



Dharmsinh Desai University, Nadiad
Faculty of Technology
Department of Computer Engineering

B. Tech. CE Semester – IV
Subject : Software Engineering Practices

A PROJECT REPORT ON
Project title: Student Information Management System

By
CHAVDA DHARMI S. (CE-018) (19CEUBS127)
CHOKSI KANAL P. (CE-020) (19CEUOG002)

Guided by : Prof. Pinkal Chauhan

&

Prof. Ankit P. Vaishnav

Contents

1. Abstract.....	4
2. Introduction	5
3. Software Requirement Specifications.....	7
4. Design Documents.....	10
5. Implementation Details.....	22
6. Testing.....	25
7. Workflow/Layouts.....	26
8. Conclusion.....	30
9. Limitation and Future Extension.....	31
10. Bibliography.....	32

1. Abstract

Student Information Management System can be used by education institutes to maintain the records of students easily. Achieving this objective is difficult using a manual system as the information is scattered, can be redundant and collecting relevant information may be very time consuming. All these problems are solved using this project.

Without a Student information Management System managing and maintaining the details of the student is a tedious job for any organization. the system will store all the details of the students including their background information, educational qualifications, personal details and all the information related to their resume . due to this many reasons we want to create a system which can handle student information so that these difficulties can be decrease.

Objectives:

- Online registration of admins
- Maintenance of student records

Users Views:

- Administrator
- Student

2.Introduction

1.1 Breif Introduction :

The objective of Student's information Management System is to allow the users of any organization to edit and find out the personal details of a student and allows the student to keep up to date his profile .It'll also facilitate keeping all the records of students, such as their id, name, mailing address, phone number, DOB etc. So all the information about an student will be available in a few seconds. Overall, it'll make Student's Information Management an easier job for the administrator and the student of any organization. The main purpose of this document is to illustrate the requirements of the project Student information System and is intended to help any organization to maintain and manage its student's personal data.

1.2 Tools / Technologies Used :

Technologies :

Front end :

- Html
- Css
- Javascript
- Bootstrap

Web designing language:

- PHP

Back end :

- RDBMS – MySQL

Tools :

- Sublime Text Editor
- Git
- Xampp server
- FireFox Browser
- WonderShare Edrawmax IDE (for design document)

3. Software Requirement Specifications

Student's Information Management System

Admin:

R.1 Manage students

Description: Admin manage studente's details here

R.1.1 Add student

Description: Admin can add student here

Input: details of students

Output: “confirmation message” (student added successfully)

R.1.2 Remove/Delete student

Description: Admin can remove student here

Input: student's roll no.

Output: “confirmation message” (student deleted successfully)

R.1.3 Edit Student

Description: Admin can edit student here

Input: student's roll no.

Output: "confirmation message" (student edited successfully)

R.1.4 Search Student

Description: Admin can search student's detail here

Input: student's roll no.

Output: Display Details (of student)

R.2 Manage Teacher

Description: Admin manage teachers here

R.2.1 View Teachers

Description: Admin can view teachers here

Input: "view option"

Output: Display Details (of teacher)

Students:

R.1 Login

Description: student can login here by provided id, Password

Input: Input Credential

Output: Home page

R.2 Manage personal details

Description: student can manage and show their personal details

R .2.1 View details

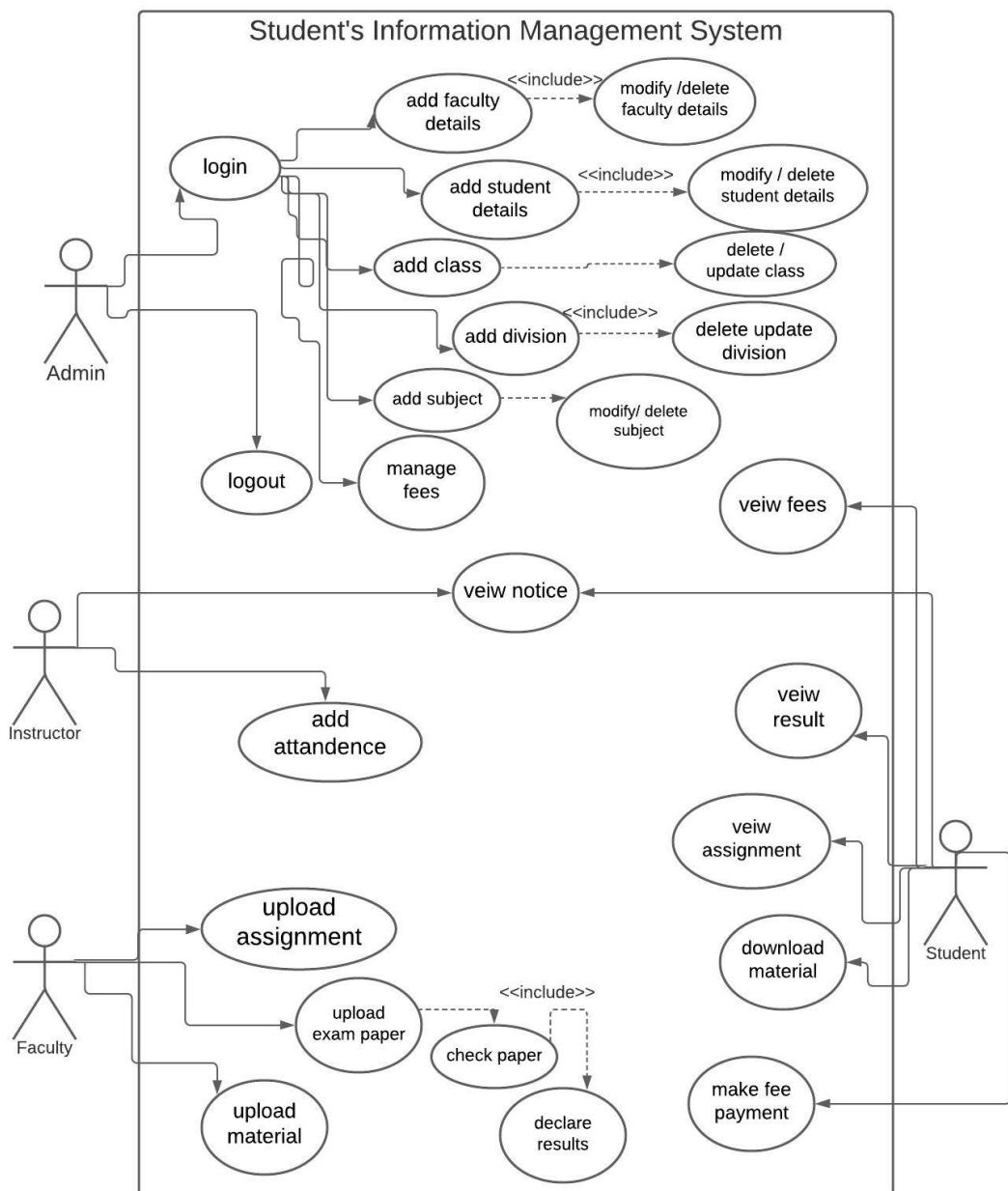
Description: students can view their detail here

Input: “view option”

