**Student Attendance system**

A project submitted to

**UKA TARSADIA UNIVERSITY**

in partial fulfillment of the requirements for the degree of

**Bachelor of Science**

in

**Information Technology**

for

**5 Years Integrated M.Sc.(IT)**

By

**Monis Pathan**

(Enr. No.201906100110018)

and

**Urvish Sojitra**

(Enr. No.201906100110036)

Guided by

**Rakesh Savant**

**<<Assistant Professor>>**



**Babu Madhav Institute of Information Technology**

**Uka Tarsadia University**

**Bardoli – 394350**

**September 2020**

**CERTIFICATE**

This is to certify that **Monis Pathan ,201906100110018 and Urvish Sojitra,201906100110036** has submitted project entitled **“Student Attendance System”** as the partial fulfillment for the award of the degree of Bachelor of Science in Information Technology for 5 Years Integrated M.Sc.(IT) in 2020 – 2021.

**Date:**

**Place:**

**Rakesh Savant Dr. Jitendra Nasriwala**

Guide Programme Coordinator,

**External Examiner**



**Babu Madhav Institute of Information Technology,**

**Uka Tarsadia University,**

**Bardoli – 394350**

**Table of Content**

| **Chapters** | **Particulars** | | | **Page no.** |
| --- | --- | --- | --- | --- |
| **1** | **Introduction** | | | 6-8 |
| 1.1 | Problem Definition | |  |
| 1.2 | Initial Requirement Document | |  |
| 1.3 | Project Objective | |  |
| 1.4 | Product Scope | |  |
|  | | | | |
| **2** | **Overall Description** | | | 9 |
| 2.1 | Product Perspective/ Environment Description | |  |
|  | 2.1.1 | Hardware Interface/ Hardware Specification |  |
|  | 2.1.2 | Software Interface/ Software Specification |  |
|  | | | | |
| **3** | **System Planning** | | | 10 |
| 3.1 | Software Engineering Model | |  |
|  | 3.2 | CPM PERT | |  |
|  | | | | |
| **4** | **System Specific Requirements** | | | 11-20 |
| 4.1 | Functional Requirement | |  |
| 4.2 | Non-functional Requirement | |  |
|  | | | | |
| **5** | **System Analysis** | | | 21-26 |
| 5.1 | Use case Diagram | |  |
| 5.2 | Activity Diagrams | |  |
|  | | | | |
| **6** | **System Design** | | | 27-39 |
| 6.1 | Database Design | |  |
| 6.1.1 | Database Schema |  |
| 6.1.2 | Data Dictionary |  |
|  | | | | |
| **7** | **System Implementation** | | |  |
| 7.1 | Screenshots | |  |
|  | | | | |
| **8** | **Testing** | | |  |
| 8.1 | Test Cases | |  |
|  | | | | |
| **9** | **Future Enhancement** | | |  |
|  |  | | |  |
|  | **Conclusion** | | |  |
|  |  | | |  |
|  | **Bibliography** | | |  |

**List of Figures**

| **Figure no.** | **Figure Description** | **Page no.** |
| --- | --- | --- |
| **5.1.1** | **Use Case Diagram for Admin** |  |
| **5.1.2** |  |  |
| **5.1.3** |  |  |
| **5.2.1** | **Activity Diagram for Manage Product** |  |
| **5.2.2** |  |  |
| **5.2.3** |  |  |
| **5.2.4** |  |  |
| **5.2.5** |  |  |
| **5.3.1** | **Sequence Diagram for “adding a product”** |  |

**List of Tables**

| **Table no.** | **Table Description** | **Page no.** |
| --- | --- | --- |
|  | Data Dictionary Tables | 50 - 60 |
| 8.1.1 | Test Cases for Login |  |
| 8.1.2 | Test Cases for Product |  |

**Chapter 1: Introduction**

**1.1 Problem Definition**

This system will reduce manual work and avoid redundant data. By maintaining the attendance manually, efficient reports cannot be generated. The system can generate efficient weekly reports based on the attendance.As the attendance is maintained in registers it is a tough task for admin and staff to maintain. Instead the software can keep and retrieve the information when needed.This system also improves on the existing record keeping by introducing newer ways to generate reports

**1.2 Initial Requirement Document**

| **Title of project** | | Student Attendance Management System |
| --- | --- | --- |
| **Stakeholders involved in capturing requirement** | | Project Leader,Guide |
| **Technique used for requirement capturing** | | Brainstorm,Interviewing |
| **Name of the people along with designation** | | Monis pathan,Urvish sojitra |
| **Date** | | August,2021 |
| **Version** | | 1.0 |
| **Consolidated list of Initial Requirements:** | | |
| **Req #1** | System shall be accessed in a single college. | |
| **Req #2** | There shall be three types of users:Admin, Student ,Faculty and Counselor. | |
| **Req #3** | System Shall have one default Admin. | |
| **Req #4** | Admin shall be able to maintain details of students, faculty, academic year, semester, subjects, courses, division,class schedule and reports. | |
| **Req #5** | Admin shall be able to generate a password and a unique username for students and faculty. | |
| **Req #6** | Admin shall be able to assign ROLE to faculty such as counselors. | |
| **Req #7** | Admin shall be able to assign students to counselors. | |
| **Req #8** | Admin shall be able to design class schedules in the system and assign lectures to the Faculty. | |
| **Req #9** | Faculty shall be able to see the class schedule . | |
| **Req #10** | Faculty shall be able to insert attendance records of the students. | |
| **Req #11** | Counselors shall be able to see students assigned to them and their reports. | |
| **Req #12** | Faculty shall be able to view reports. | |
| **Req #13** | Students shall be able to view their profile. | |
| **Req #14** | Students shall be able to update their password. | |
| **Req #15** | System shall be able to generate reports in particular:   * The percentage of present and absent days by subject and lectures of a student * The most absent student ,by lectures.  Average Daily AttendanceAverage Attendance by DayAbsence Summary  * Attendance for given duration * Subject wise absent report * Student wise absent report | |

**1.3 Project Objective**

This project aims to make an effective student attendance management system for teachers,and to

1. To reduce the cost of operation
2. To Reduce time consumption
3. To Reduce human error
4. To reduce paperwork
5. To make report generation easy
6. To improve efficiency
7. To Improve data security

**1.4 Project Scope**

The scope of the system is confined to one college in one department containing multiple courses which is an internet-based system and it only contains required functionality of the attendance system and not external modules like student grade management or marksheet generation. This system does not consist of the role of Director and does not consist of leave management.The system also does not contain elective subjects.

**2 Overall Description**

**2.1 Product Perspective**

The Product is a web-based system implementing a client-server model.

The Student Attendance Management System provides a simple way for managing student attendance.

The following are the main features that are included in the Student attendance system:

* User account: The system allows the user to create their accounts in the system and provide features of updating and viewing profiles.
* Number of users being supported by the system: System is able to support a large number of online users at a time.
* Report generation:The system can generate reports like weekly attendance,student absence.

**2.1.1 Hardware Interface/ Hardware Specification**

1. Screen resolution of at least 640 x 480 or above.
2. Computer System will be in a networked environment.
3. Minimum ram required 4 gb
4. Minimum memory of 500mb

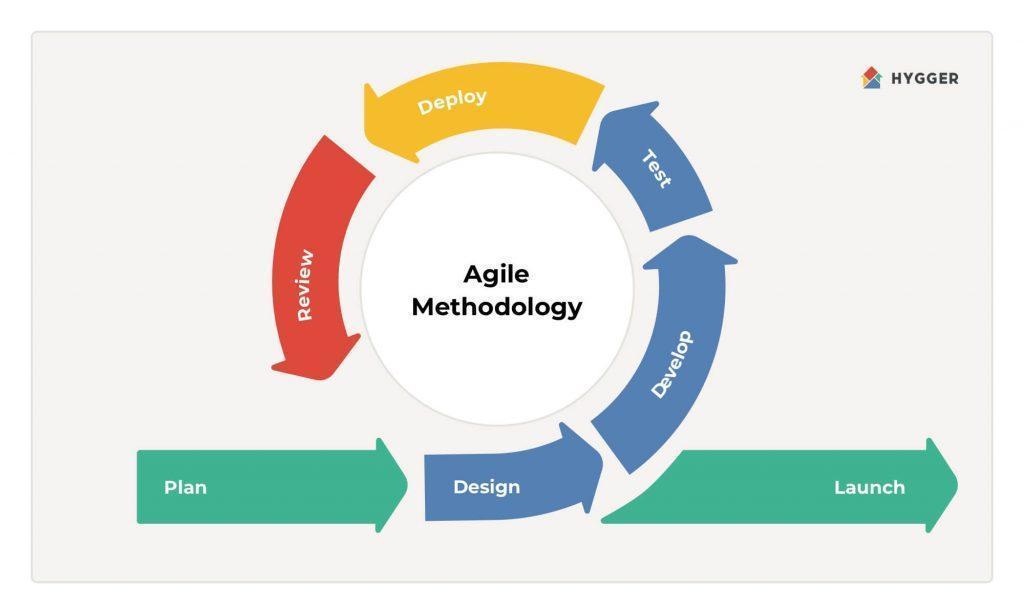
**2.1.2 Software Interface/ Software Specification**

1. MS-Windows Operating System (7/8.1/10)
2. HTML,css designing front-end
3. PHP,Ms Sql Server 2019 for back end

**3.System Planning**

**3.1 Software Engineering Model**

Software engineering model used to build the proposed system is “Agile Software Development model”



**4.System Specific Requirements**

**4.1 Functional Requirements**

1. **Manage Login/Registration**
2. **Manage Profile**
3. **Manage academic details** 
   1. Academic year
   2. Course
   3. Semester\_Division
   4. Subject
4. **Manage Faculty**
5. **Manage Student**
6. **Assign academic detail**
7. Academic year
8. Course
9. Semester\_Division
10. Subject
11. **Manage Counselor**
12. **Manage Class Schedule**
13. **Manage Attendance**
14. **Generate reports**

**1.Manage Login/Registration**

| **RN** | **Description** | **Comments**  **(operation)** |
| --- | --- | --- |
| **FR1** | System shall allow Admin to register Students and Faculty either manually or using csv file | Add student page,  Add Faculty page |
| **FR2** | System shall validate Admin ,Student and faculty with username, password, role and status and only then allow them to login into the system. | Login page |
| **FR3** | System shall provide ‘forget password’ functionality to valid Students,faculty and who have forgotten their password.  Password shall be reset by sending a confirmation email | Forget password page  (OTP by mail) |
| **FR4** | To manage this module the following operations shall be performed by the:  Admin:   * Shall be able to Add Student and Faculty details * Shall be able to view Student and Faculty details * Shall be able to Update Student and faculty details * Shall be able to assign counselor role to faculty * shall be able to set status as active or inactive | Add Student page  Add Faculty page  Assign counselor page |

**2. Manage Profile**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| **FR1** | After successful registration by Admin,Student and Faculty shall be able to view their profile | profile page |
| **FR2** | Valid admin,students and faculty shall be able to update their passwords using ‘change password’ | profile page  ‘change password |
| **FR3** | If student and Faculty wants to update their profile details ,they have to file a written application and and give it to admin | student page  faculty page |

**3. Manage academic details**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| **FR1** | In this module admin shall manage:   * academic year * course * semester * division * subject | pages involved:  Add academic year  Add course  Add semester  Add division  Add subject |
| **FR2** | To manage this module Admin shall:   * Add new details like academic year, course, semester, division and subject * Update existing details * View existing Details | pages involved:  Add academic year  Add course  Add semester  Add division  Add subject |

**4.Manage Faculty**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| **FR1** | In this module admin shall be able to:   * Add new faculty details * Update existing faculty details * View and search existing Faculty details * Set faculty status as active or inactive * Assign subject to Faculty * Assign counselor role to faculty * set class schedule | Add faculty page  Assign Counselor page  Class Schedule page |
| **FR2** | Inactive faculty can not access the system | login denied |
| **FR3** | Faculty shall be able to view details assigned by admin   * academic year * course * semester * division * subject * add attendance * view attendance * update attendance * Class Schedule | Faculty page  attendance page  Class Schedule page |

**5.Manage Students**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| FR1 | In this module admin shall be able to:   * Add new student details * Update existing students details * View and search existing students details * Set student status as active or inactive | Add student page |
| FR2 | Inactive Students cannot access the system | login denied |
| FR3 | After Allocation Students shall be able to view details assigned by admin.   * academic year * course * semester * division | student page |

**6.Assign academic detail**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| **FR1** | After adding academic details and faculty  Admin shall allocate these details to faculty:   * academic year * course * semester * division * subject   Faculty shall be able to view details assigned by admin | Dashboard and Faculty profile |
| **FR2** | After adding academic details and Student  Admin shall allocate these details to Student:   * academic year * course * semester * division   Students shall be able to view details assigned by the admin. | Dashboard and student profile |

**7.Manage Counselor**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| **FR1** | Admin shall assign counselor role to faculty | Assign Counselor |
| **FR2** | Admin shall assign students to counselors | Assign Counselor |
| **FR3** | Admin shall view number of active or inactive counselors | Dashboard  Assign Counselor |
| **FR4** | Counselor can see the students assigned to them and their reports | Counselor page |

