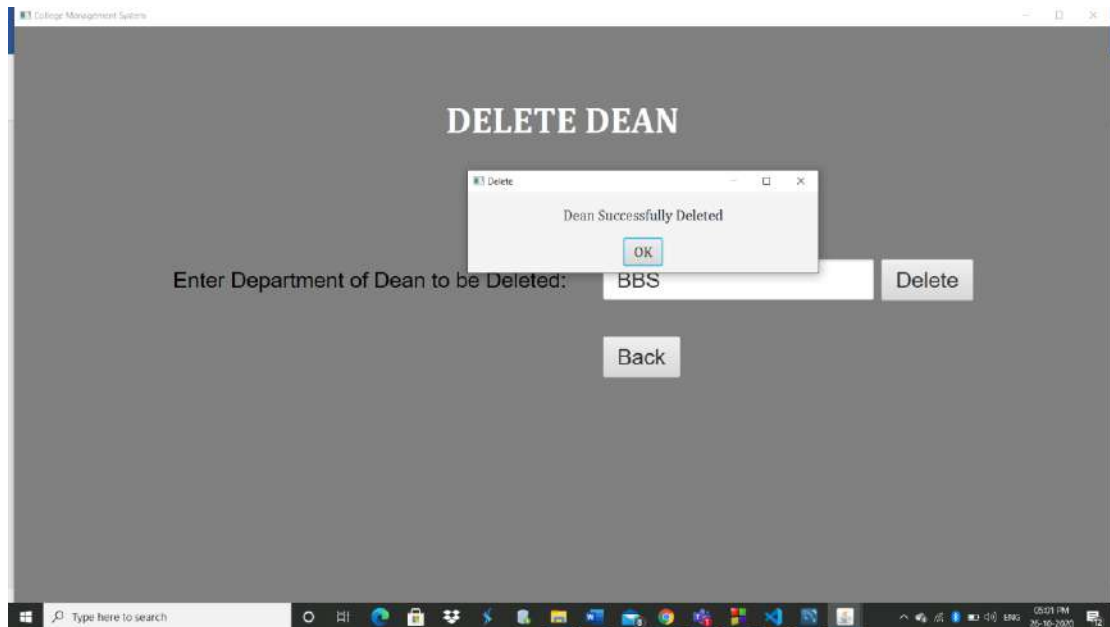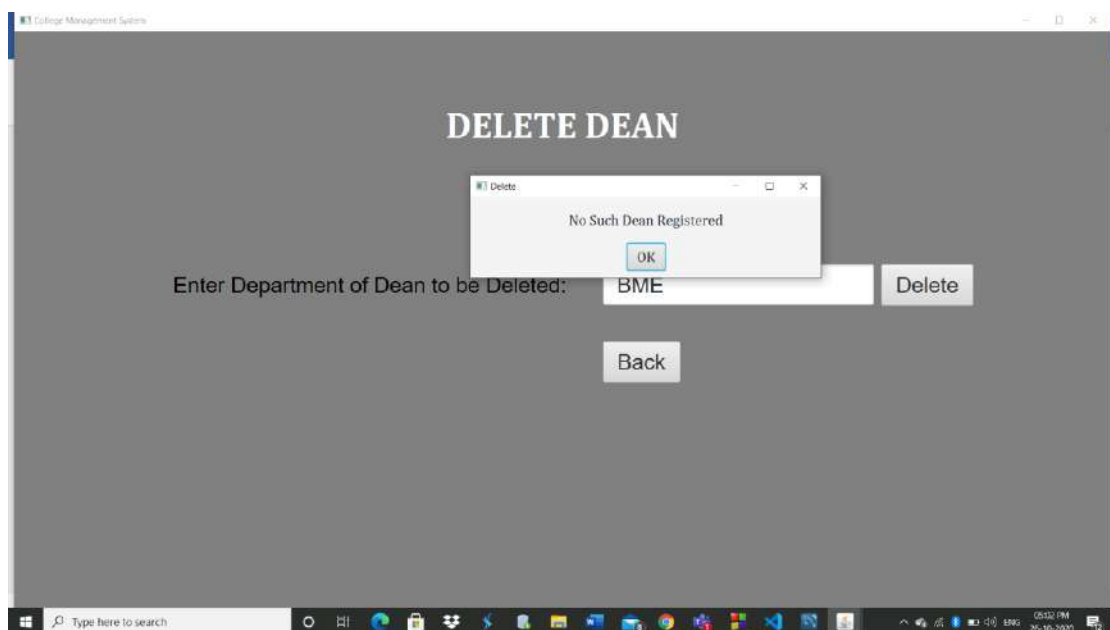

3. DELETE DEAN

- Click on Delete Dean
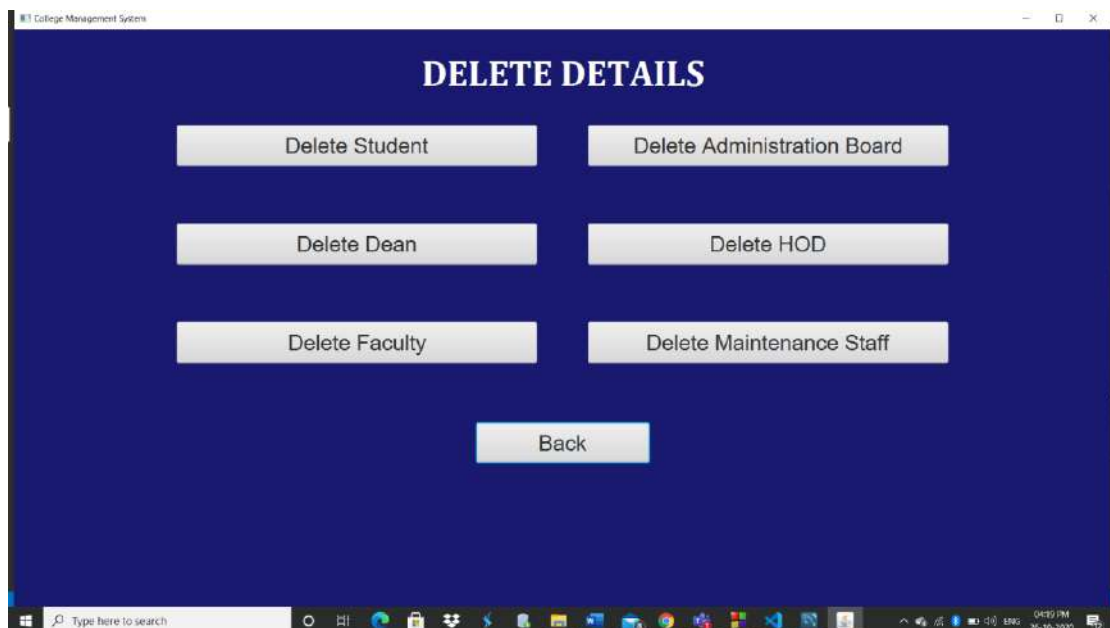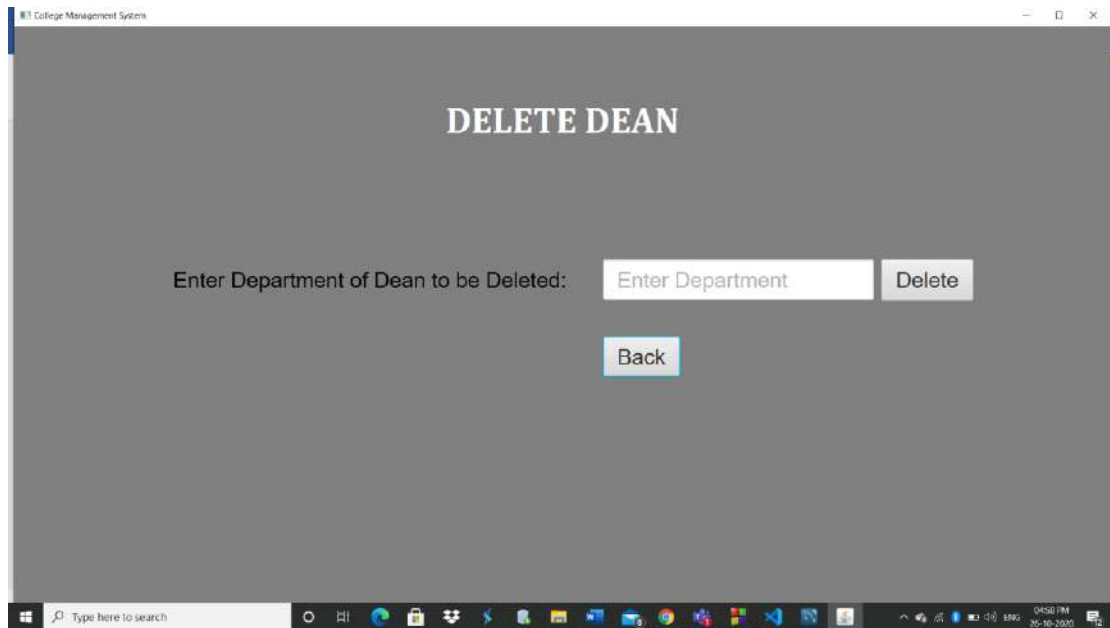
- Enter Department of Dean whose details is to deleted.
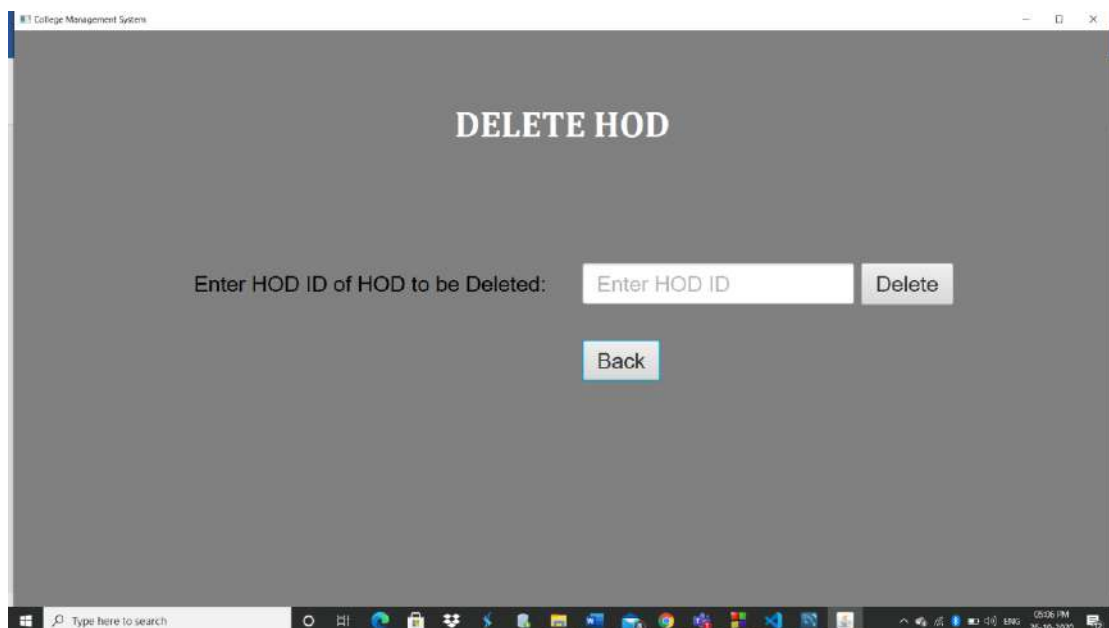
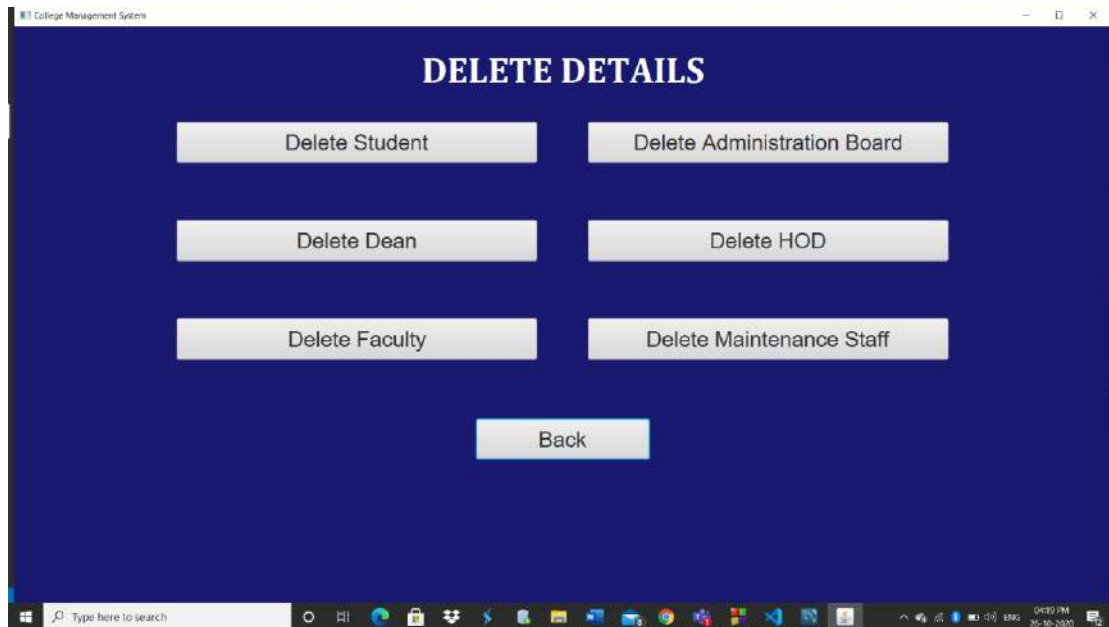- Click on delete button to delete Dean from data base.