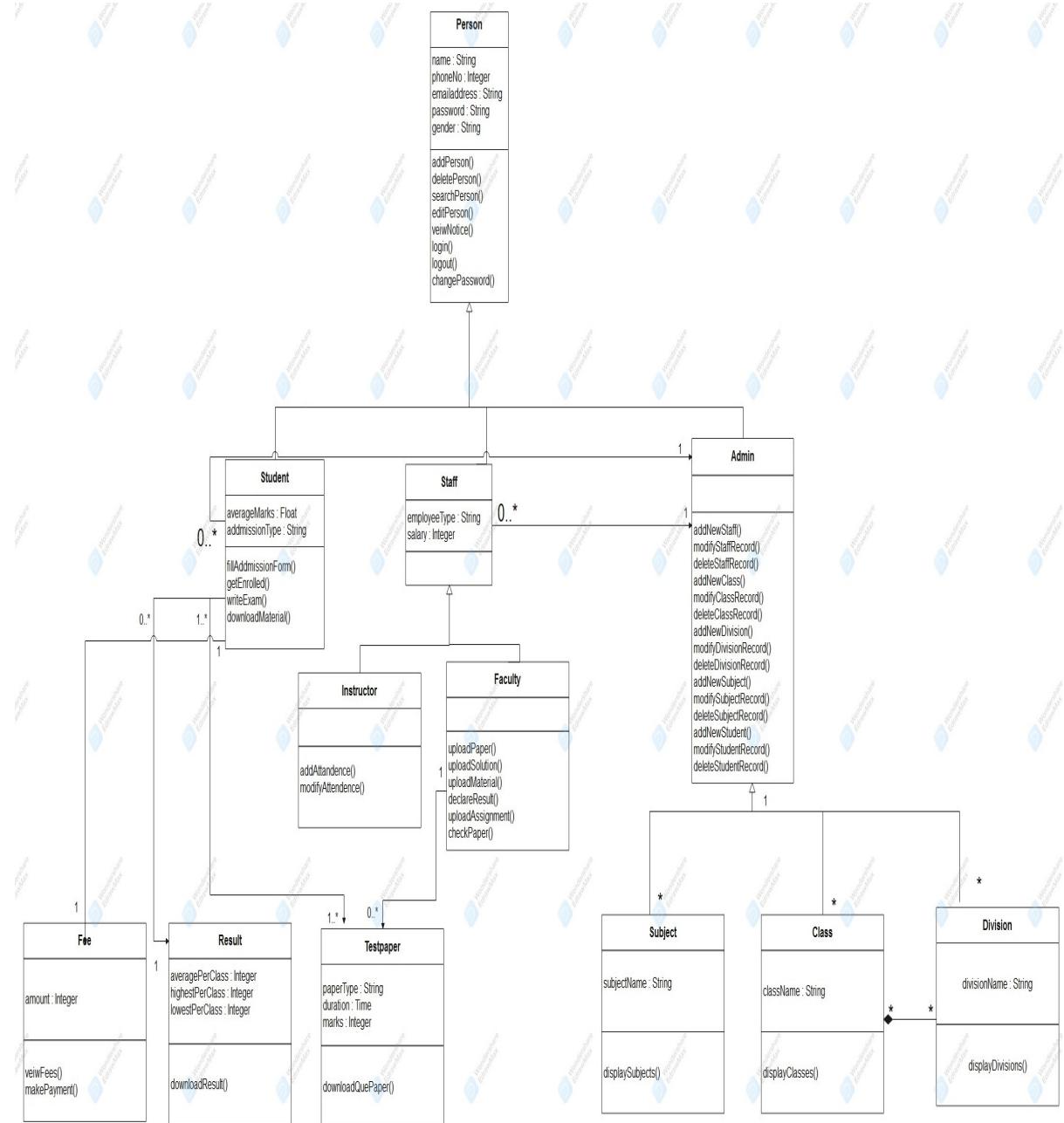
Output: Display Details

4.Design Documents

1. Use case Diagram :