**8.Manage Class Schedule**

| **RN** | **D**odule the following operations shall b**escription** | **Comments** |
| --- | --- | --- |
| FR1 | To manage this m  update thee performed by the Admin:   * Schedule the lecture in the Class Schedule lectures in Class Schedule | Class Schedule |
| FR2 | Admin shall assign subject division and course to faculty and make the Class Schedule based on these parameters. | Assign subject  Class Schedule page |
| FR3 | After lectures have been assigned Faculty shall be able to see their Class Schedule | Class Schedule page |
| FR4 | This module will contain:   * subject * semester * Start time of lecture * End time of lecture | Class Schedule page |

**9.Manage Attendance**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| FR1 | After allocation of academic details faculty can take attendance  Academic details:   * academic year * course * semester * division * subject | Attendance |
| FR2 | Faculty can set whether the student was present or absent during their assigned lectures.  Faculty can enter attendance by the subject assigned to them | Attendance page |
| FR3 | After taking attendance Faculty shall be able to view and update attendance till end of week | Attendance page |

**10. Generate reports**

| **RN** | **Description** | **Comments** |
| --- | --- | --- |
| FR1 | System shall generate reports based on the attendance taken by faculty | Report page |
| FR2 | Faculty can view reports based on their subject and role like counselor | Report page |
| FR3 | Reports shall be generated on:   * The percentage of present and absent days by subject and lectures of a student * The most absent student ,by lectures.  Average Daily AttendanceAverage Attendance by DayAttendance ChartAbsence Summary  * Attendance for given duration * Subject wise absent report * Student wise absent report   Admin can view all the reports | Report page |
| FR4 | Admin and Faculty can download the required reports in pdf form | Report page |
| FR5 | To manage this module the following operations shall be performed by the Faculty and admin:   * View the reports appropriate to them * Faculty can see reports on the students assigned to them * Counselor shall be able to see the reports of students assigned to them | Report page |

**4.2 Non-Functional Requirements**

**1)Performance**

Easy tracking of records and updating can be done due to:

This software uses an underlying database which shall be on a central server while the frontend shall be available to Administrative staff ,faculty and students.

The Software can handle many users at the same time.

**2)Reliability**

This software shall not be able to connect to the centralized database in the event that the Server is down due to hardware or software failure.

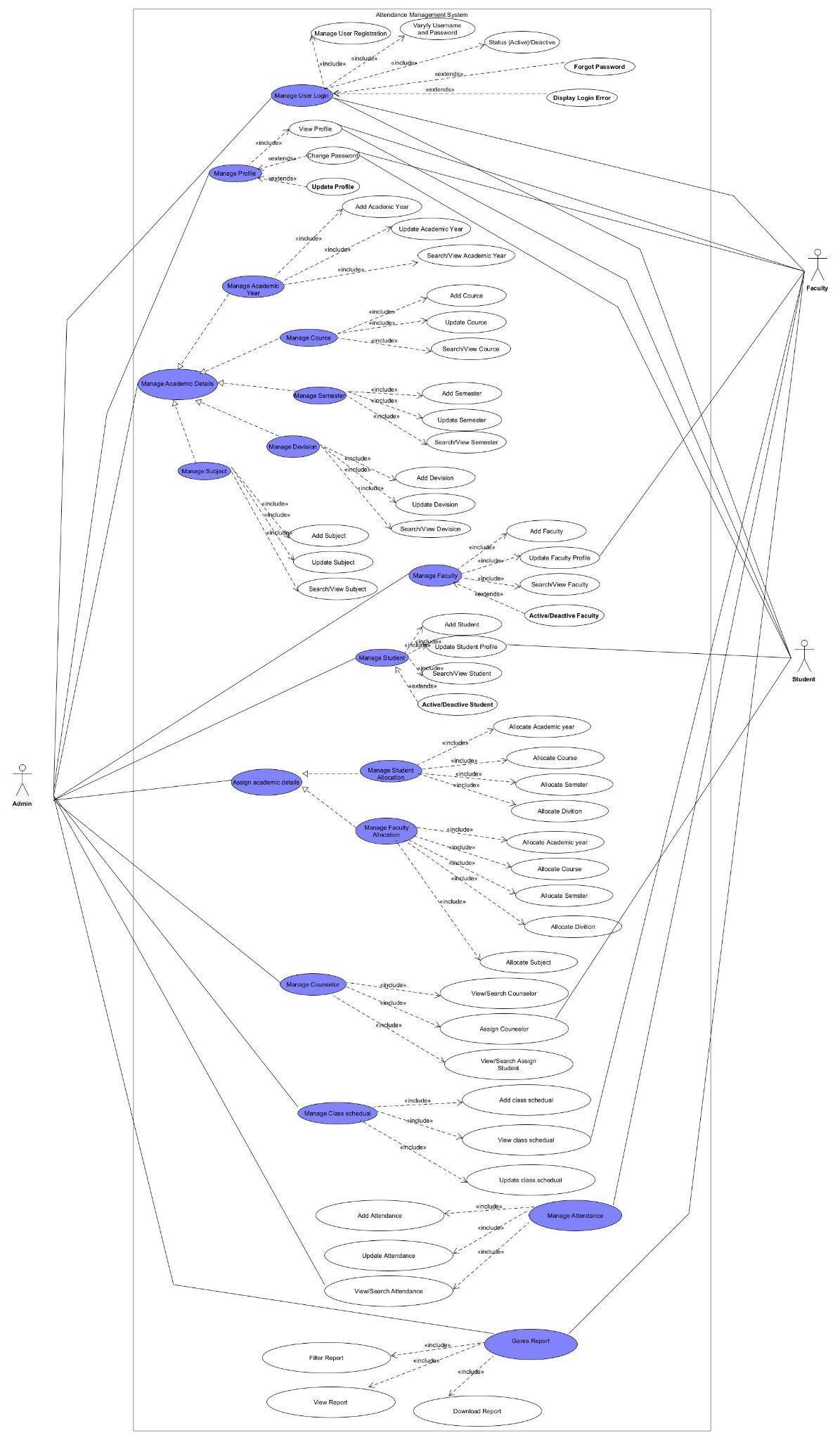
**3)Availability**

The software shall be only available to authorized users of the college like Teachers who shall mark the attendance of students,Students to check their leave and Admin to maintain the system.

**5: SystemAnalysis**

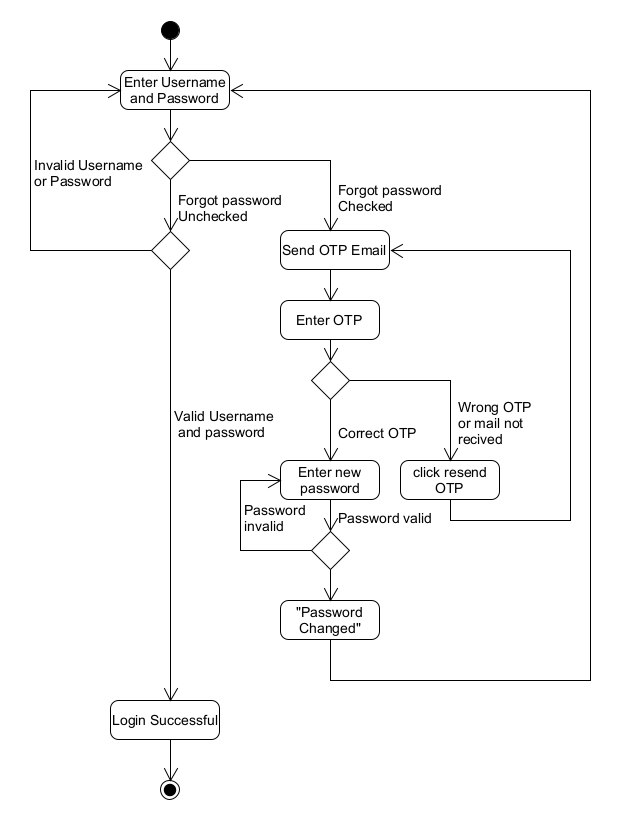
**5.1 Use CaseDiagram**

**Login Use case Diagram**

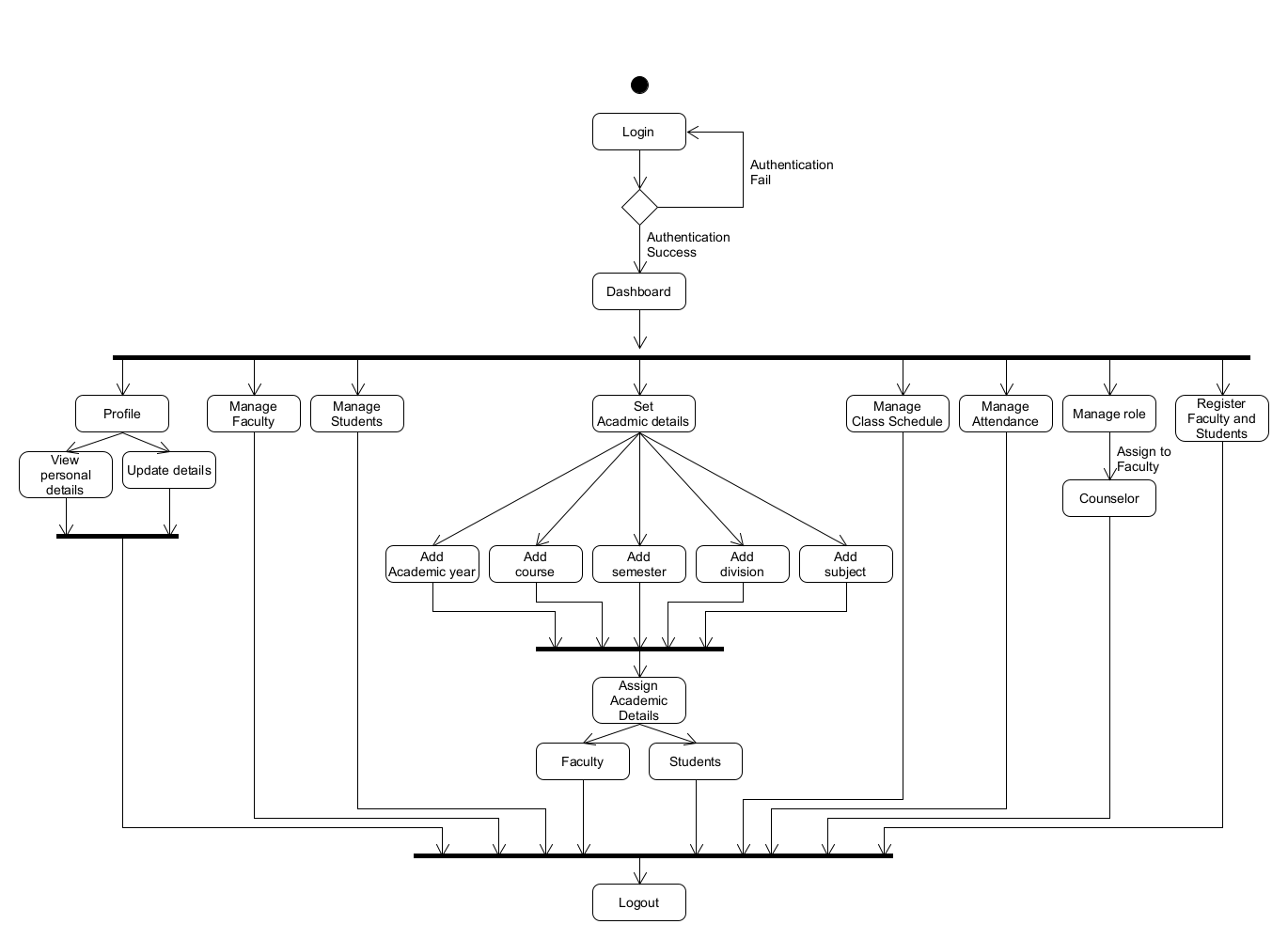


**5.2 Activity diagram**

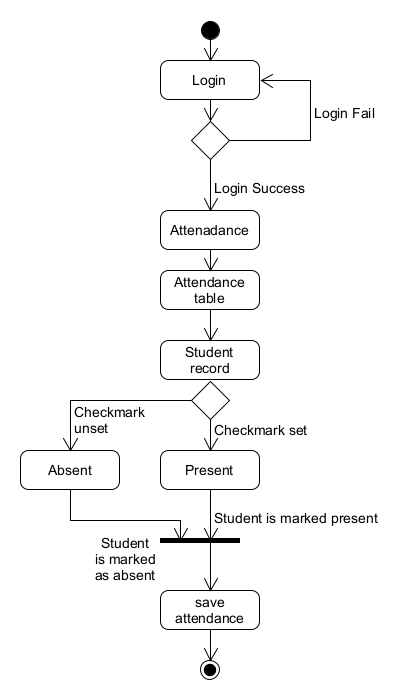
**Activity diagram for Login**



**Activity diagram for Admin**



**Activity diagram for Attendance**



**6: System Design**

**6.1 Database Design**

**6.1.1 Database Schema**

**6.1.1.1 tbl\_academicYear:-**

tbl\_academicYear(AcademicYearID[PK],AcademicYear,AcademicYearCreatedAt,AcademicYearEndAt)