- If we enter wrong Department or Department which does not exist in database then pop up window will show message as shown below
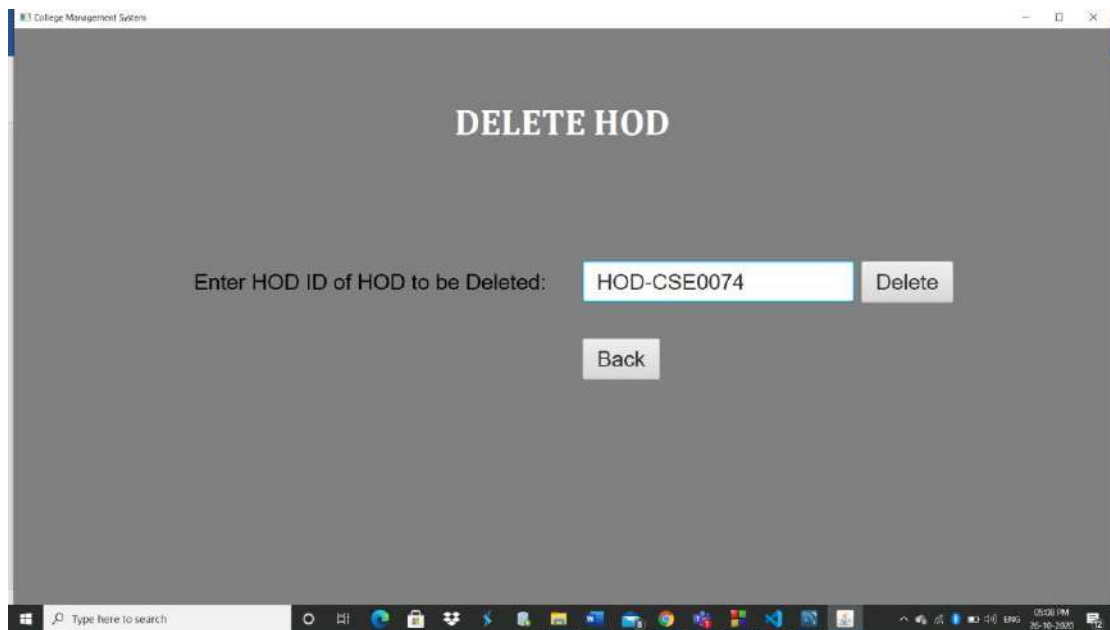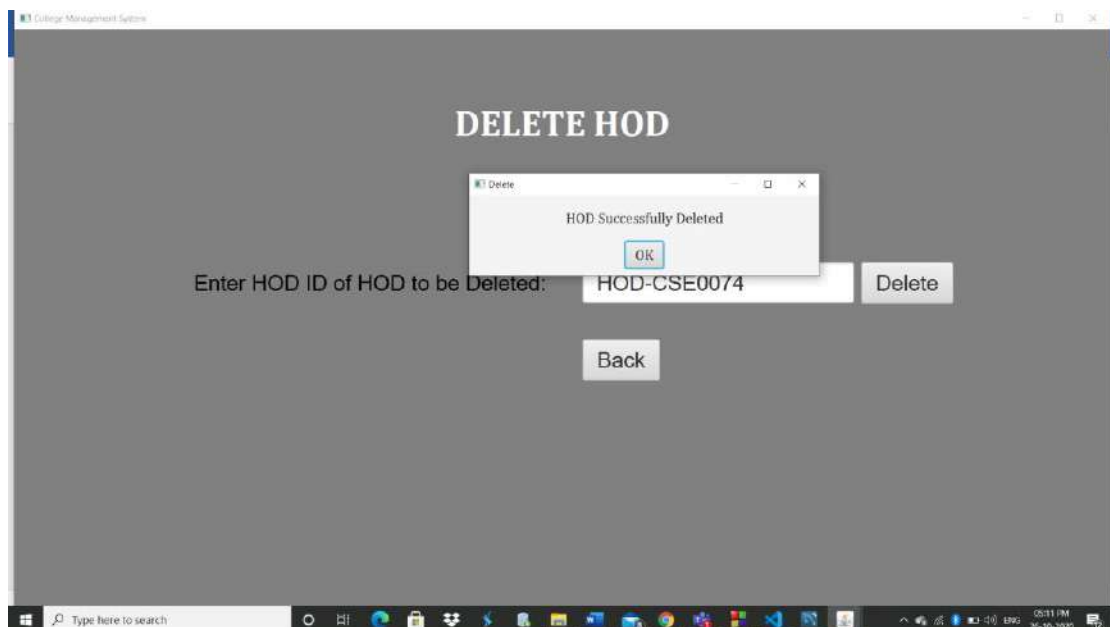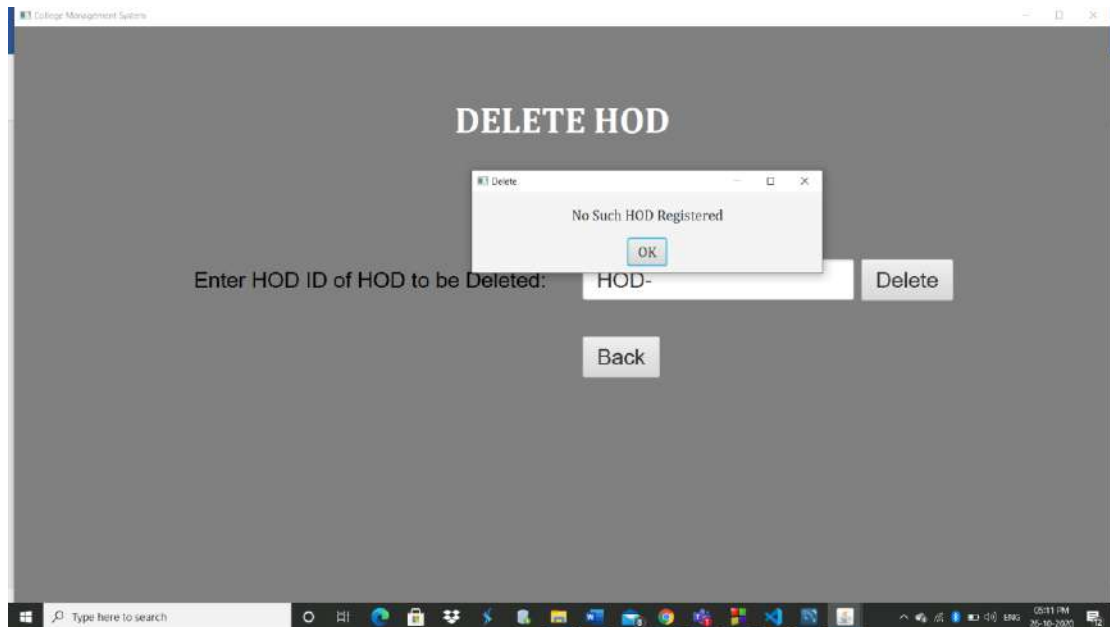
4. DELETE HOD

- Click on Delete HOD

- Enter HOD ID of HOD whose details is to deleted.
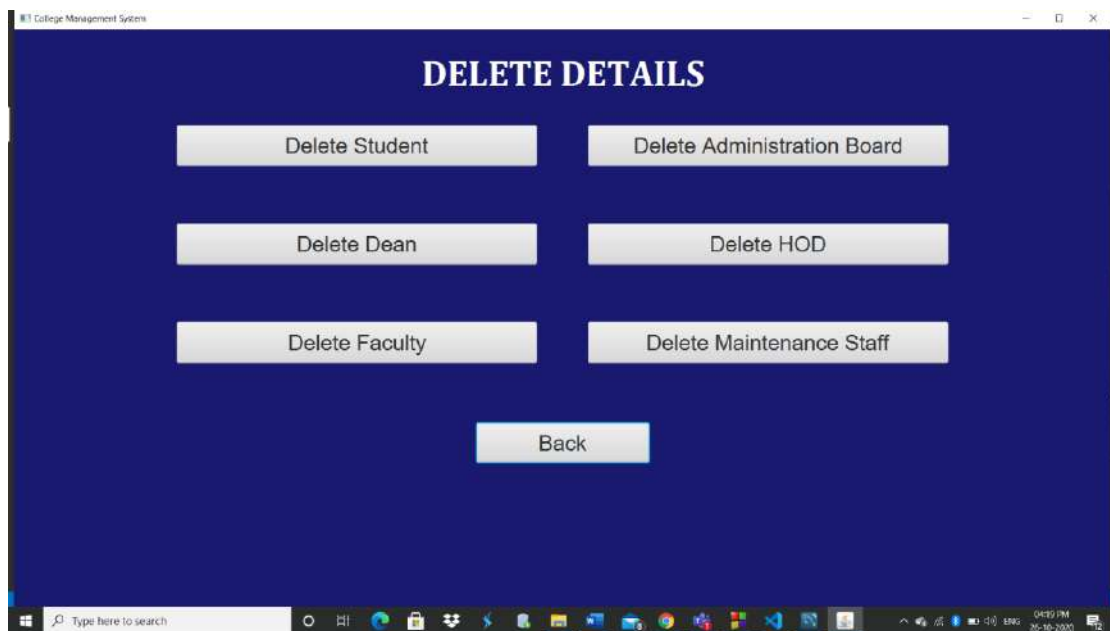

- Click on delete button to delete HOD from data base.

- If we enter wrong Department or Department which does not exist in database then pop up window will show message as shown below