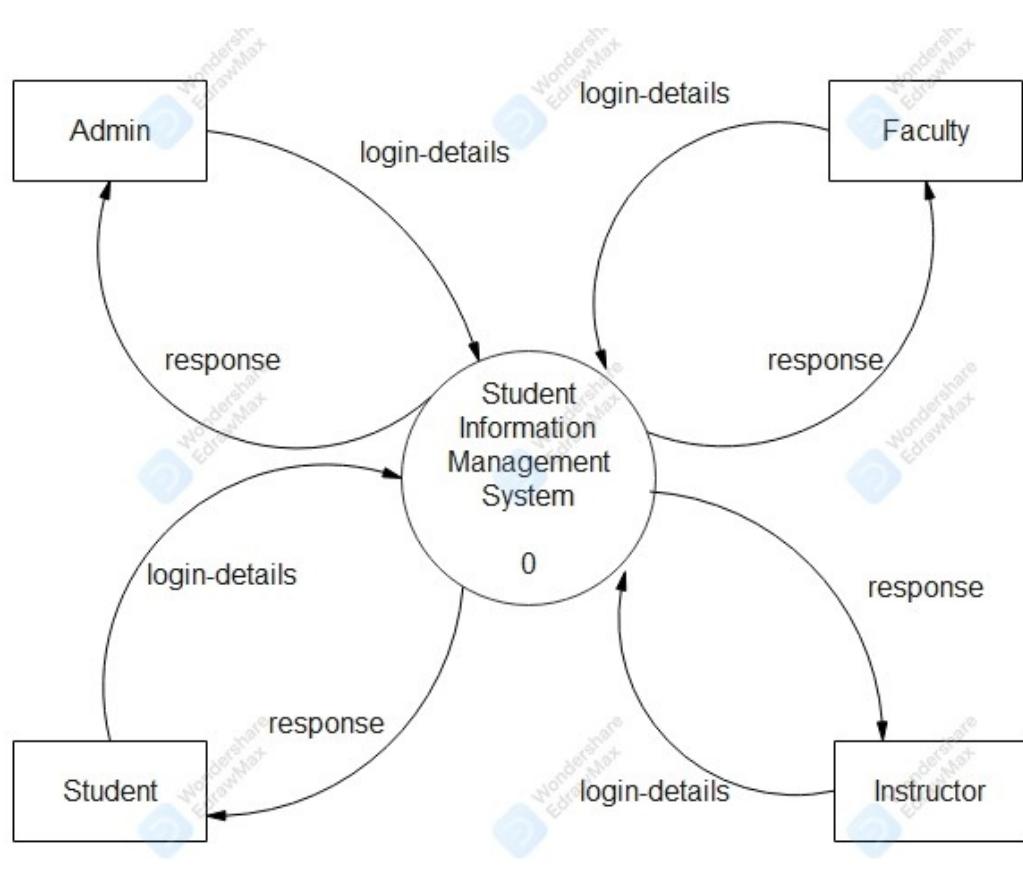


2. Class Diagram :

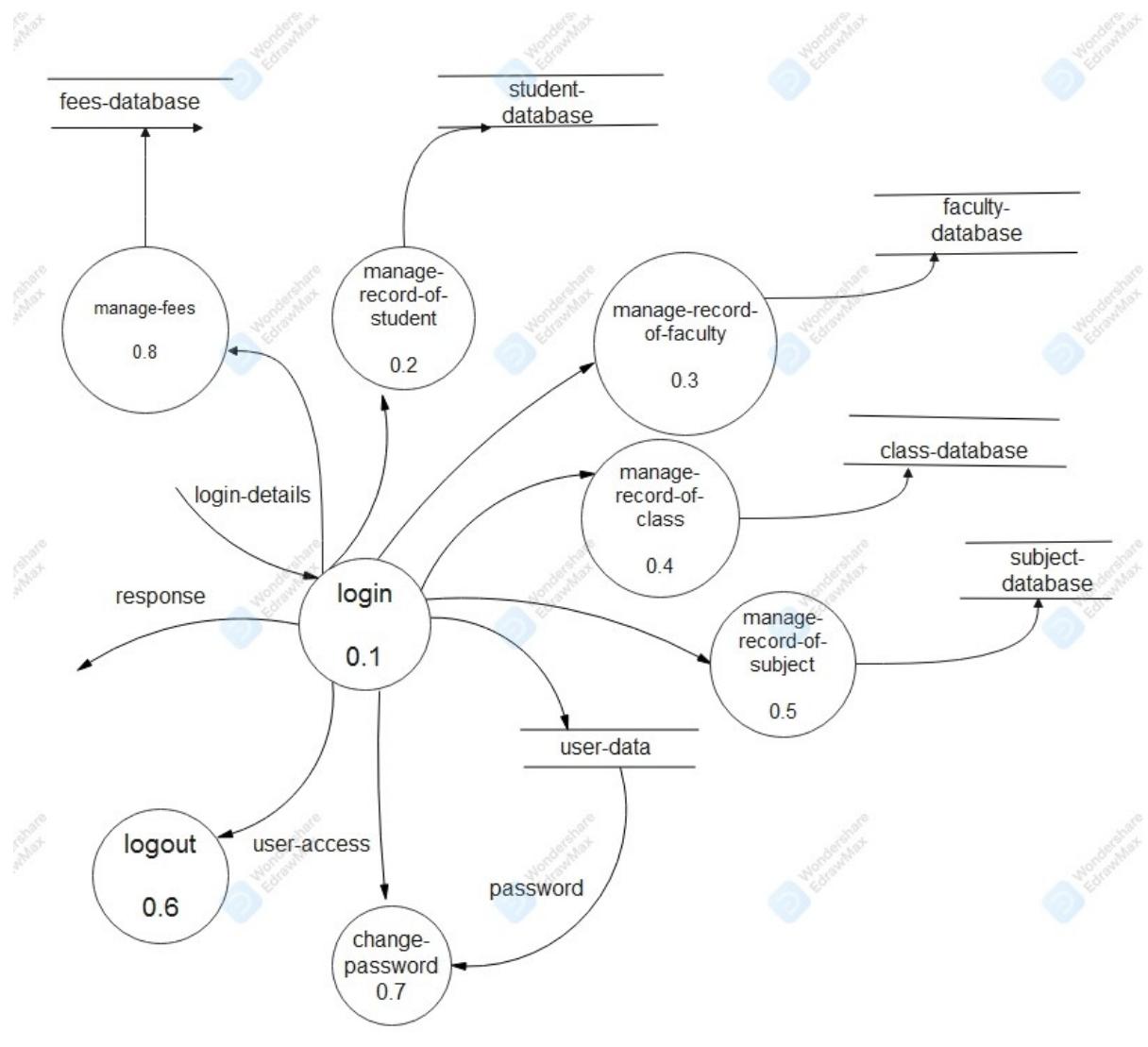


Structure-analysis:

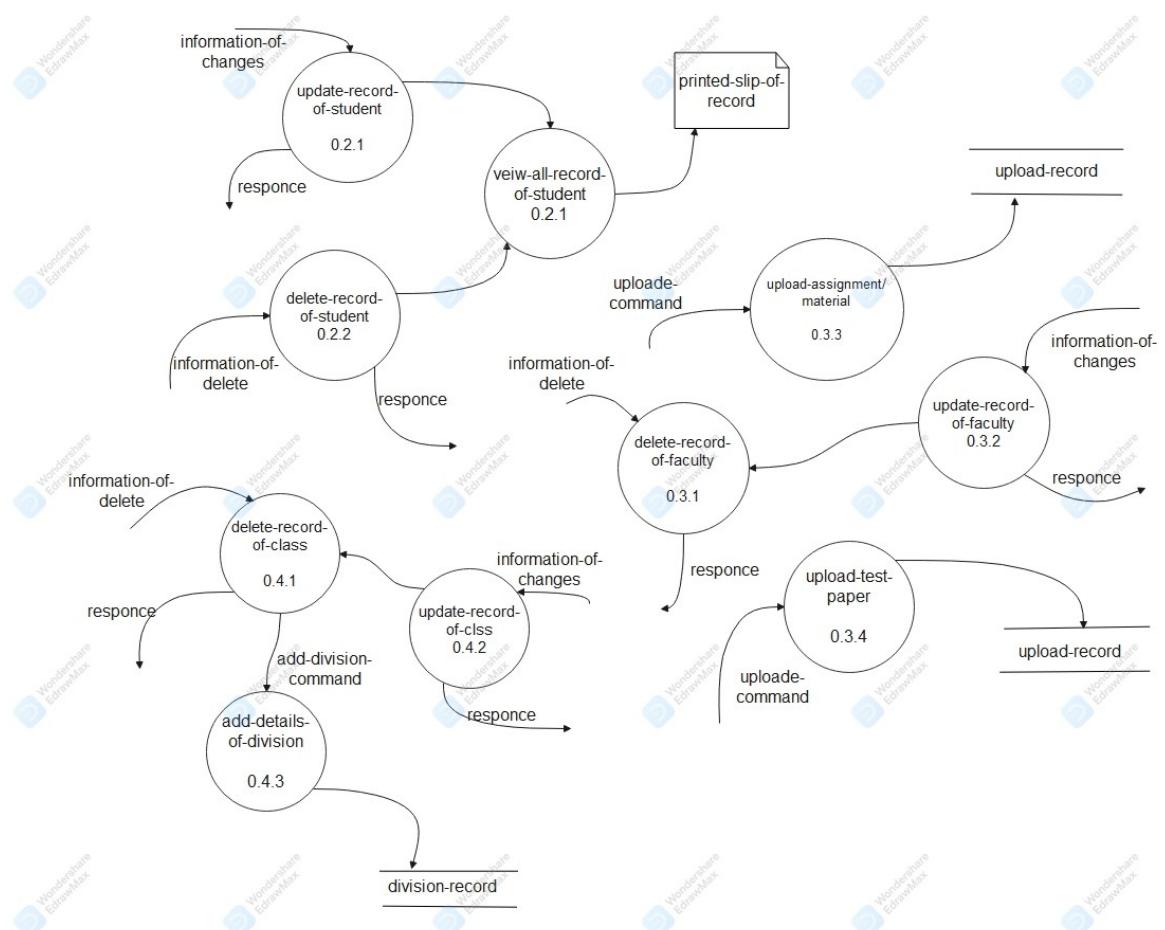
3.Data model :



Level 0 (context-diagram)