FD:AcademicYearID→ {AcademicYear,AcademicYearCreatedAt,AcademicYearEndAt}

**6.1.1.2 tbl\_course**

tbl\_course(CourseID[PK],CourseName,CourseDescription,CourseCreatedAt)

FD:CourseID→ {CourseName,CourseDescription,CourseCreatedAt}

**6.1.1.3 tbl\_semester\_division**

tbl\_semester\_division(SemesterDivisionID[PK],SemesterName,DivisionName)

FD: SemesterDivisionID→{SemesterName,DivisionName}

**6.1.1.4 tbl\_subject**

tbl\_subject(SubjectID[PK],SubjectName)

FD:SubjectID→{SubjectName}

**6.1.1.5 tbl\_user**

tbl\_user(UserID[PK],UserName,Password,Usertype,UserStatus,UserCreaetedAt,UserDeletedAt)

FD:UserID→{UserName,UserPassword,Usertype,UserStatus,UserCreaetedAt,UserDeletedAt}

**6.1.6 tbl\_state**

tbl\_state(StateID[PK],StateName)

FD:StateID->{StateName}

**6.1.7 tbl\_city**

tbl\_city(CityID[PK],CityName,SateID[FK])

FD:CityID->{SateID[FK]}

**6.1.1.8 tbl\_admin**

tbl\_admin(AdminID[PK],AdminFname,AdminMname,,AdminLnameAdminPhoto,AdminGender,AdminDOB,AdminlocalAddress,AdminpermanantlAddress,AdminCaste,AdminSubCaste,AdminBloodGroup,AdminEmail,AdminContactNo,AdminCity,AdminState,AdminMaritalStatus,AdminSpouse,AdminDisability,AdminAadharcardno,AdminHighestQualification,UserID[FK])

FD:AdminID→{AdminFname,AdminMname,AdminLnameAdminPhoto,AdminGender,AdminDOB,AdminlocalAddress,AdminpermanantlAddress,AdminCaste,AdminSubCaste,AdminBloodGroup,AdminEmail,AdminContactNo,AdminCity,AdminState,AdminMaritalStatus,AdminSpouse,AdminDisability,AdminAadharcardno,AdminHighestQualification,UserID[FK])

**6.1.1.9 tbl\_faculty**

tbl\_faculty(FacultyID[PK],FacultyFname,FacultyMname,FacultyLname,,FacultyPhoto,FacultyGender,FacultyDOB,FacultyLocalAddress,FacultyPermanantAddress,FacultyCaste,FacultySubCaste,FacultyBloodGroup,FacultyEmail,FacultyContactNo,FacultyStateCityID,FacultyMaritalStatus,FacultySpouse,FacultyDisabilty,FacultyAadharcardno,FacultyHighestQualification,AcadmicYearID[FK],FacultyCourseID[FK],UserID[FK])

FD:FacultyID→{FacultyFname,FacultyMname,FacultyLname,,FacultyPhoto,FacultyGender,FacultyDOB,FacultyLocalAddress,FacultyPermanantAddress,FacultyCaste,FacultySubCaste,FacultyBloodGroup,FacultyEmail,FacultyContactNo,FacultyStateCityID,FacultyMaritalStatus,FacultySpouse,FacultyDisabilty,FacultyAadharcardno,FacultyHighestQualification,AcadmicYearID[FK],FacultyCourseID[FK],UserID[FK]}

**6.1.1.10 tbl\_students**

tbl\_students\_details(StudentID[PK],StudentEnrollmentNo,StudentFname,StudentMname,StudentLname,StudentPhoto,StudentGender,StudentPermanantAddress,StudentLocalAddress,StudentFatherName,StudentFatherContactNo,StudentMotherName,StudentMotherContactNo,StudentCaste,StudentSubCaste,StudentBloodGroup,StudentEmail,StudentContactNo,StudentStateCityID[FK],StudentAadharcardno,StudentHighestQualification,,EnrollYear,AcademicYearID[FK],CourseSemesterID[FK],UserID[FK])

FD:StudentID→{StudentEnrollmentNo,StudentFname,StudentMname,StudentLname,StudentPhoto,StudentGender,StudentPermanantAddress,StudentLocalAddress,StudentFatherName,StudentFatherContactNo,StudentMotherName,StudentMotherContactNo,StudentCaste,StudentSubCaste,StudentBloodGroup,StudentEmail,StudentContactNo,StudentStateCityID[FK],StudentAadharcardno,StudentHighestQualification,,EnrollYear,AcademicYearID[FK],CourseSemesterID[FK],UserID[FK]}

**6.1.1.11 tbl\_course\_semester**

tbl\_course\_semester(CourseSemesterID[PK],AcademicYearID[FK],CourseID[FK],

SemesterDivisionID[FK],SemesterCreatedDate,SemesterEndDate)

FD:CourseSemesterID→{AcademicYearID[FK],CourseID[FK],SemesterDivisionID[FK],SemesterCreatedDate,SemesterEndDate}

**6.1.1.12 tbl\_semester\_subject**

tbl\_semester\_subject(SemesterSubjectID[PK],CourseSemesterID[FK],SubjectID[FK],SubjectCode,SubjectDescription,SubjectType,SubjectCreatedDate,SubjectEndDate)

FD:SemesterSubjectID→{CourseSemesterID[FK],SubjectID[FK],SubjectCode,SubjectDescription,SubjectType,SubjectCreatedDate,SubjectEndDate}

**6.1.1.13 tbl\_faculty\_subject\_allocation**

tbl\_faculty\_subject\_allocation(FacultyAllocationID[PK],FacultyID[FK],SemesterSubjectID[FK],SemesterCreatedDate,SemesterEndDate)

FD:FacultyAllocationID→{FacultyID[FK],SemesterSubjectID[FK],SemesterCreatedDate,SemesterEndDate}

**6.1.1.14 tbl\_counselor\_allocation:-**

tbl\_counselor\_allocation(CouncelorAllocationI[PK],FacultyID[FK],StudentID[FK],AllocatedDate,DeallocatedFDate)

FD:CouncelorAllocationID→{FacultyID[FK],StudentID[FK],AllocatedDate,DeallocatedFDate}

**6.1.1.15 tbl\_classschedule:-**

tbl\_class\_schedule(ClassScheduleID[PK],FacultyAllocationID[FK],ClassScheduleDay,LectureFromTime,LectureToTime,ClassScheduleStartedDate,ClassScheduleEndedDate)

FD:ClassScheduleID→{,FacultyAllocationID[FK],ClassScheduleDay,LectureFromTime,LectureToTime,ClassScheduleStartedDate,ClassScheduleEndedDate}

**6.1.1.16 tbl\_attendancemaster:-**

tbl\_attendancemaster(AttendanceMasterID[PK],SemesterSubjectID[FK],FacultyID[FK],AttendanceDate,FromTime,ToTime,ClassScheduleID[FK],Topic)

FD:AttendanceMasterID→{SemesterSubjectID[FK],FacultyID[FK],AttendanceDate,FromTime,ToTime,ClassScheduleID[FK],Topic}

**6.1.1.17 tbl\_attendance:-**

tbl\_attendance(AttendanceID[PK],StudentID[FK],AttendanceMasterID,AttendanceStatus)

FD:AttendanceID→{StudentID[FK],StudentID[FK],AttendanceMasterID,AttendanceStatus}

**6.1.2 Data Dictionary**

**6.1.2.1 tbl\_academicYear:-**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| AcademicyearID | int | 11 | Primary Key | Id for Academic year |
| Academicyear | varchar | 10 | Not null | Current and previous  Academic year |
| AcademicYearCreatedAt | date | - | Not null | Start of Academic Year |
| AcademicYearEndAt | date | - | null allowed | End of Academic Year |

**6.1.2.2 tbl\_Course**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| CourseID | int | 11 | Primary Key | Id number of course |
| CourseName | varchar | 20 | Not null | Name of the course |
| CourseDescription | varchar | 50 | Not null | Description of the course explaining the contents of the course |
| CourseCreatedAt | date | - | Not null | Date of the course creation in dd/mm/yyyy  format |

6.1.2.3 tbl\_semester\_division

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| SemesterDivisionID | int | 11 | Primary Key | Id number of semester |
| SemesterName | int | 2 | Not null | Name of semester like ‘5’ |
| DivisionName | varchar | 2 | Not null | Name of divsion like ‘ A’ |

6.1.2.4 tbl\_subject

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| SubjectID | int | 11 | Primary Key | Id number of subject |
| SubjectName | varchar | 20 | Not null | Name for the subjects |

**6.1.2.5 tbl\_user**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| UserID | int | 11 | Primary Key | Id number |
| UserName | varchar | 320 | Not null | Username of users provided by admin  Enrollment number for Students  ex:’201906100110011’  Email address for Faculty  ex:jhon@gmail.com |
| UserPassword | varchar | 255 | Not null | Password is encrypted |
| Usertype | char | 1 | Not null | User type is faculty,admin or student |
| UserStatus | tiny int | 1 | Not null | Active or inactive |
| UserCreatedAt | date | - | Not null | Date of data creation |
| UserDeletedAt | date | - | Null allowed | Date of inactivating the users |

**6.1.2.6 tbl\_State**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| StateID | int | 11 | Primary Key | Primary key |
| StateName | varchar | 50 | not null | Sate name |

**6.1.2.7 tbl\_City**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| CityID | int | 11 | Primary | Id of city |
| CityName | varchar | 255 | not null | Name of City |
| StateID | int | 11 | Foreign Key | Sate if from tbl\_state |

**6.1.2.8 tbl\_admin**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| AdminID | int | 11 | Primary Key | Admin year |
| AdminFname | varchar | 30 | Not null | First Name of admin |
| AdminMname | varchar | 30 | Not null | Middle Name of admin |
| AdminLname | varchar | 30 | Not null | Last Name or surname of admin |
| AdminPhoto | varchar | 256 | Null allowed | Photo source |
| AdminGender | char | 1 | Not null | One letter for gender |
| AdminDOB | date | - | Not null | Date of birth of admin |
| AdminLocalAddress | varchar | 255 | Null Allowed | Full address of admin |
| AdminpermanentAddress | varchar | 255 | Not null | Original Full address of admin |
| AdminCaste | varchar | 10 | Not null | Caste of admin |
| AdminSubCaste | varchar | 10 | Not null | SubCaste of admin |
| AdminBloodGroup | varchar | 3 | Not null | Blood group |
| AdminEmail | varchar | 320 | Not null,Unique | Email of admin |
| AdminContactNo | bigint | 10 | Not null,Unique | Mobile number |
| AdminCity | varchar | 255 | Not null | Name of the city |
| AdminState | varchar | 255 | Not null | Name of State |
| AdminMaritalStatus | tinyint | 1 | Not null | If Admin is married or not |
| AdminSpouse | varchar | 70 | Null allowed | Name of spouse if admin is married |
| AdminDisabilty | varchar | 255 | Null Allowed | If Admin has a disability state what it is and  if no then leave it empty |
| AdminAadharcardno | bigint | 12 | Not null,Unique | Aadhar card number |
| Admin Highest Qualification | varchar | 70 | Null allowed | Highest Qualification of admin |
| UserID[FK] | int | 11 | Foreign key | Foreign key  Id from  tbl\_user |

**6.1.2.7 tbl\_faculty**

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| FacultyID | int | 11 | Primary key | id number |
| FacultyFname | varchar | 30 | Not null | First name of faculty |
| FacultyMname | varchar | 30 | Not null | Middle name of faculty |
| FacultyLname | varchar | 30 | Not null | Last name of faculty |
| FacultyPhoto | varchar | 50 | Null allowed | Photo source |
| FacultyGender | char | 1 | Not null | Gender in one character |
| FacultyDOB | date | - | Not null | Date of birth |
| FacultyLocalAddress | varchar | 255 | Not null | Full Local address |
| FacultyPermanentAddress | varchar | 255 | Not null | Original Full address |
| FacultyCaste | varchar | 15 | Not null | Caste of faculty |
| FacultySubCaste | varchar | 20 | Not null | SubCaste of faculty |
| FacultyBloodGroup | varchar | 3 | Not null | Blood group |
| FacultyEmail | varchar | 256 | Not null,unique | email |
| FacultyContactNo | bigint | 10 | Not null,unique | Mobile number of faculty |
| FacultyStateCityID | int | 11 | Foreign key | City and State id from tbl\_state |
| FacultyMaritalStatus | tinyint | 1 | Not null | if faculty is married or not |
| FacultySpouse | varchar | 70 | Null allowed | Name of spouse if admin is |
| FacultyDisability | varchar | 255 | Null allowed | If Faculty has a disability state what it is and  if no then leave it empty |
| FacultyAadharcardNo | bigint | 12 | null allowed,unique | Aadhar card no |
| FacultyHighestQualification | varchar | 70 | not null | Highest Qualification of faculty |
| Academic  YearID | int | 11 | Foreign key | academic year id |
| FacultyCourseID | int | 11 | Foreign key | Faculty course id |
| UserID[FK] | int | 11 | Foreign key | Foreign key form tbl\_User |