5. DELETE FACULTY

- Click on Delete Faculty

- Enter Faculty ID of Faculty whose details is to deleted.
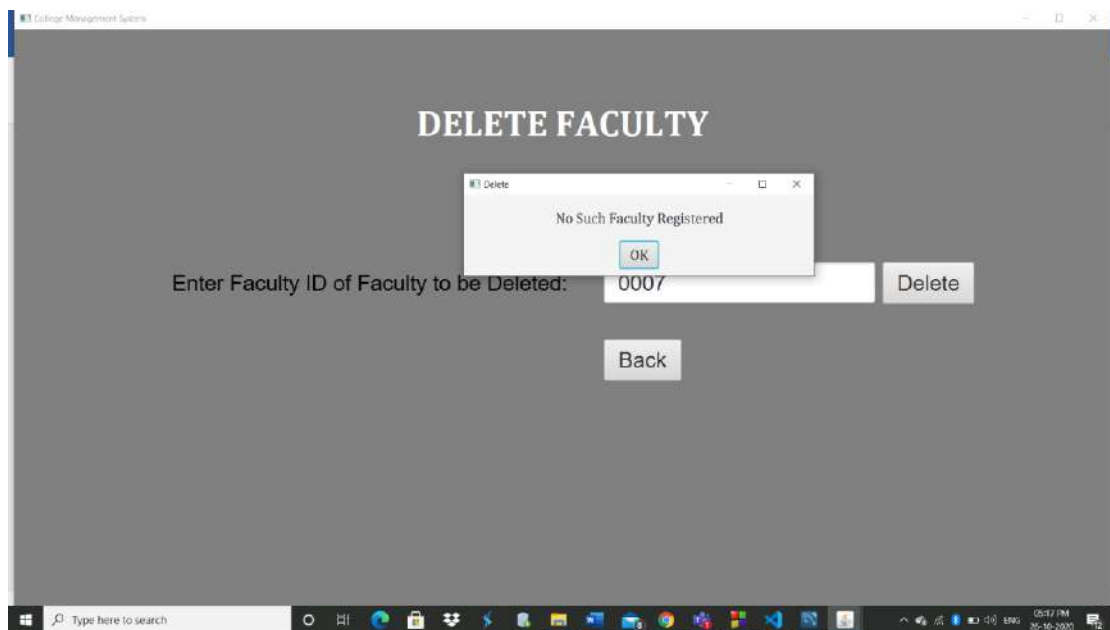
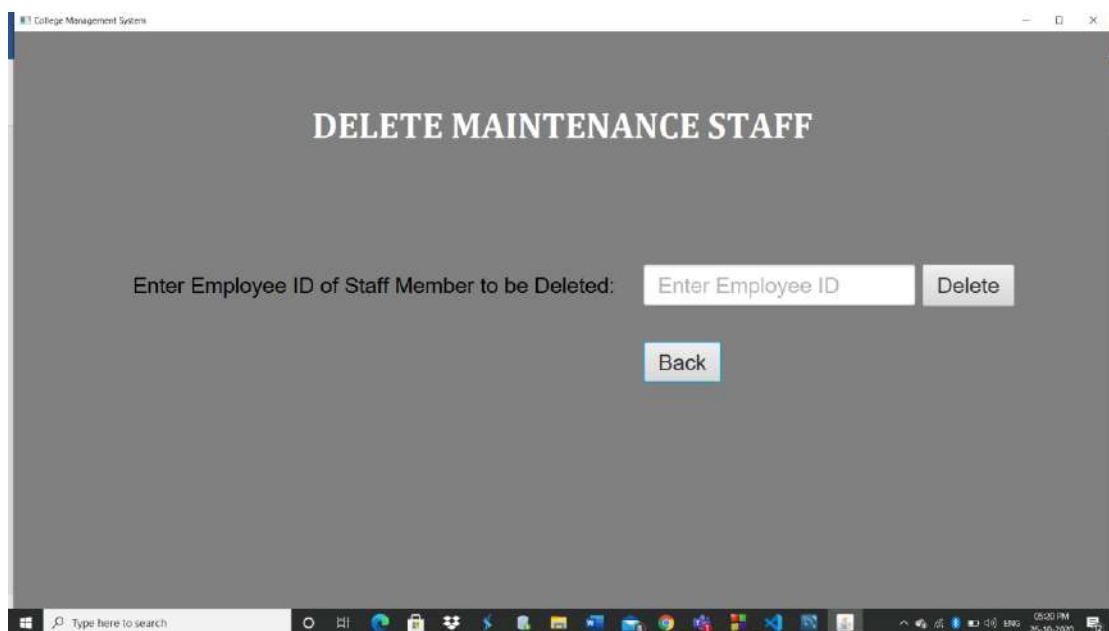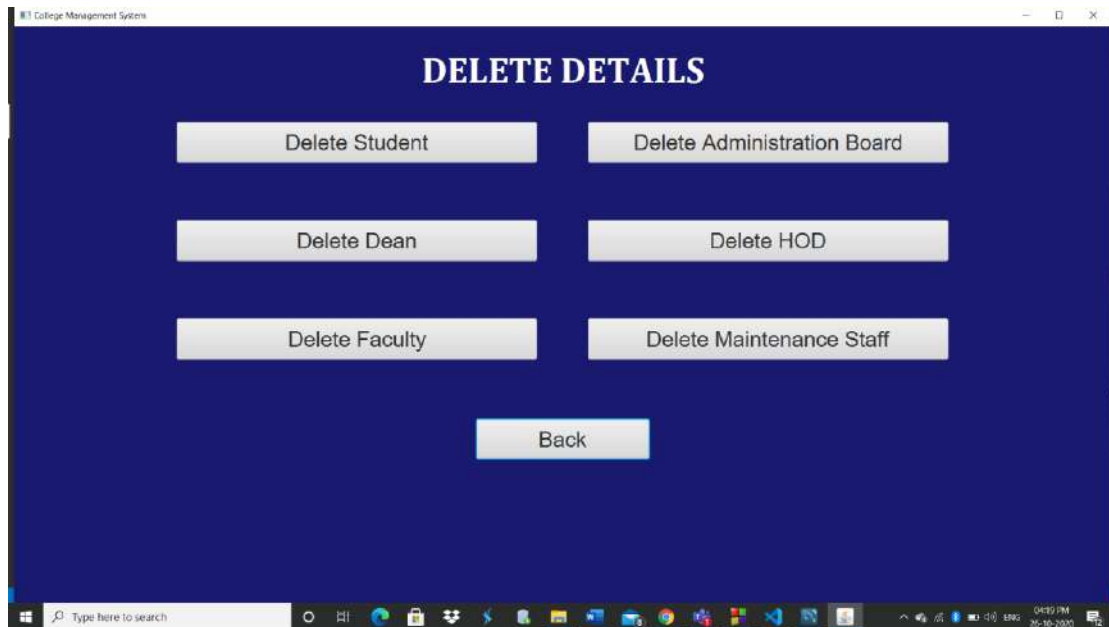- Click on delete button to delete Faculty from data base.