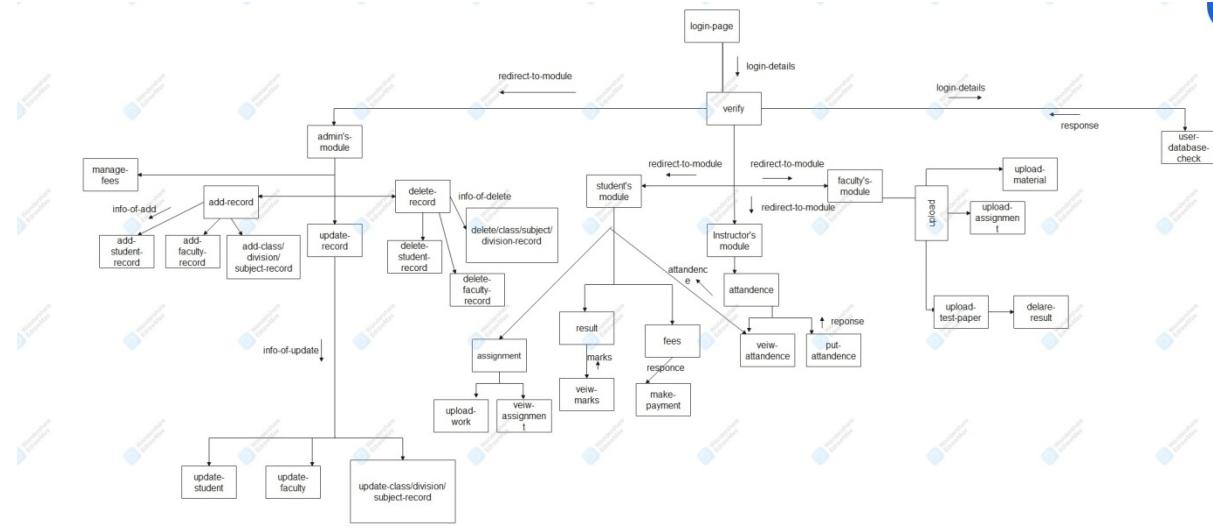
Level 1 diagram



Level 2 diagram

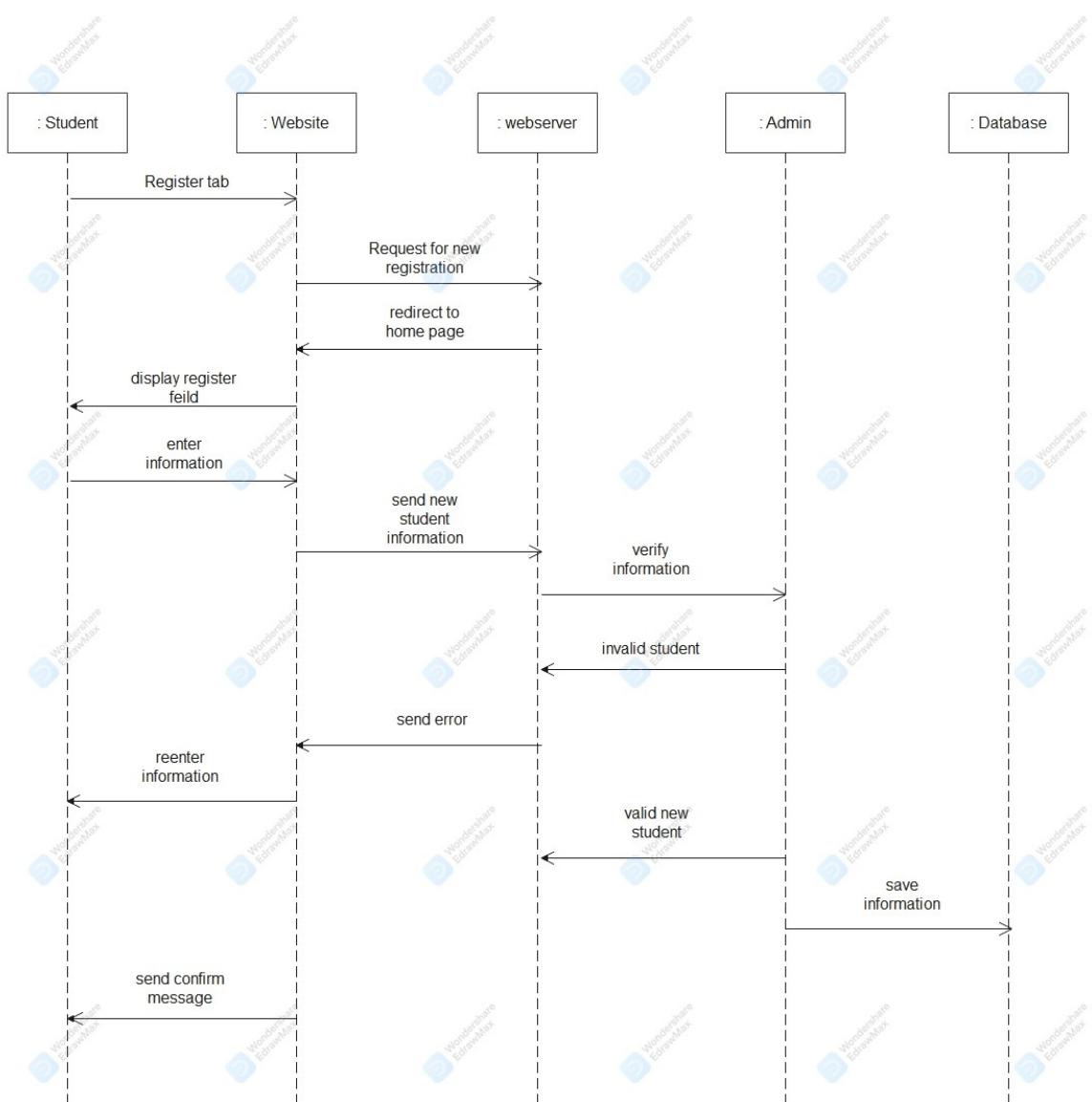
Structure design:

4.structure chart

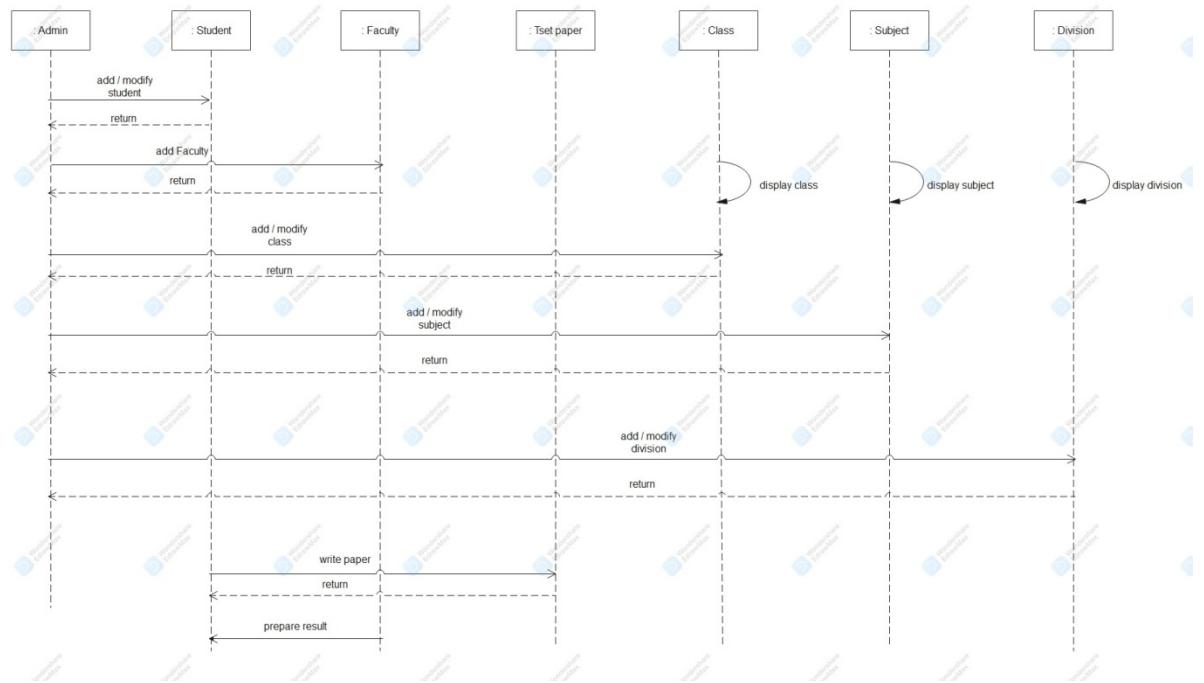


5. Sequence diagram:

1)

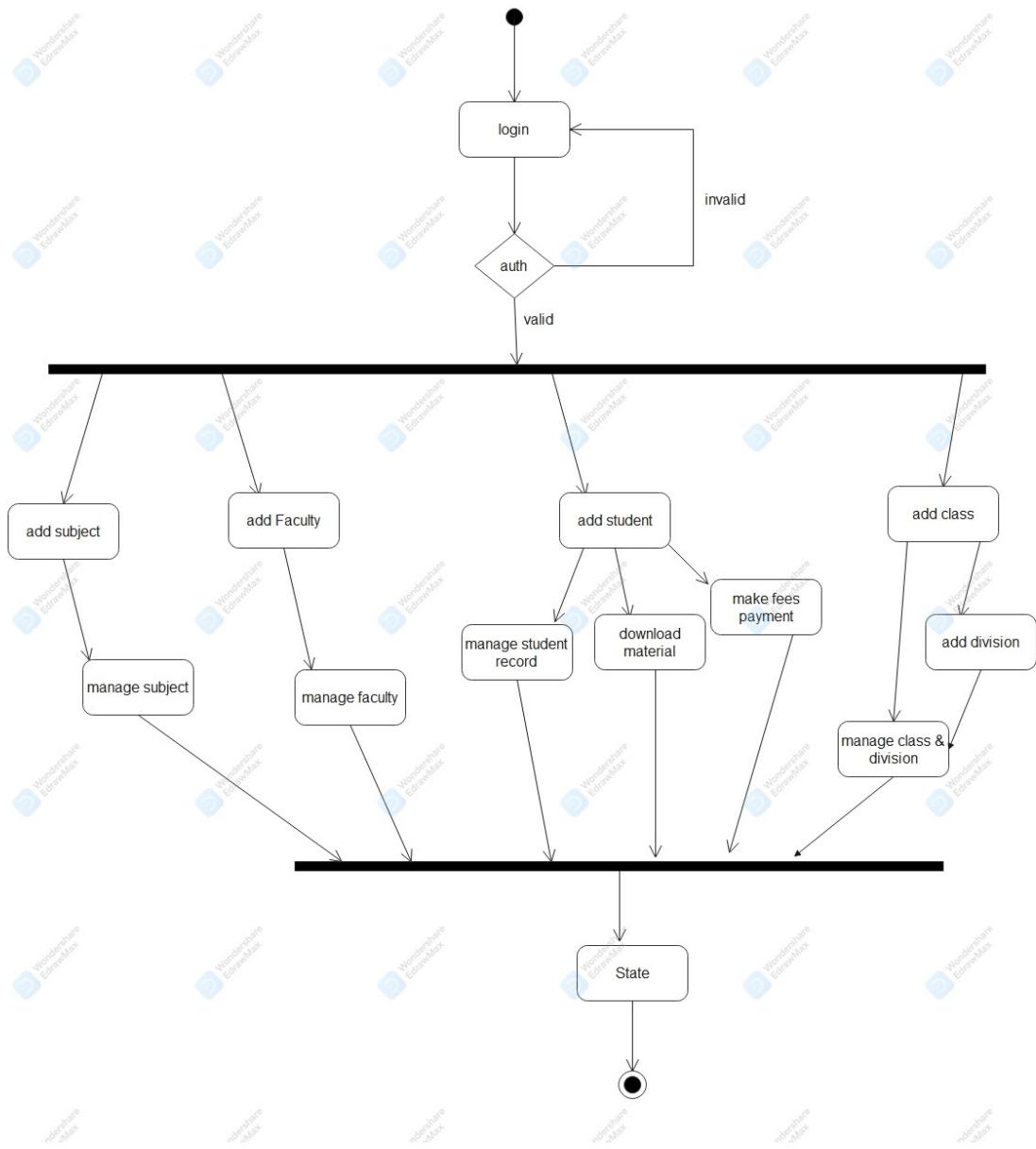


2)