6.1.2.8 tbl\_students

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| StudentID | int | 11 | Primary key | Id for student |
| StudentEnrollment | bigint | 15 | Not null | Enrollment number of student |
| StudenFname | varchar | 30 | Not null | First name of student |
| StudenMname | varchar | 30 | Not null | Middle name of student |
| StudenLname | varchar | 30 | Not null | Last name of student |
| StudentPhoto | varchar | 256 | Null allowed | Source of photo(location) |
| StudentGender | char | 1 | Not null | Single character for gender |
| StudentDOB | date | - | Not null | Date of birth of student |
| StudentPermanantAddress | varchar | 256 | Not null | Where students  originally are from |
| StudentLocalAddress | varchar | 256 | Not null | Where student currently reside in |
| StudentFatherName | varchar | 70 | Not null | Full Name of father |
| StudentFatherContactNo | bigint | 10 | Not null | Father’s mobile number |
| StudentMotherName | varchar | 35 | Not null | Mothers full name |
| StudentMotherContactNo | int | - | Not null | Mother’s mobile number |
| StudentCaste | varchar | 10 | Not null | Caste of student |
| StudentSubCaste | varchar | 10 | Not null | Sub Caste of student |
| StudentBloodGroup | varchar | 3 | Not null | Blood group of student |
| StudentEmail | varchar | 256 | Not null | Email address |
| StudentContactNo | int | - | Not null | Mobile number of student |
| StudentStateCityID | int | 11 | Foreign key | State and City where student resides |
| StudentAadharcardno | bigint | 12 | Not null | Aadhar card no of student |
| StudentHighestQualification | varchar | 70 | Not null | Highest qualification of student |
| EnrollYear | varchar | 2 | Not null | Year when student is admitted in college |
| AcademicYearID | int | 11 | Foreign key | Foreign key id  From tbl\_academic-year |
| CourseSemesterID | int | 11 | Foreign key | Foreign key  from tbl\_course\_semester |
| UserID | int | - | Foreign key | Foreign key id from tbl\_users |

6.1.2.9 tbl\_Course\_semester

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| CourseSemesterID | int | 11 | Primary Key | Id number |
| AcademicYearID | int | 11 | Foreign key | Foreign key  Id from tbl\_academicYear |
| CourseID | int | 11 | Foreign key | Foreign key  Id from tbl\_Course |
| SemesterDivisionID | int | 11 | Foreign key | Foreign key  Id from  tbl\_semester\_division |
| SemesterCreatedDate | date | - | not null | Creation date  Date in dd/mm/yyyy  Format |
| SemesterEndDate | date | - | Null allowed | End date  Date in dd/mm/yyyy  Format |

6.1.2.10 tbl\_semester\_subject

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| SemesterSubjectID | int | 11 | Primary Key | Id for subject |
| CourseSemesterID | int | 11 | Foreign Key | Foreign key  Id from  tbl\_Course\_semester |
| SubjectID | int | 11 | Foreign Key | Foreign key  Id from  tbl\_subject |
| SubjectCode | varchar | 10 | Not null | Subject code assigned for semester |
| SubjectType | varchar | 3 | Not null | Theory or practical subject types |

6.1.2.11 tbl\_faculty\_subject\_allocation

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| FacultyAllocationID | int | 11 | Primary Key | Id number |
| FacultyID | int | 11 | Foreign Key | Foreign key  Id from  tbl\_faculty\_details |
| SemesterSubjectID | int | 11 | Foreign Key | Foreign key  Id from  tbl\_semester\_subject |
| SemesterCreateddate | date | - | Not null | start of semester |
| SemesterDatedate | date | - | null allowed | end of semester |

6.1.2.12 tbl\_counselor\_allocation

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| CouncelorAllocationID | int | 11 | Primary Key | Id number for allocating counselor students |
| FacultyID | int | 11 | Foreign key | Foreign key  Id from tbl\_faculty |
| StudentID | int | 11 | Foreign key | Foreign key  Id from tbl\_student |
| AllocatedDate | date | - | Not null | Date when information is added |
| DeallocatedFDate | date | - | Not null | Date when counselor is deallocated  I.e counselor no longer is assigned to students |

6.1.2.13 tbl\_class\_schedule

| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| ClassScheduleID | int | 11 | Primary Key | Id number |
| FacultyAllocationID | int | 11 | Foreign key | Foreign key  Id from  tbl\_faculty\_subject\_allocation |
| ClassScheduleDay | varchar | 3 | Not null | Day of the week in three letters like sun,mon,tue,wed,thu,fri,sat |
| LectureFromTime | time | - | Not null | Start time of lectures |
| LectureToTime | time | - | Not null | End time of lectures |
| ClassScheduleStartedDate | date | - | Not null | Start time of schedule |
| ClassScheduleEndedDate | date | - | Null allowed | End time of schedule  Ie ster |

6.1.2.14 tbl\_attendance\_master

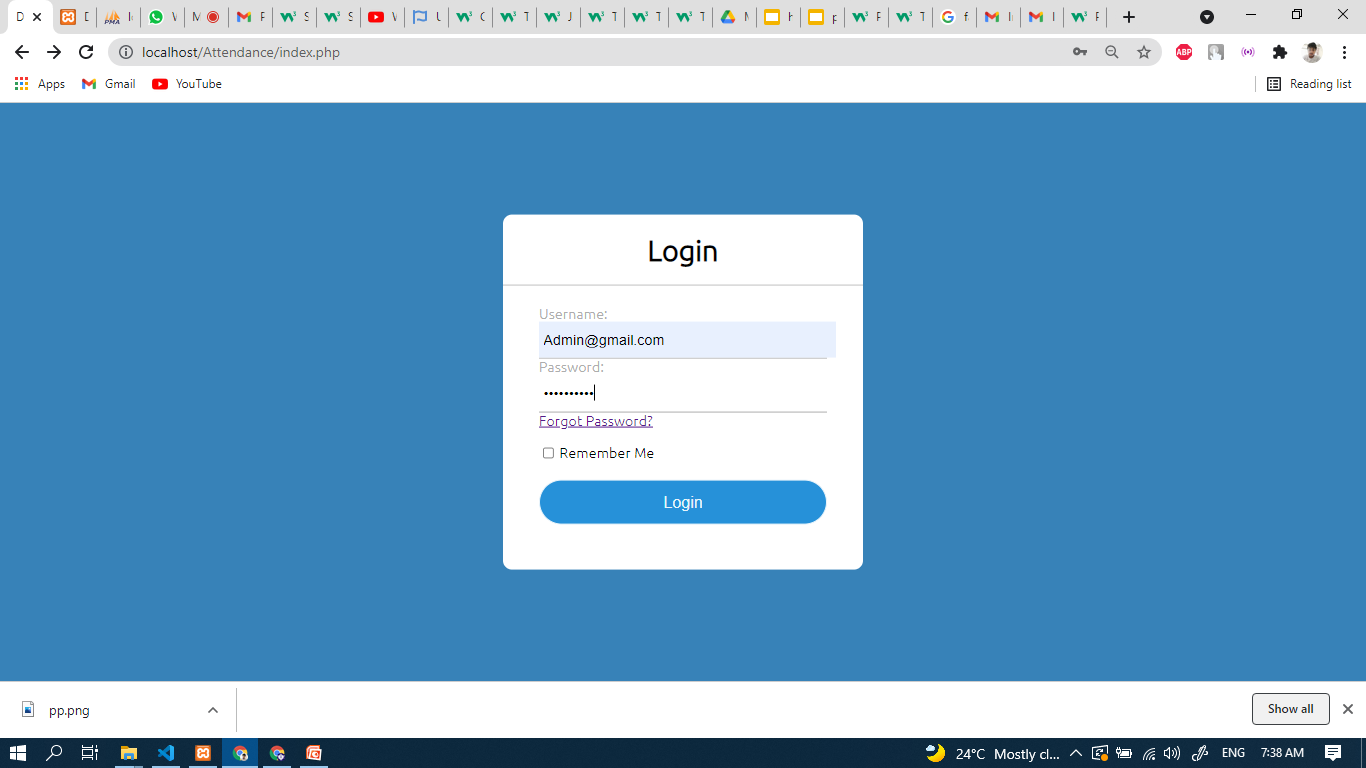
| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| AttendanceMasterID | int | 11 | Primary Key | id number for master attendance table |
| SemesterID | int | 11 | Foreign Key | foreign key id from tbl\_semester |
| FacultyID | int | 11 | Foreign key | Foreign key  Id from  tbl\_faculty |
| AttendanceDate | date | - | Not null | Date of attendance taken |
| FromTime | time | - | not null | lecture start time |
| ToTime | time | - | not null | Lecture End Time |
| ClassScheduleID | int | 11 | Foreign Key | forign key id from tbl\_classsheduel |
| Topic | varchar | 70 | not null | What the lecture contained |