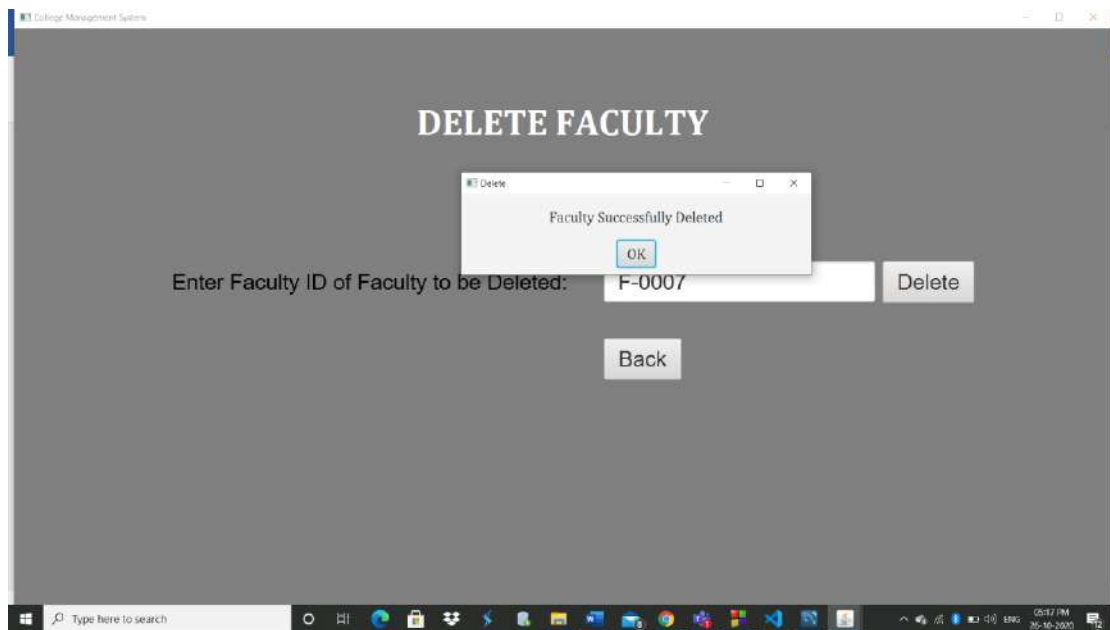
- If we enter wrong Faculty ID or Faculty ID which does not exist in database then pop up window will show message as shown below
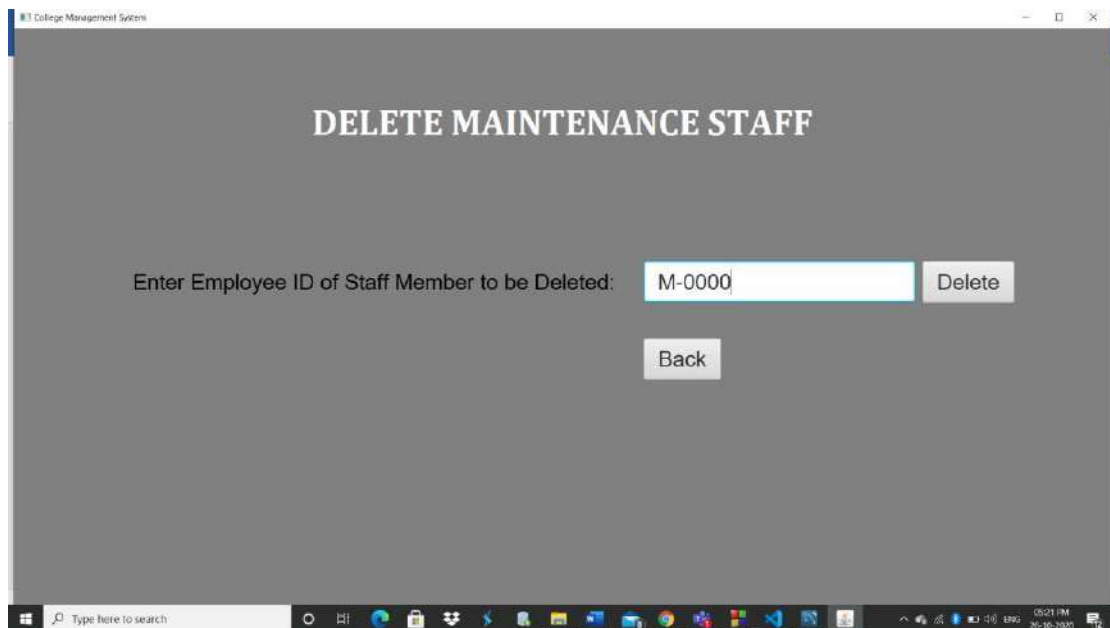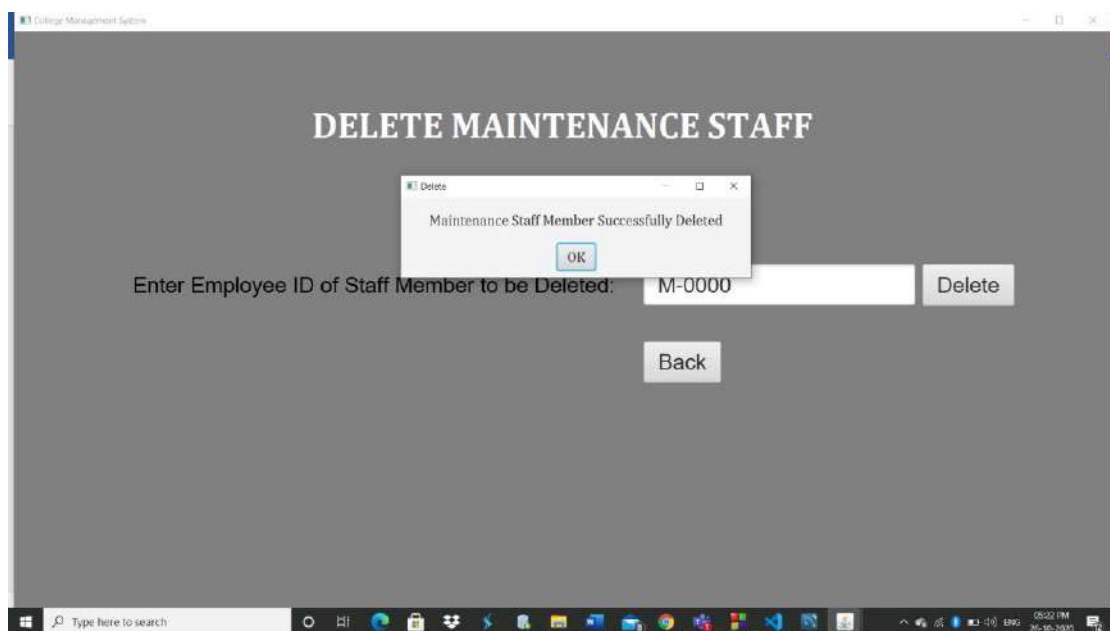
6. DELETE MAINTENANCE STAFF