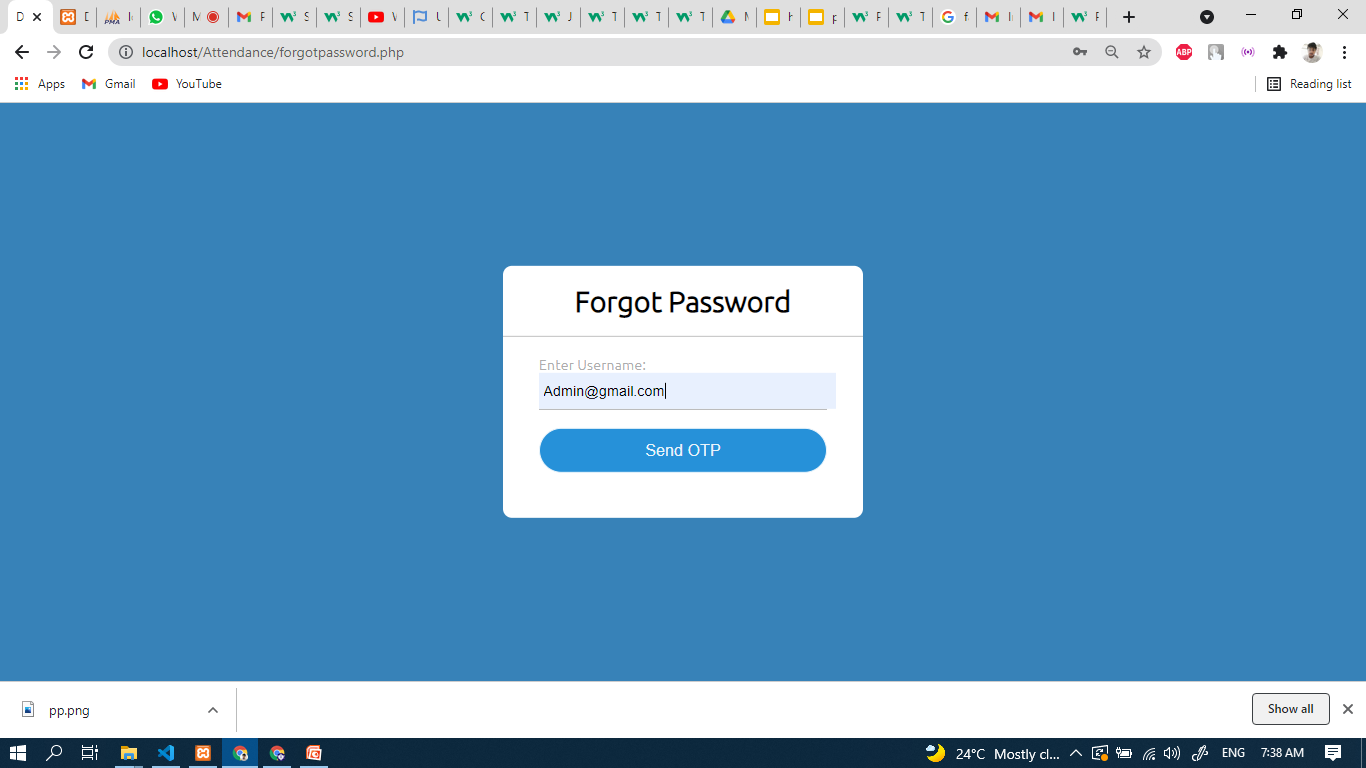
6.1.2.15 tbl\_attendance

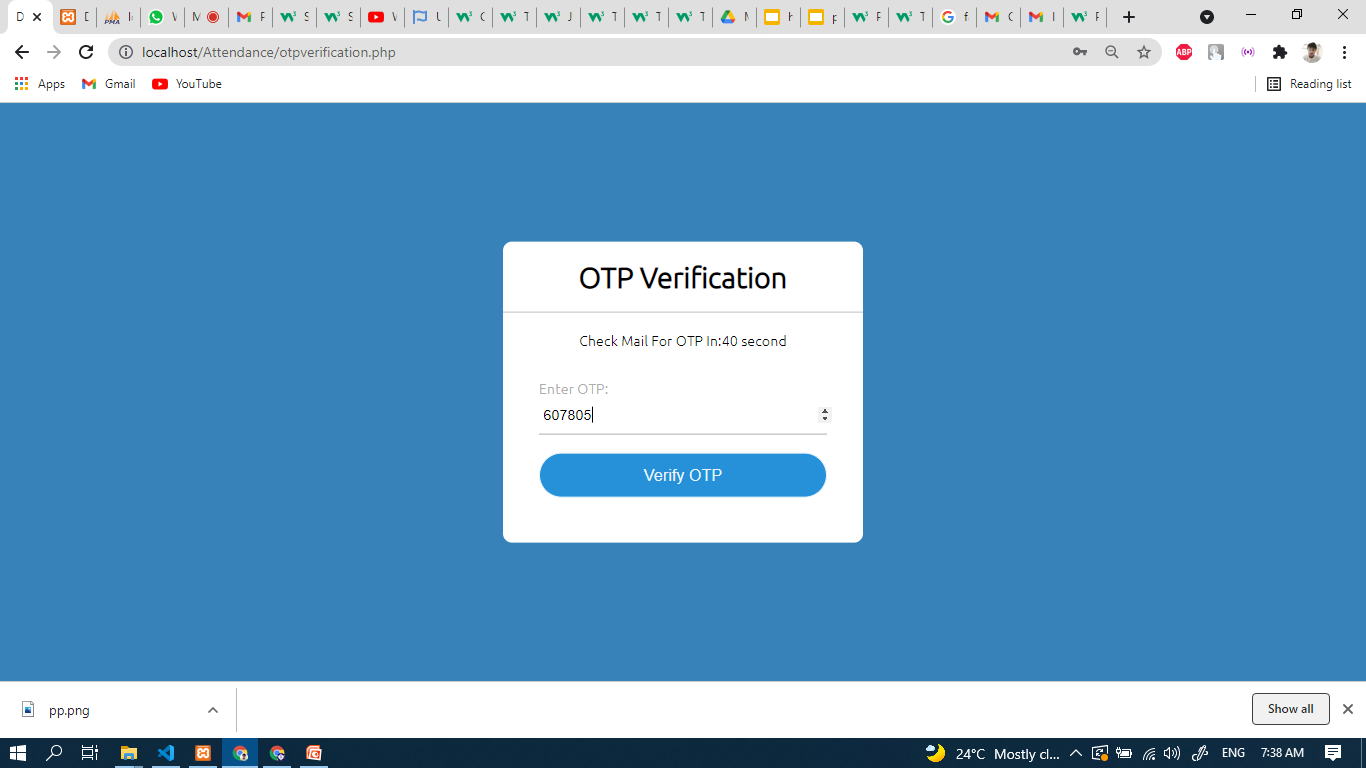
| Field | Type | Size | Constraint | Description |
| --- | --- | --- | --- | --- |
| AttendanceID | int | 11 | Primary Key | Id number of Attendance information |
| StudentID | int | 11 | Foreign key | Foreign key  Id from  From tbl\_student |
| Attendance MasterID | int | 11 | Foreign key | Foreign key  Id from  From tbl\_attendancemaster |
| AttendanceStatus | tinyint | 1 | Not null | Active or inactive |

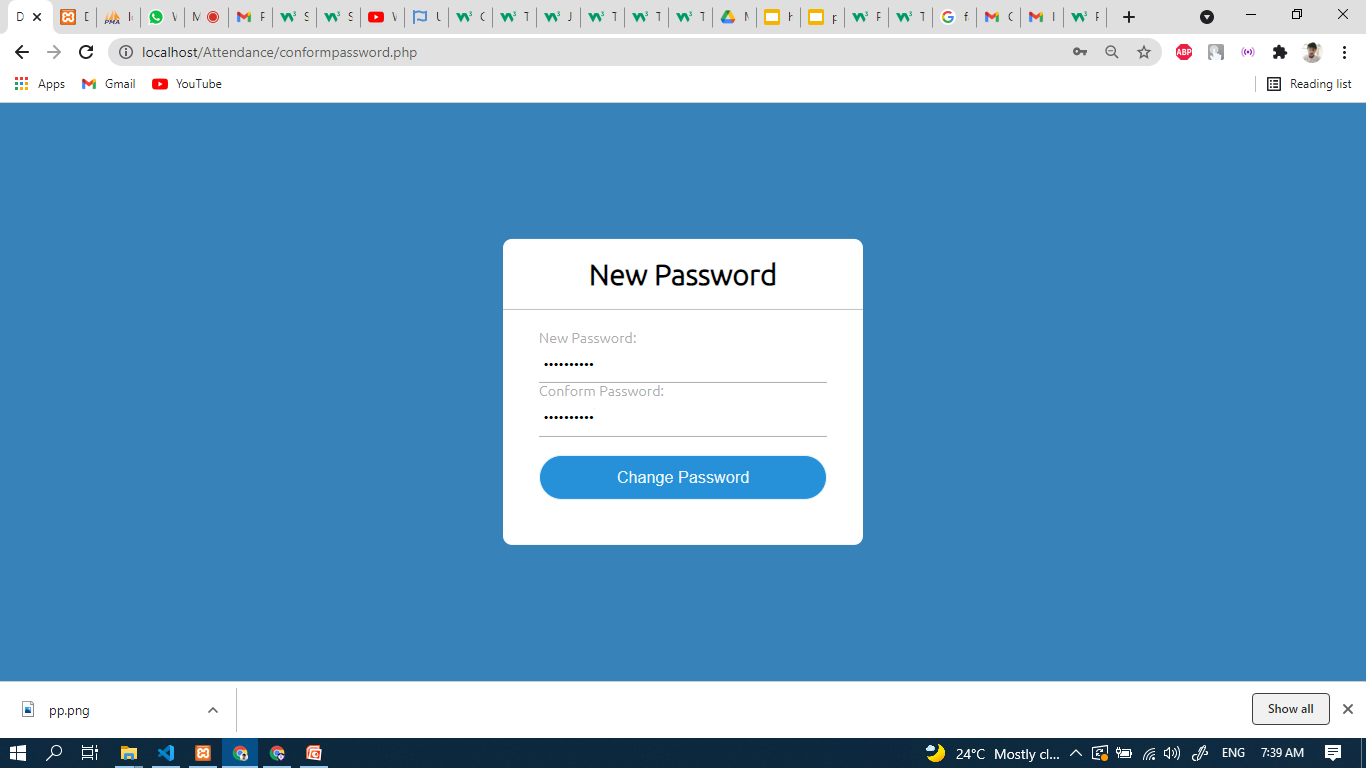
**7: System Implementation**

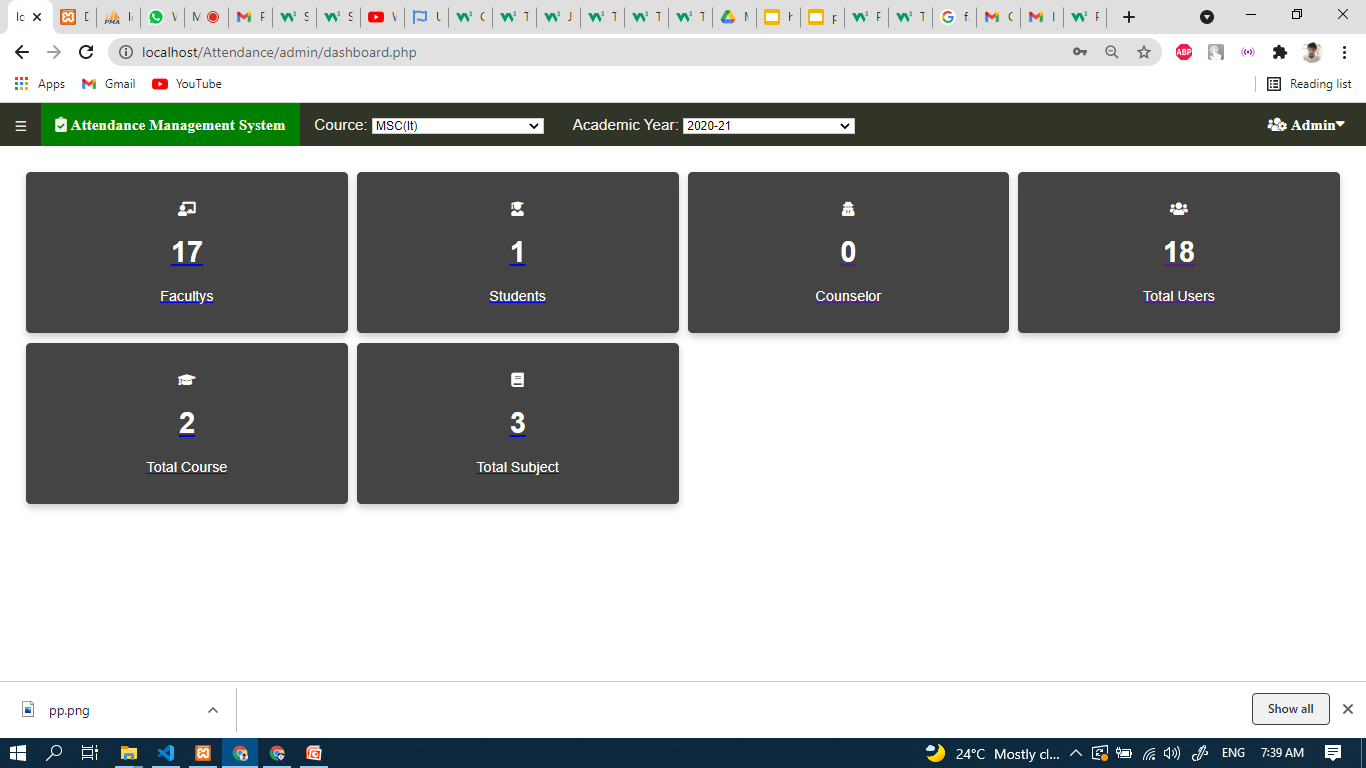
**7.1:Screenshots**

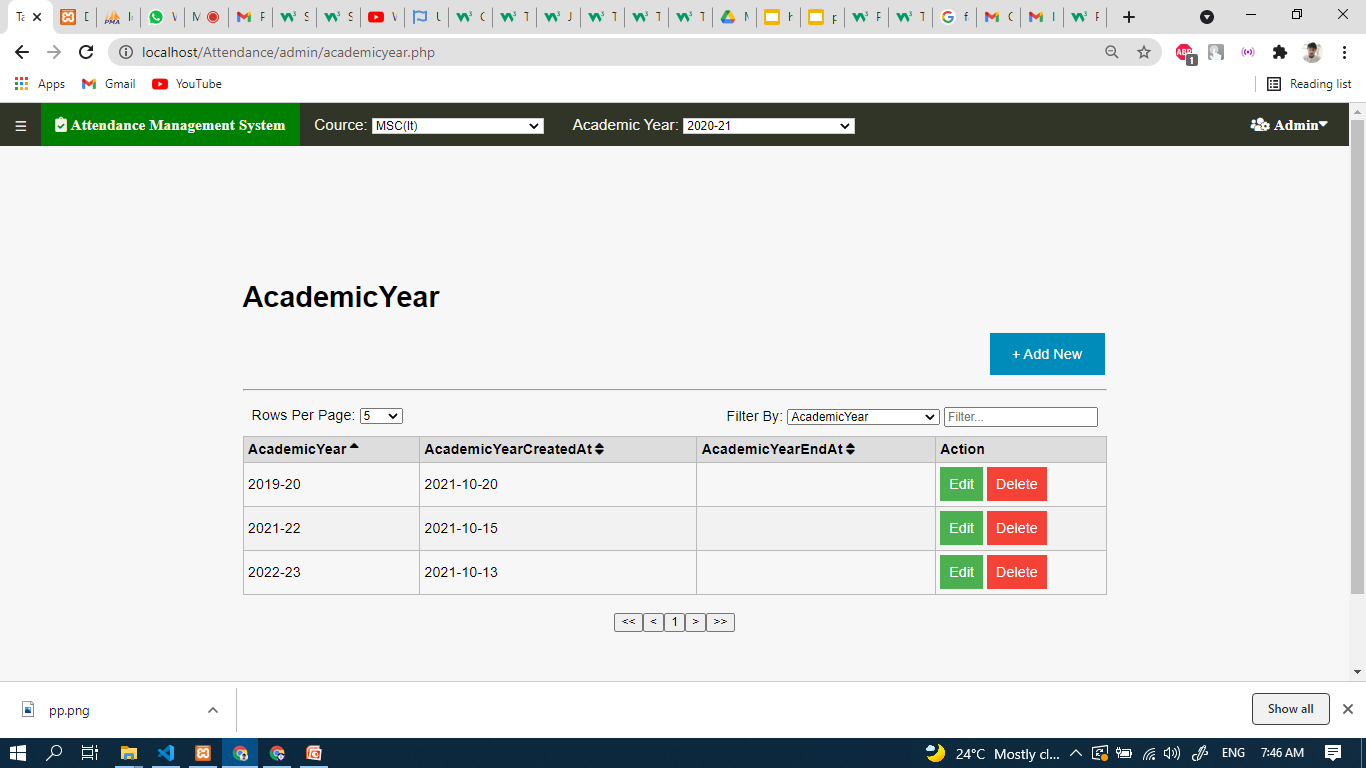
****

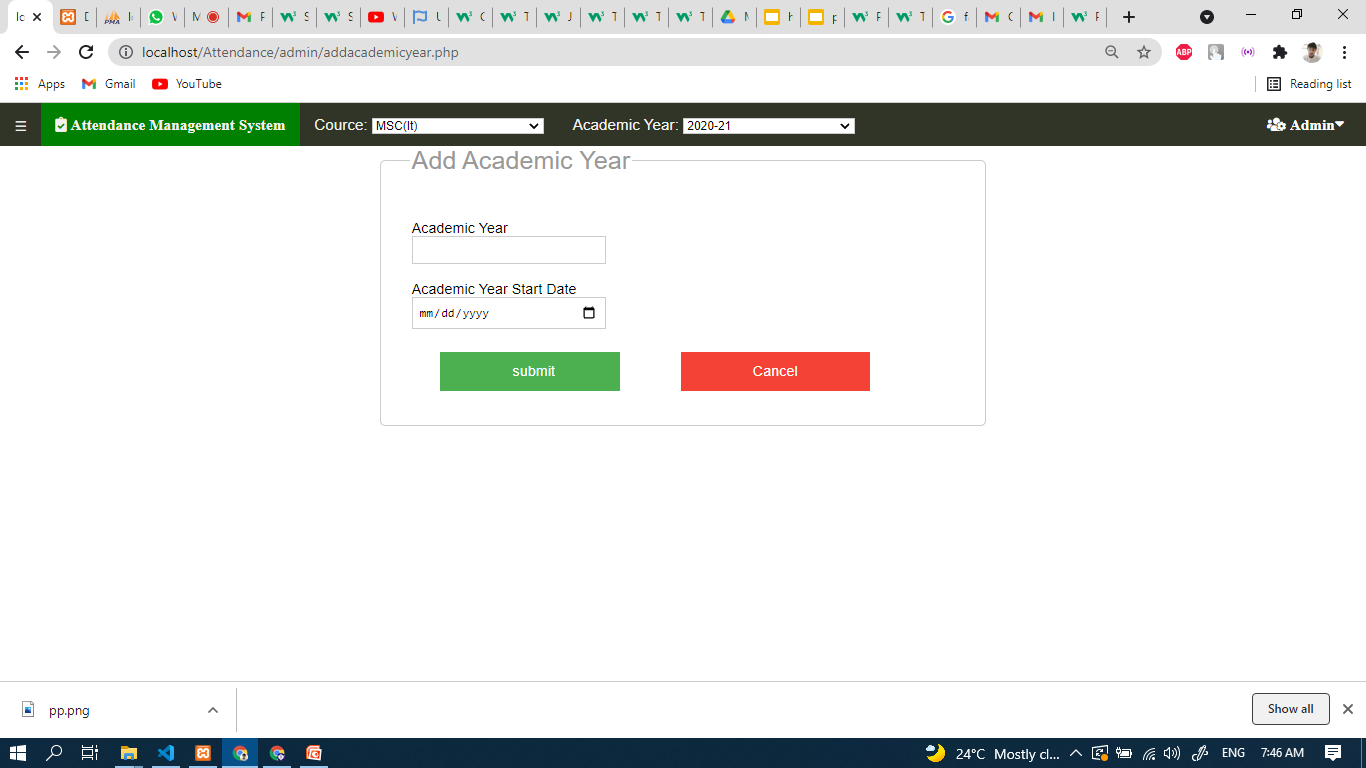
****

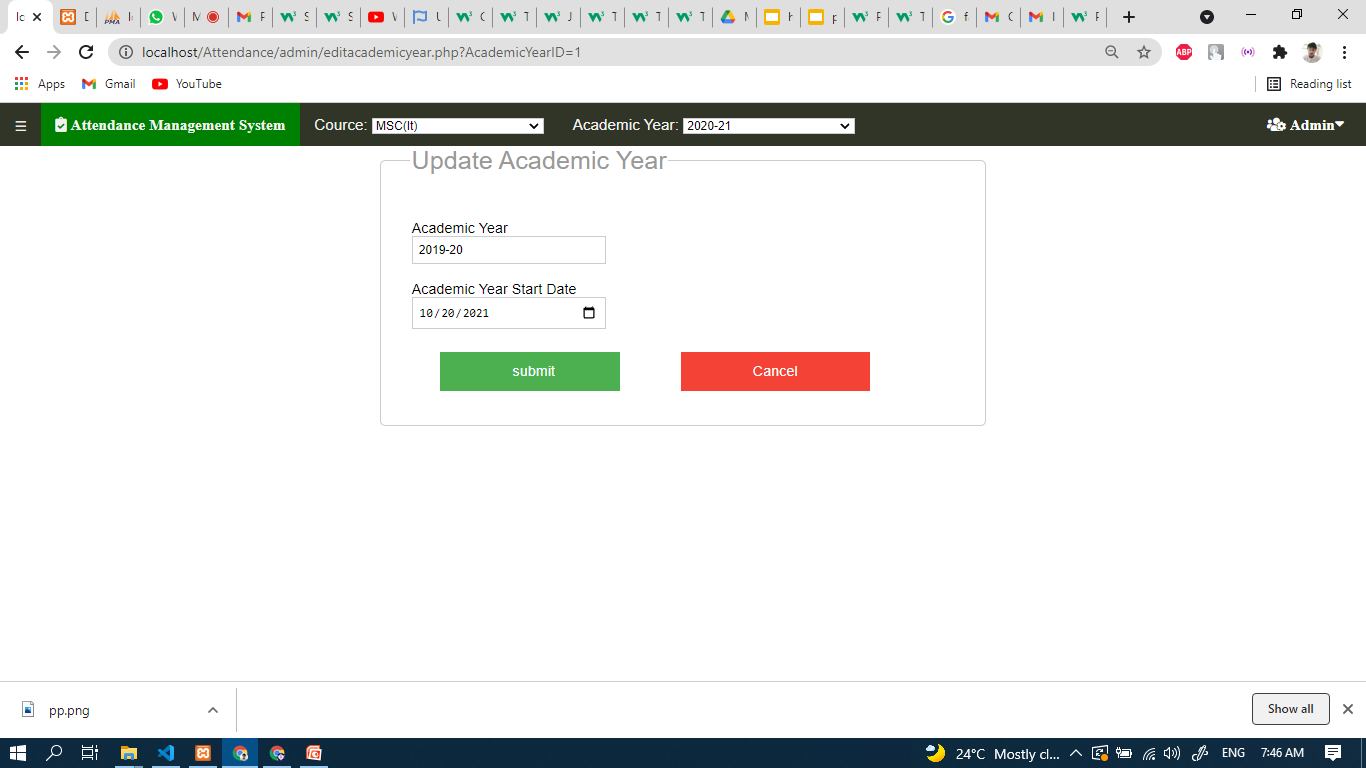
****

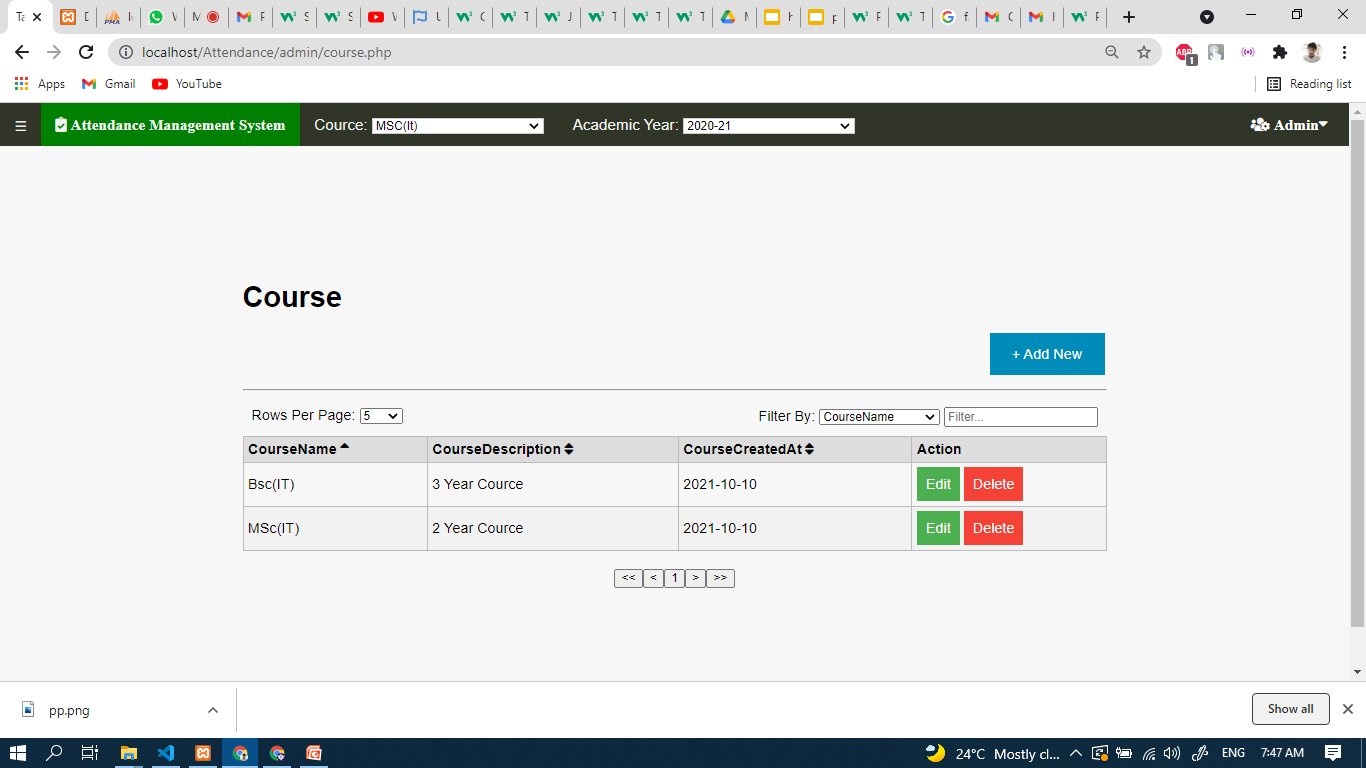
****

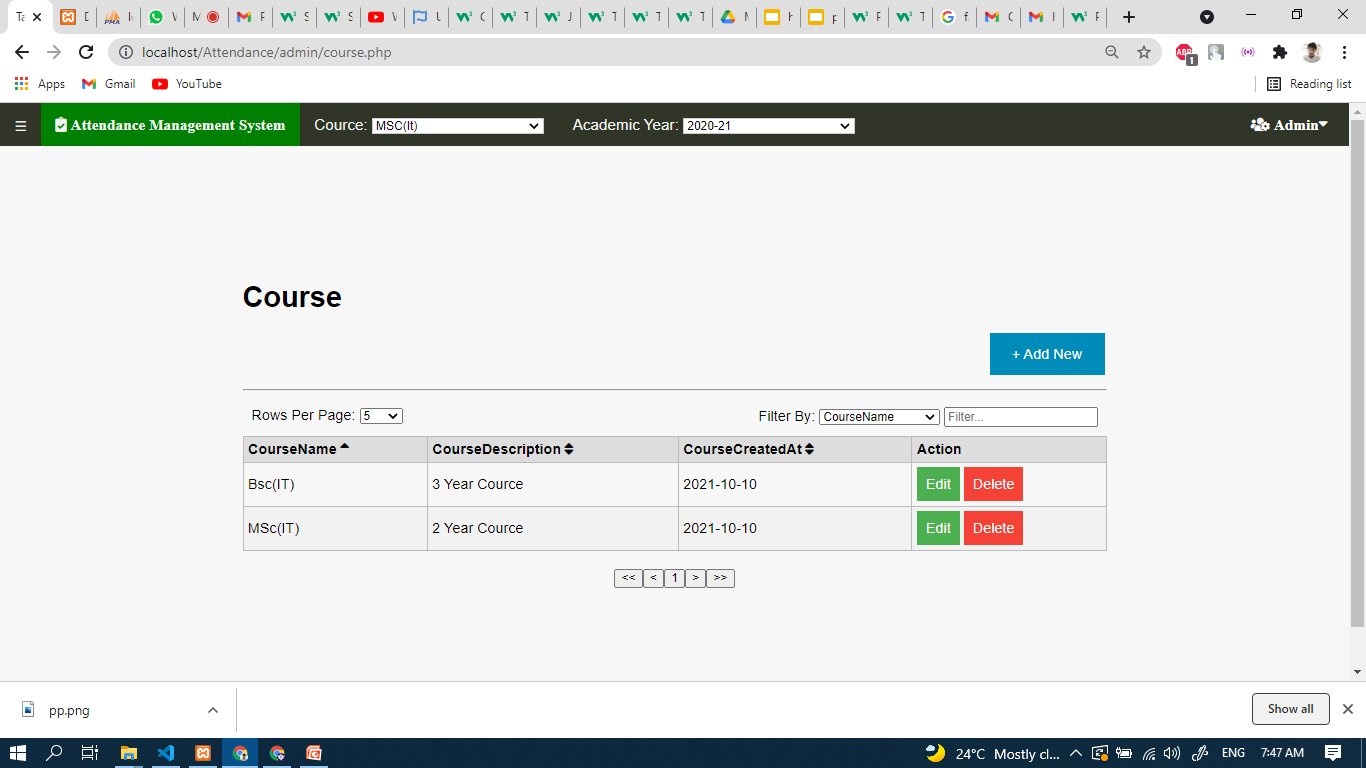
****

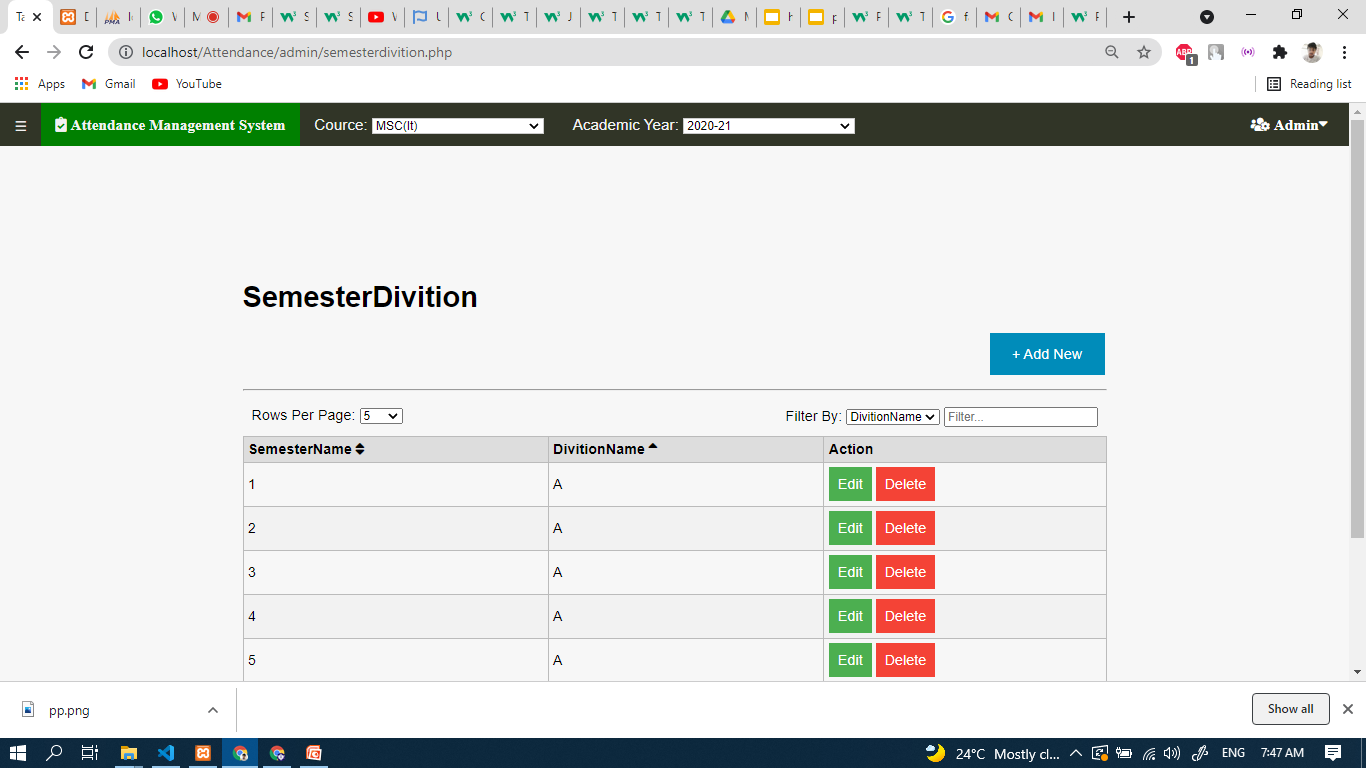
****

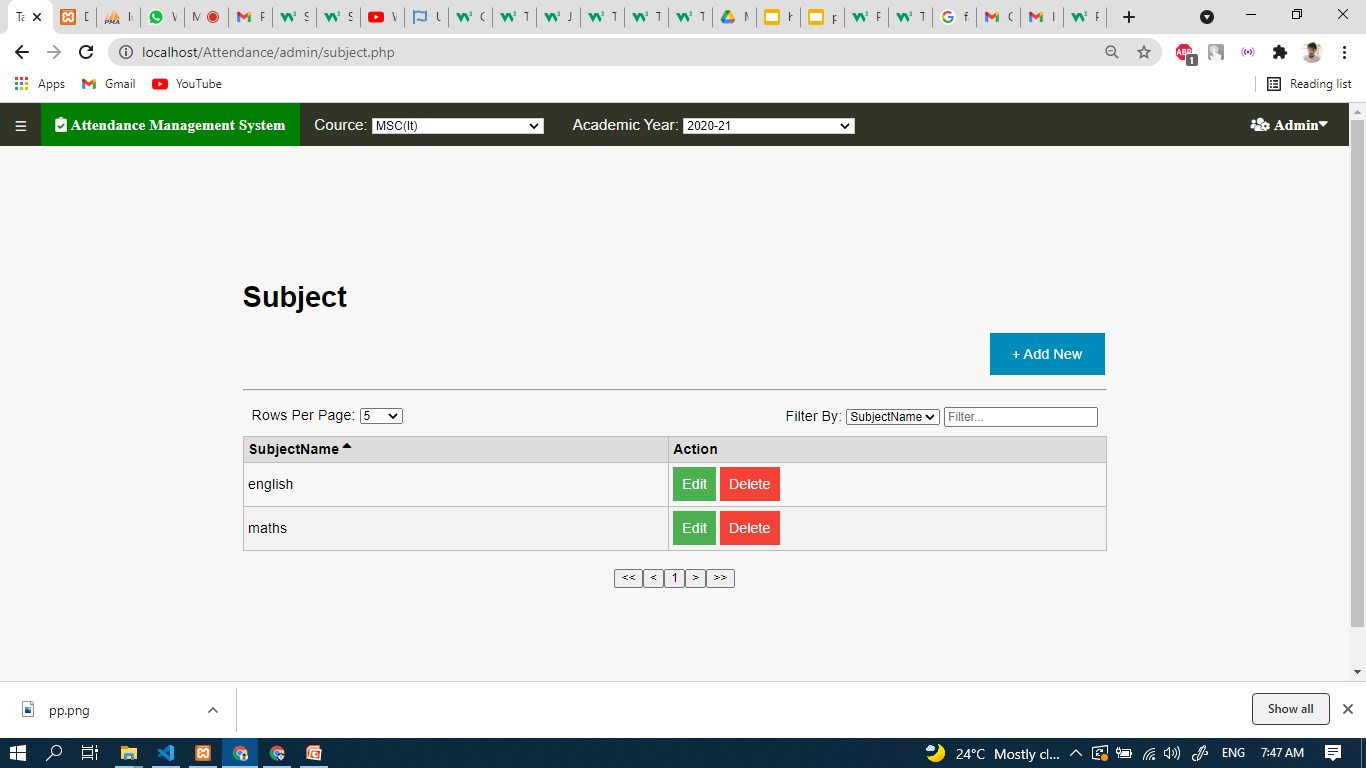
****

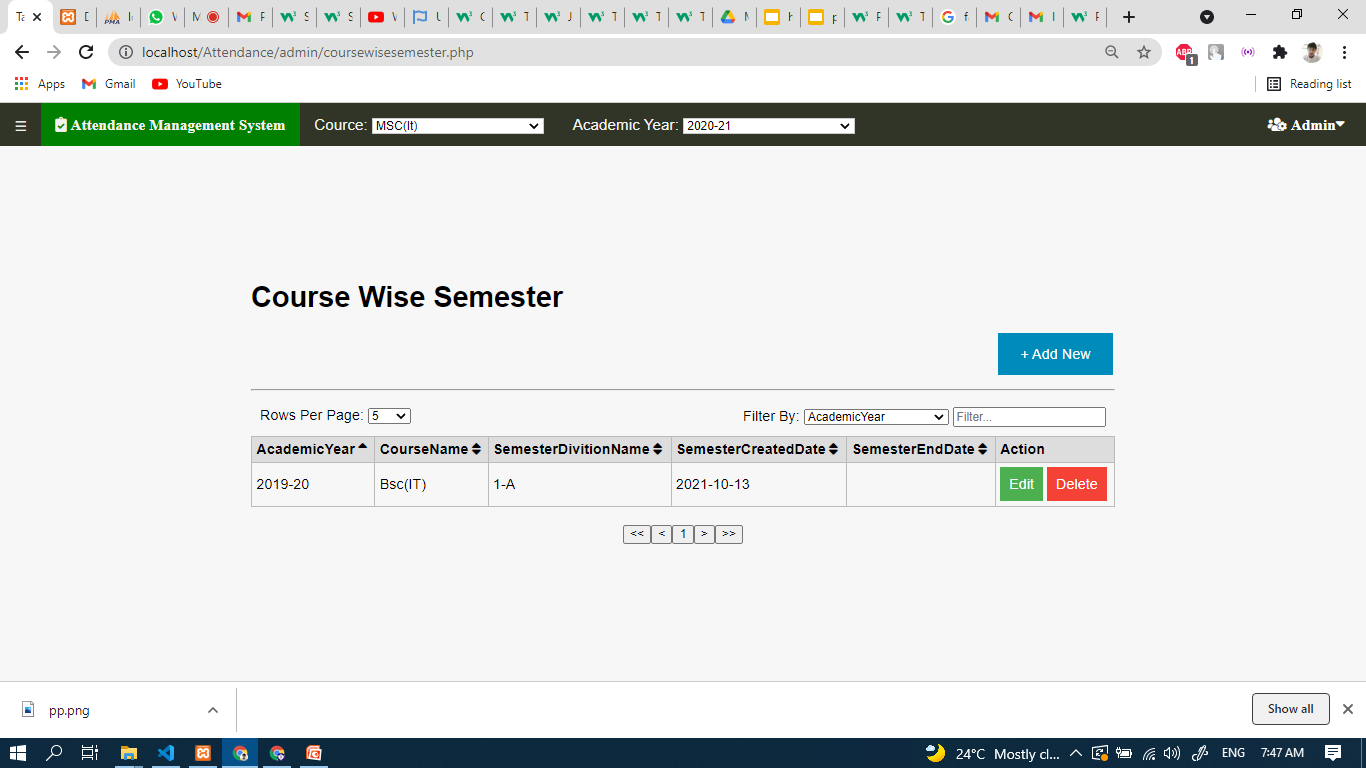
****

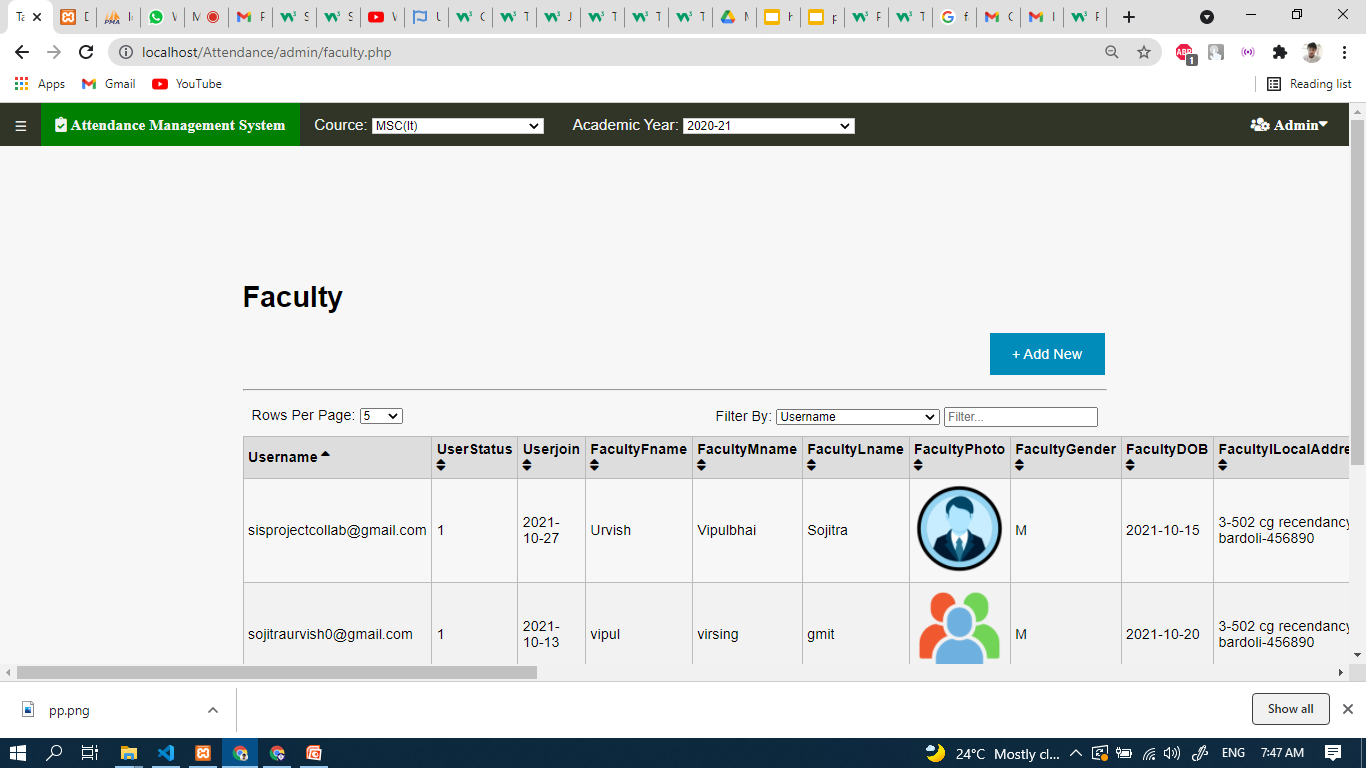
****

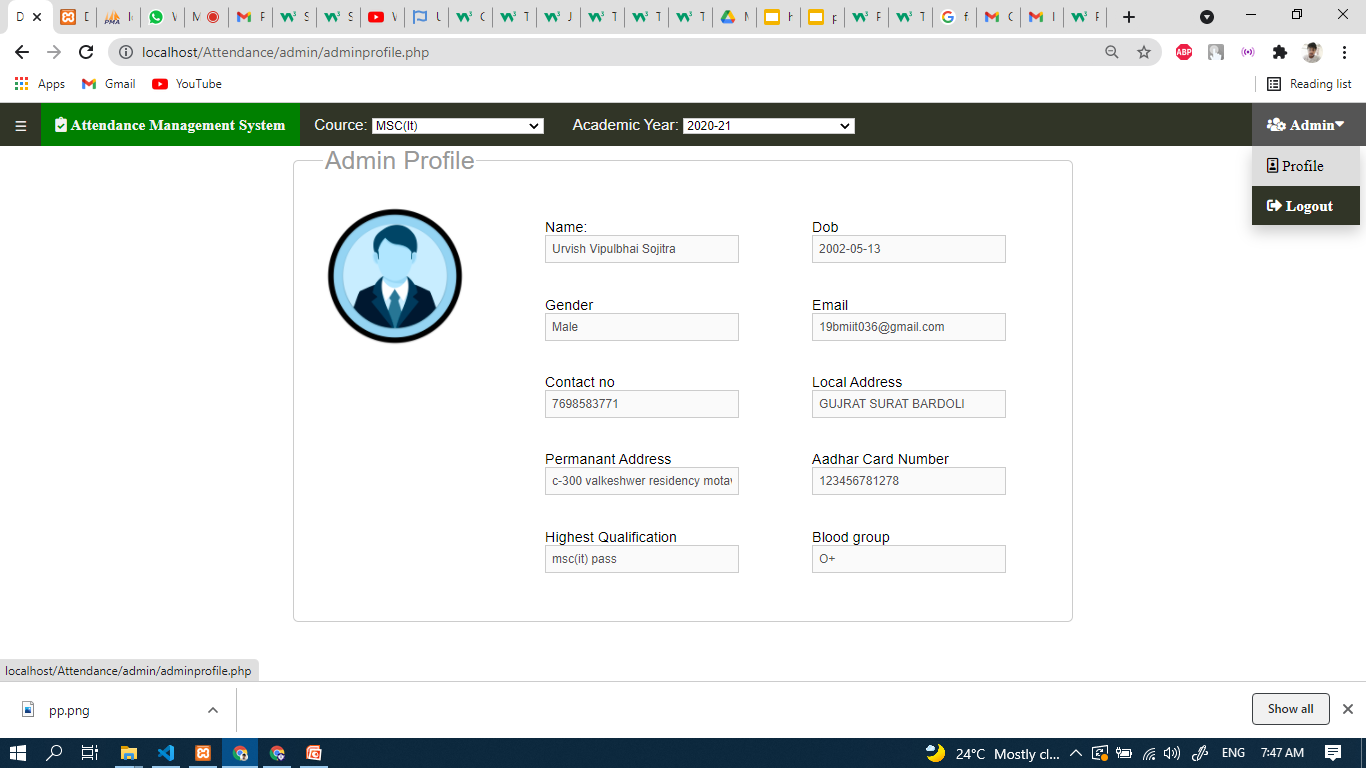
****

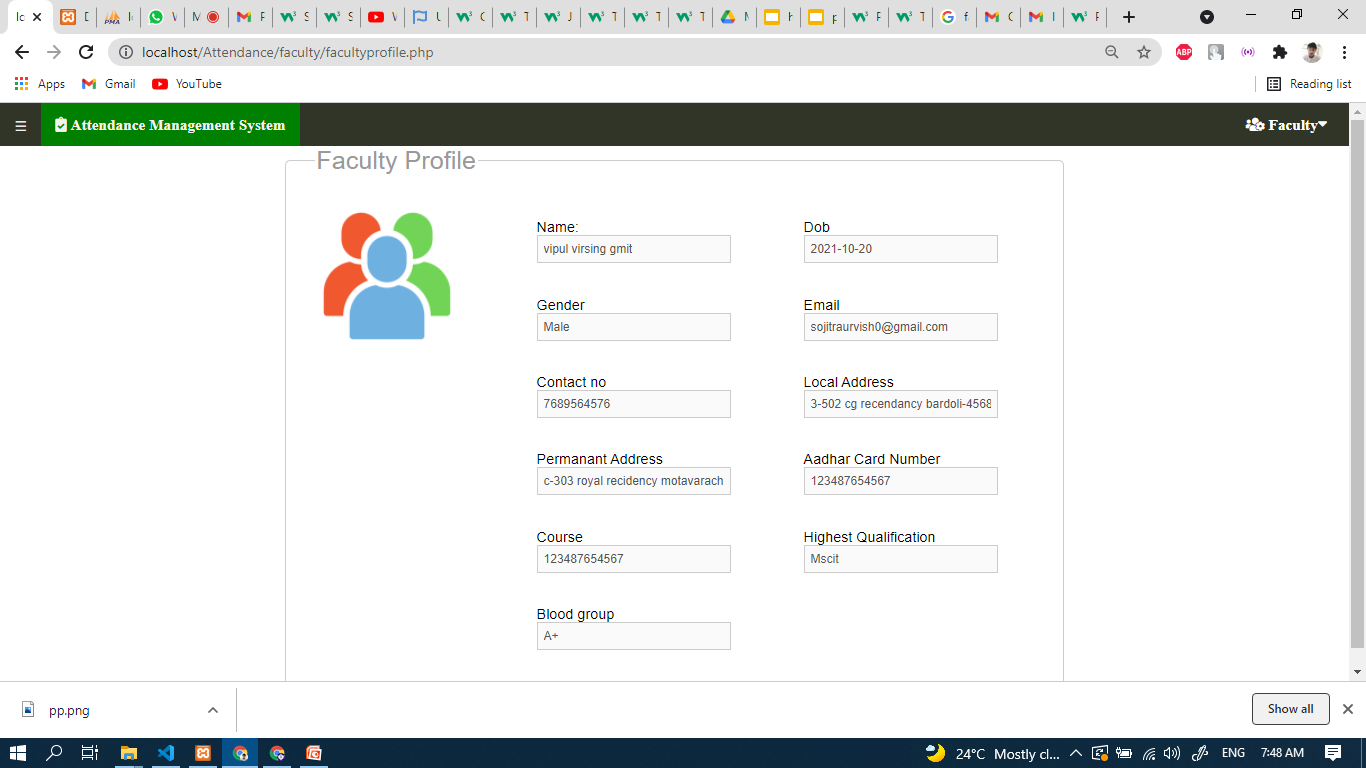
****

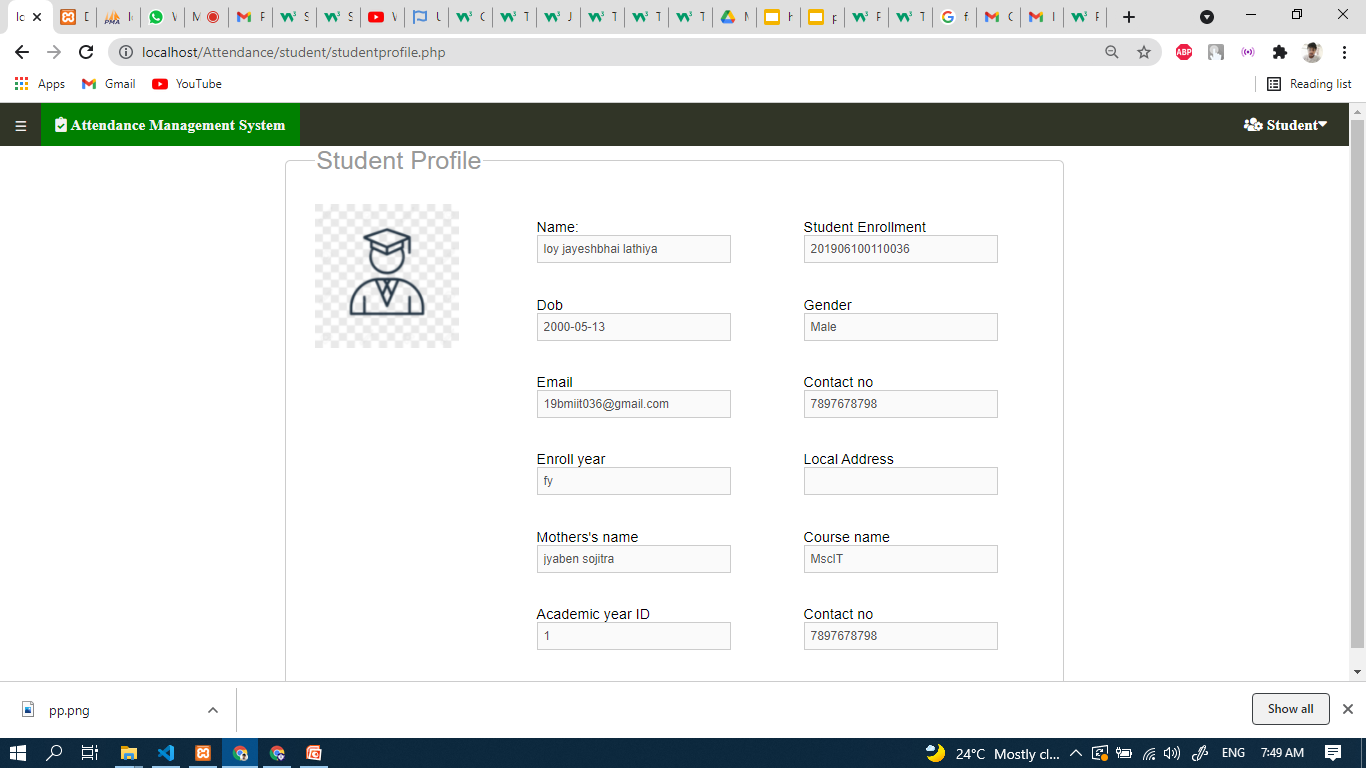
****

****

****

****

****

****

**8:Testing**

**8.1:Test Cases**

**8.1.1 Login**

| **Test Case Id** | **Test Case Description** | **Input-1** | **Input-2** | **Input-3** | **Input-4** | **Input-5** | **Expected Result** | **Remark (if any)** |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Username | Password | OTP | New Password | Retype new Password |
| **TC1** | Attempt to login | Valid input | Valid input | - | - | - | Login Success | OTP,NEw password and Retype password are for forget password |