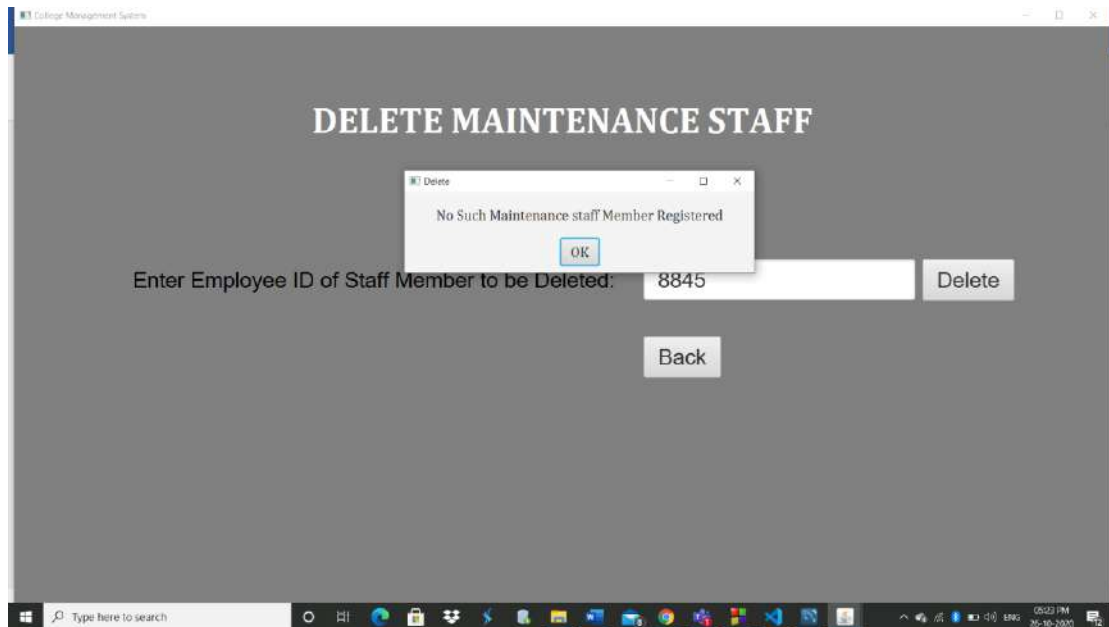- Click on Delete Maintenance Staff

- Enter Employee ID of Staff Member whose details is to deleted.

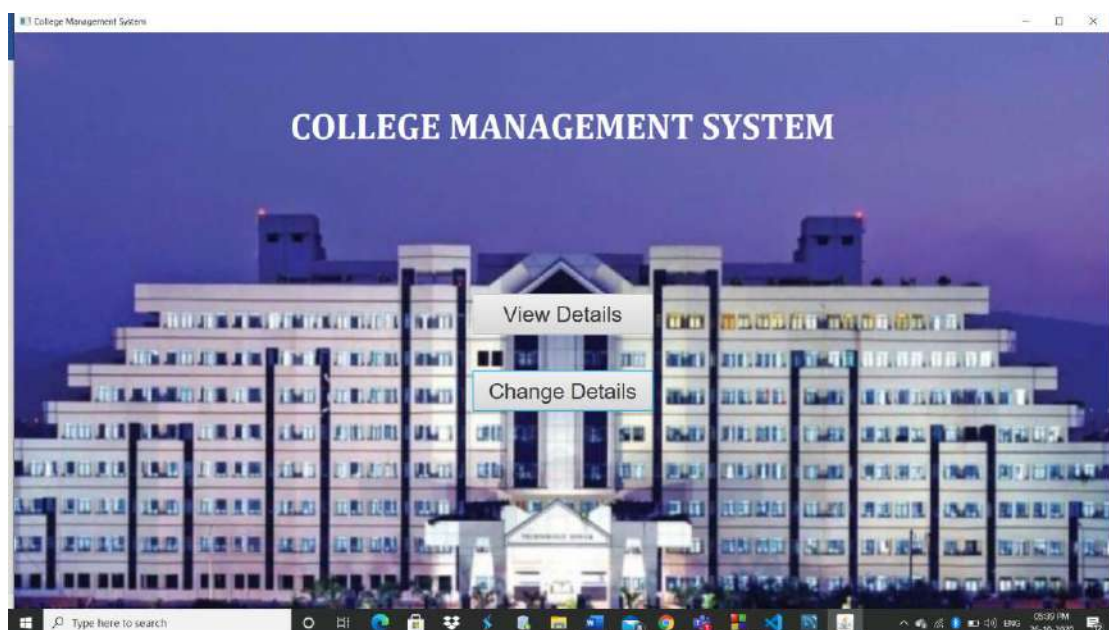- Click on delete button to delete Staff Member from data base.

- If we enter wrong Faculty ID or Faculty ID which does not exist in database then pop up window will show message as shown below
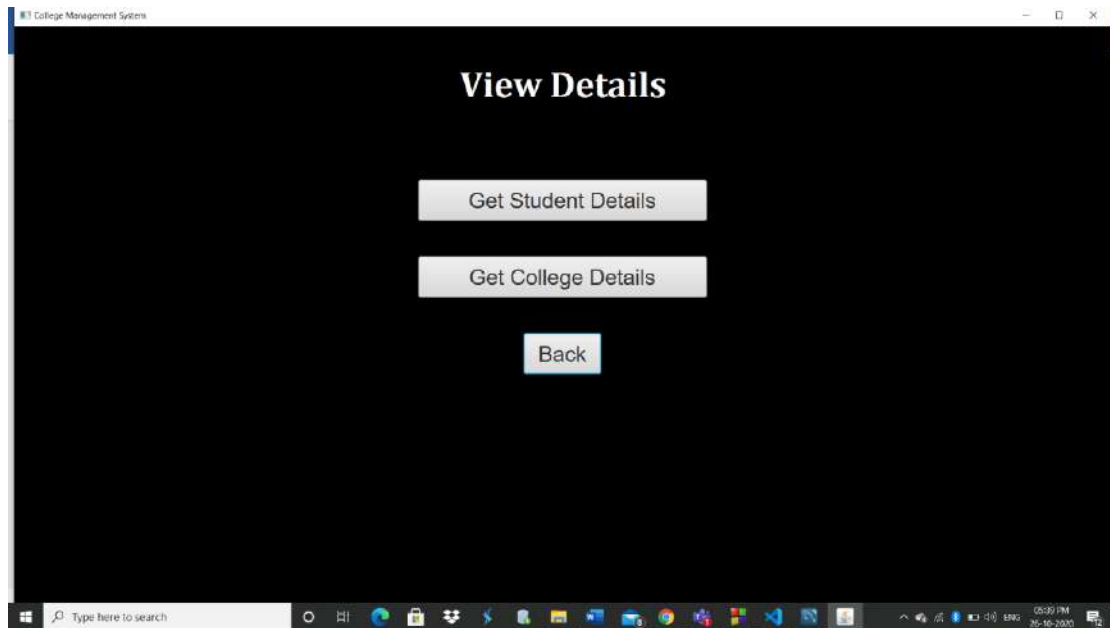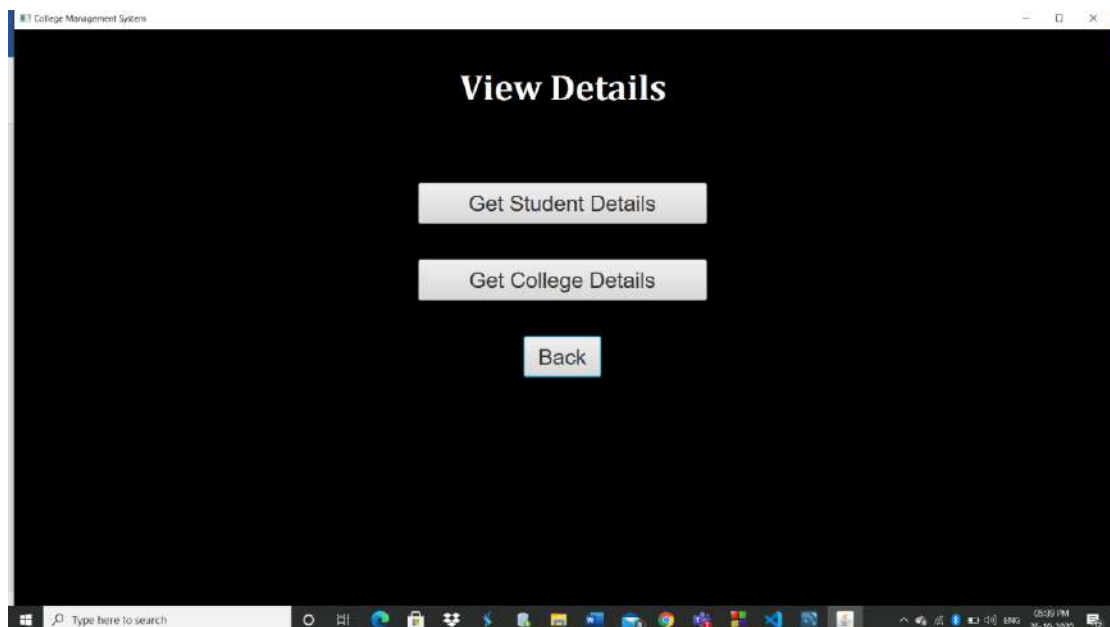

VIEW DETAILS

- Click on View details

- To see Student Details click on Get Student Details
- To see College Details click on Get College Details