| **TC2** | Attempt to login | Invalid  Input | Valid/Invalid input | - | - | - | Login Fail | Both username and password need to be valid |
| **TC3** | Attempt to login | Valid/Invalid Input | Invalid Input | - | - | - | Login Fail |  |
| **TC3** | Forget Password | Valid  Input | - | Valid Input | Valid Input | Valid Input | Password  changed  successfully |  |
| **TC4** | Forget Password | Invalid  Input | - | Valid/Invalid  Input | Valid/Invalid Input | Valid/Invalid Input | Password change failed | Correct username must be added to change password |
| **TC5** | Forget Password | Valid  Input | - | Invalid  Input | Valid/Invalid  Input | Valid/Invalid  Input | Password change failed | Correct OTP must be added for password change |
| **TC6** | Forget Password | Valid  Input | - | Valid/Invalid  Input | Invalid  Input | Valid/Invalid  Input | Password change failed | New password can't be the same as old password |
| **TC7** | Forget Password | Valid  Inpu | - | Valid/Invalid  Input | Valid/Invalid  Input | nvalid  Input | Password change failed | Retyped password  must be the same as new password |

**8.1.1 Change password**

| **Test Case Id** | **Test Case Description** | **Input-1** | **Input-2** | **Expected Result** | **Remark (if any)** |
| --- | --- | --- | --- | --- | --- |
| old password | new password |
| **TC1** | Attempting to change password | valid input | valid input | password changed |  |
| **TC2** | Attempting to change password | Invalid Input | Valid/Invalid input | Password change failed | User must know their old password to change it |
| **TC3** | Attempting to change password | Valid Input | Invalid input | Password change failed | Old password can't be as same as the new password |

**9:Future Enhancement**

System is only available for one department and in the future it will be available for multiple departments.System will also be expanded to have multiple other features like Leave request and will be able to accommodate elective subjects as well.The role of Director will also be added.

**Conclusion**

Attendance management System is the right solution for any institution like university or college which requires taking attendance.This will help reduce human effort and reduce human errors as well making attendance an easy and efficient task.

**Bibliography**

1.[www.w3school.com](http://www.w3school.com)

2.The complete Reference PHP by Steven Holzner

3.www.php.net

4.www.stackoverflow.com