VIEW STUDENT DETAILS

- Click on Get Student Details

- Enter Registration Number of Student of view details
- Click on Search

- Details will appear as shown below

VIEW COLLEGE DETAILS

- Click on Get College Details

1. VIEW ADMINISTRATION BOARD DETAILS

- Click on Get Administration Board Details

- Details will appear as shown below
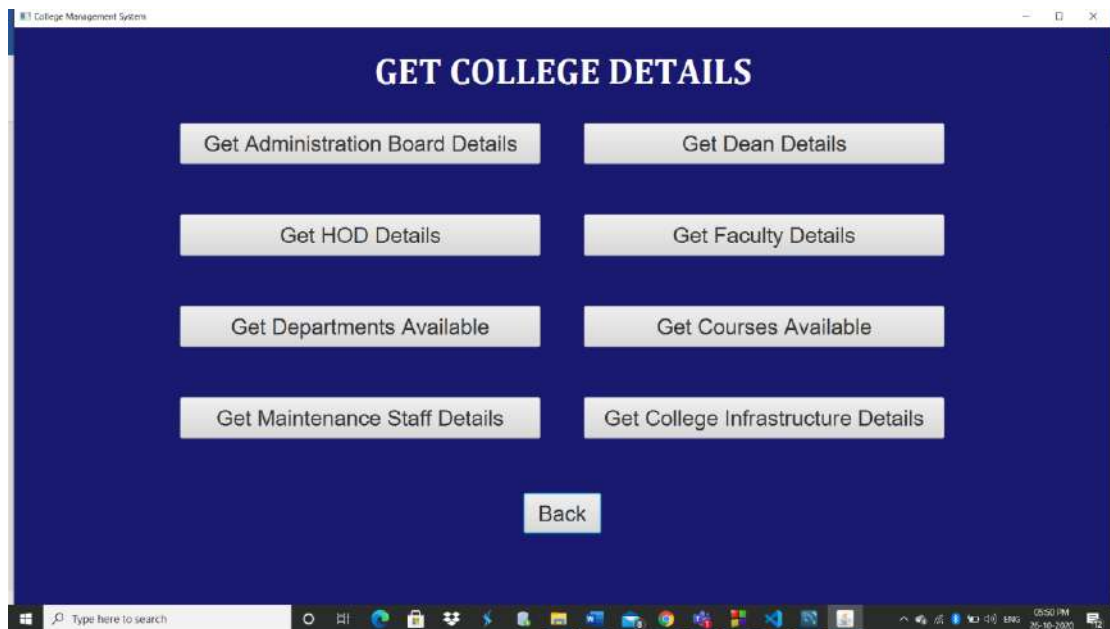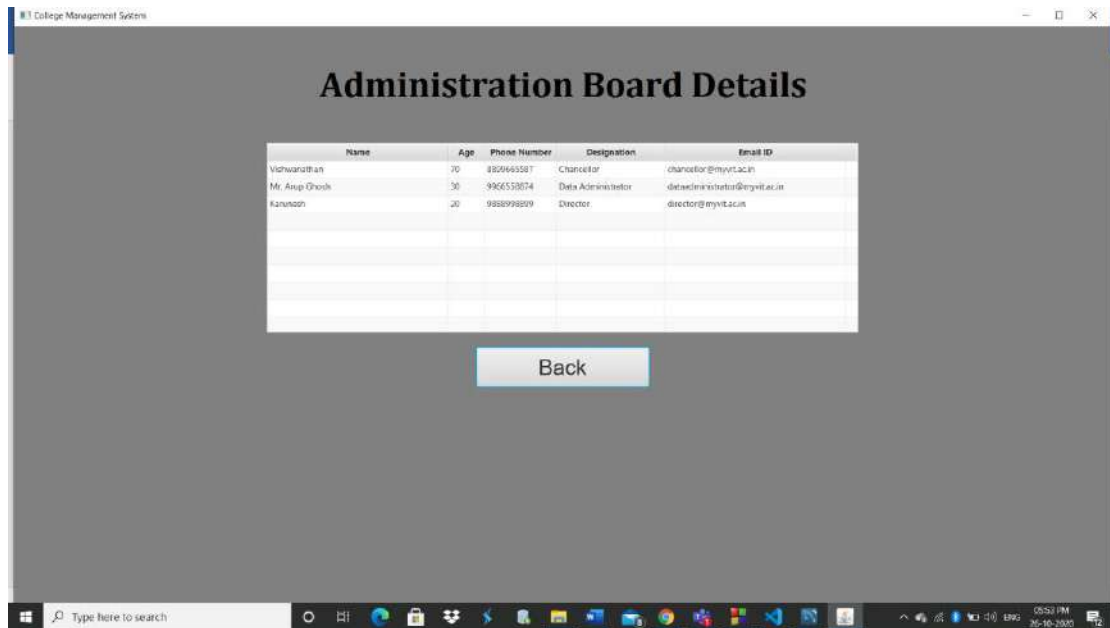- Details has all previously stored data + newly stored data
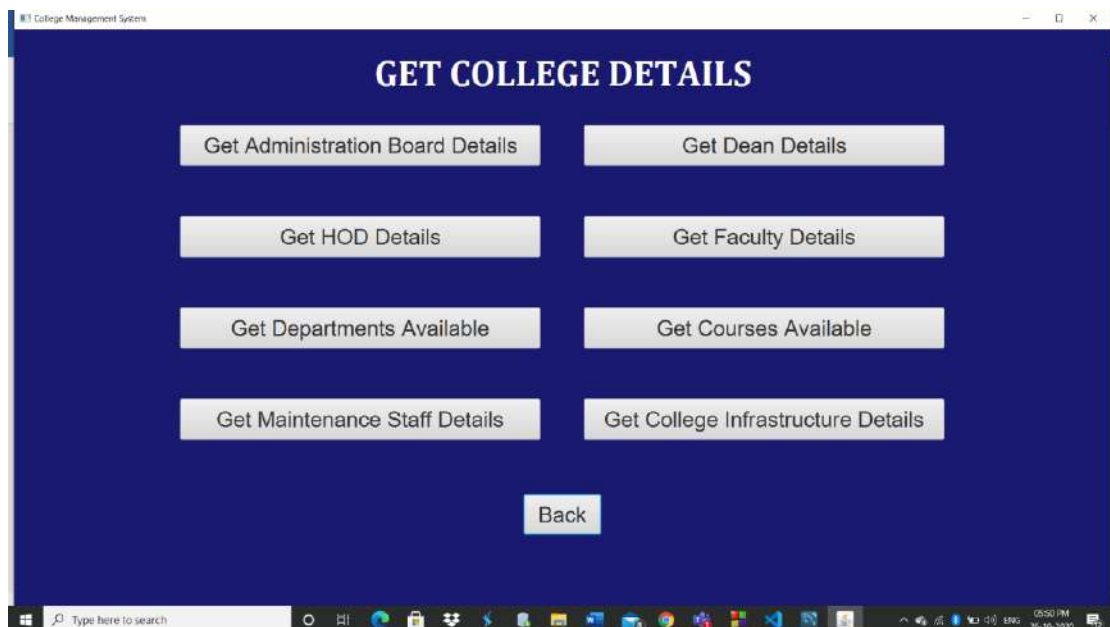

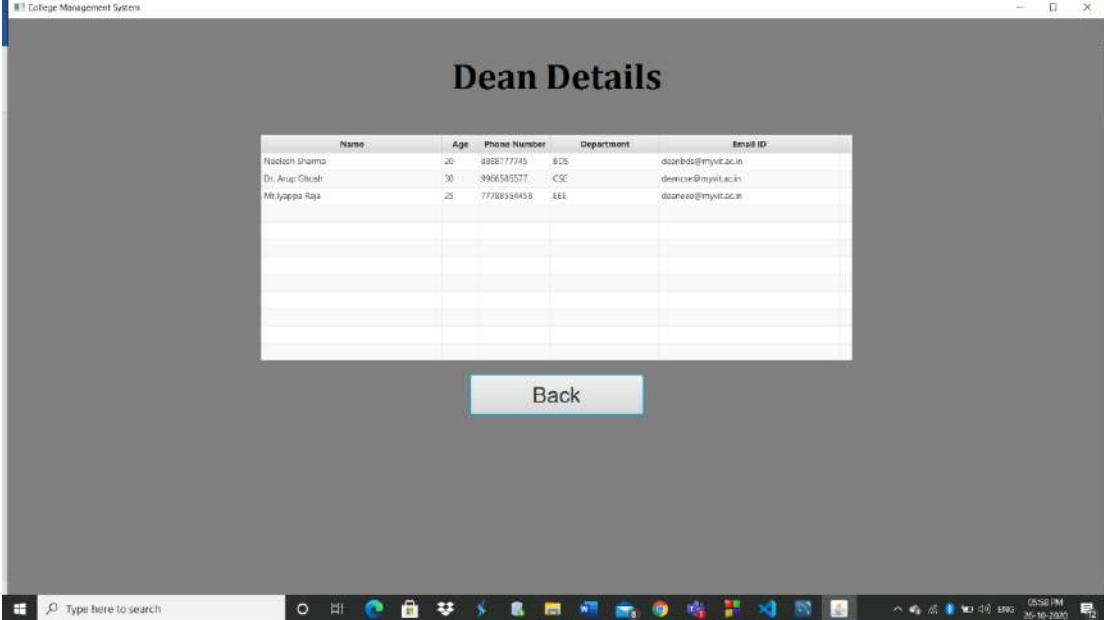## 2. VIEW DEAN DETAILS

- Click on Get Administration Board Details

- Details will appear as shown below
- Details has all previously stored data + newly stored data

3. VIEW HOD DETAILS

- Click on Get HOD Details

- Details will appear as shown below
- Details has all previously stored data + newly stored data

4. VIEW FACULTY DETAILS

- Click on Get Faculty Details

- Details will appear as shown below
- Details has all previously stored data + newly stored data

5. VIEW DEPARTMENTS IN COLLEGE

- Click on Get Departments Available

- Details will appear as shown below
- Details has all previously stored data + newly stored data

6. VIEW COURSES IN COLLEGE

- Click on Get Courses Available

- Details will appear as shown below
- Details has all previously stored data + newly stored data

7. VIEW MAINTENANCE STAFF DETAILS

- Click on Get Maintenance Staff Details

- Details will appear as shown below
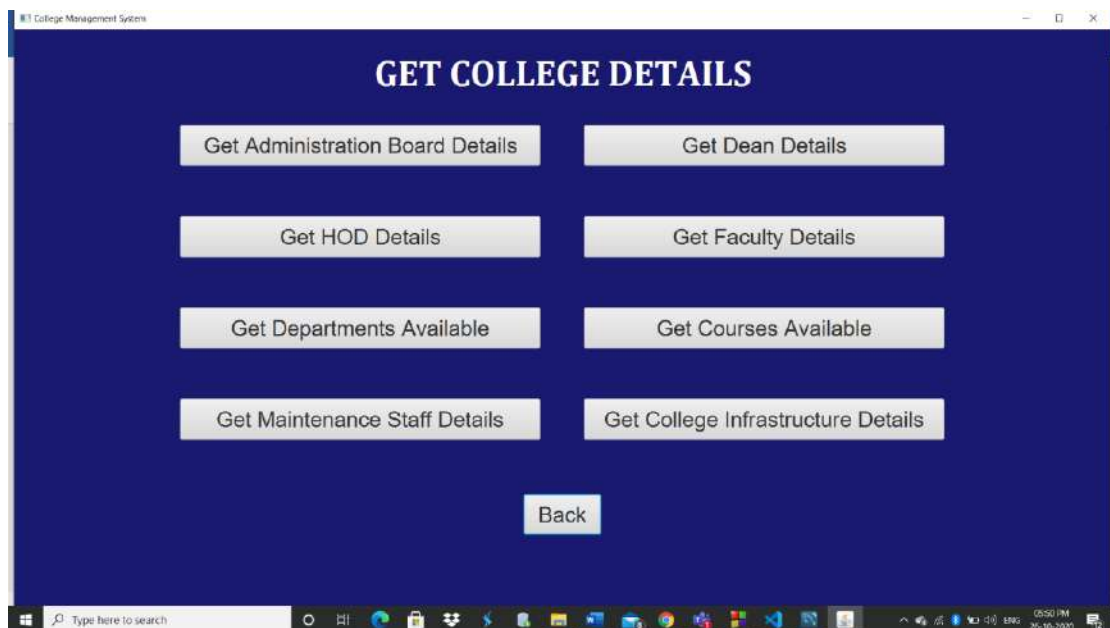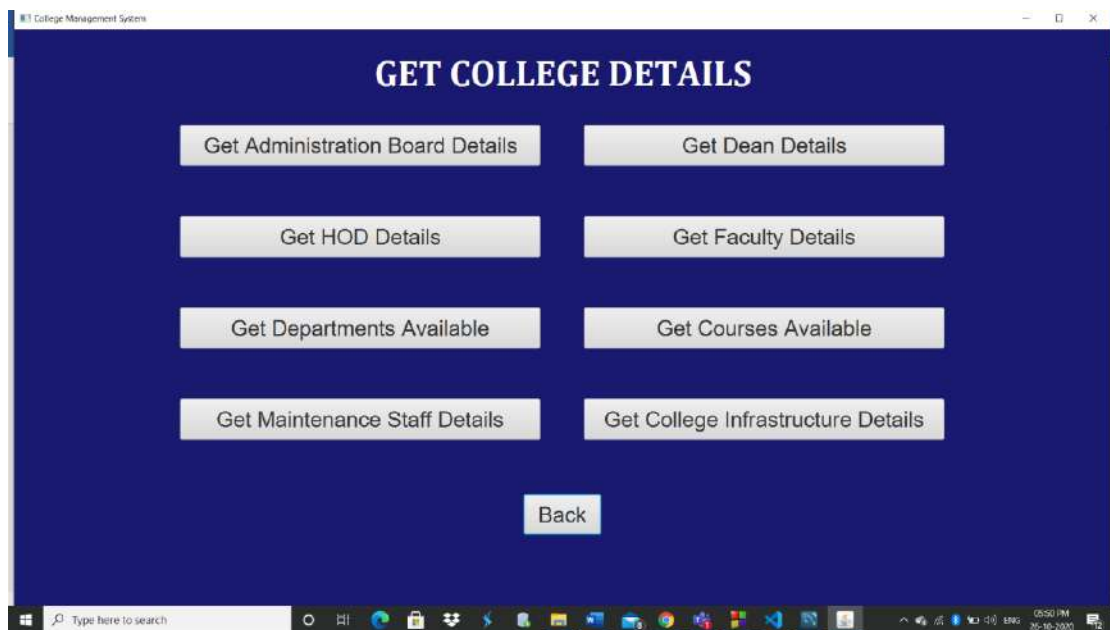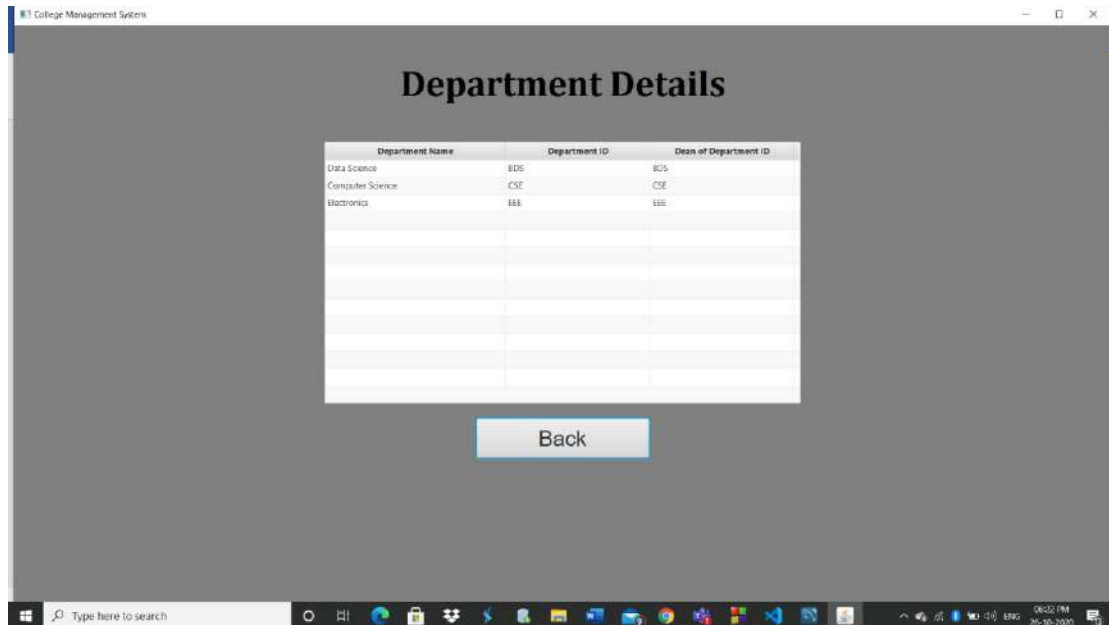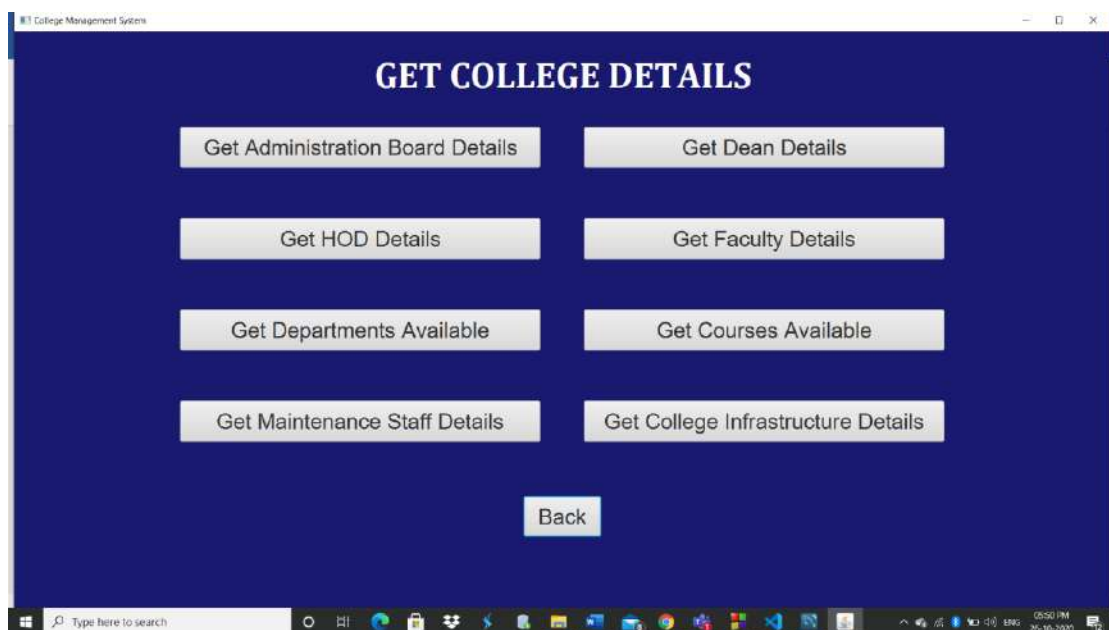- Details has all previously stored data + newly stored data

8. VIEW COLLEGE INFRASTRUCTURE DETAILS

- Click on Get College Infrastructure Details

- Details will appear as shown below
- Details consist of all:-
  1. Academic Building Details
  2. Hostel Building Details
  3. Shops Details
- Details has all previously stored data + newly stored data

FUTURE PROSPECTS

1. The database can be designed in such a way that monitoring student's performance becomes easier.
2. Online examination Module can be introduced here to Conduct Online exams .
3. The college can create a separate section which is visible to students and where faculties can upload notes and videos which might prove to be benefical for students especially during exam hours.

Conclusion

The project is designed to reduce human effort and keeping in the view the day to day problems faced by a college.
This project is successfully implemented with all the features mentioned in system requirements specification.
The application provides appropriate information to users according to the chosen service.
Deployment of our application will certainly help the college to reduce unnecessary wastage of time in personally going to each department for basic information.
Awareness and right data about any institution is necessary for the development of both student as well as the faculty. So this serves the right purpose in achieving the desired requirements of both the communities.