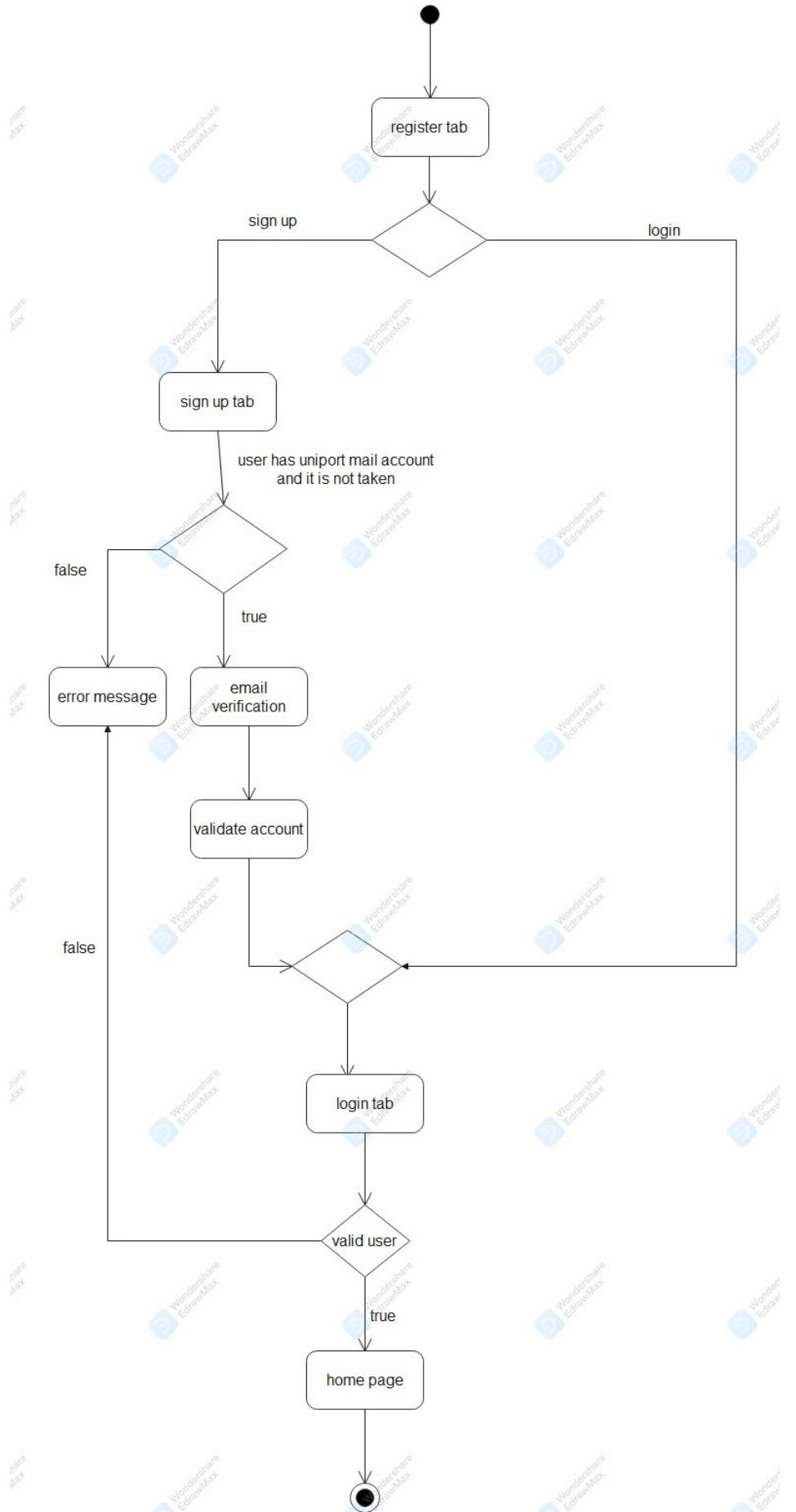


6. Activity diagram:

1.



2)



7.Data Dictionary :

7.1 Login Table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	s_no	int(11)			No	None		AUTO_INCREMENT
2	name	varchar(50)	utf8mb4_general_ci		No	None		
3	email	varchar(250)	utf8mb4_general_ci		No	None		
4	password	varchar(250)	utf8mb4_general_ci		No	None		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	s_no	0	A	No	

7.2 Students Table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	s_no	int(11)			No	None		AUTO_INCREMENT
2	roll_no	int(11)			No	None		
3	name	varchar(100)	utf8mb4_general_ci		No	None		
4	father_name	varchar(100)	utf8mb4_general_ci		No	None		
5	class	int(11)			No	None		
6	mobile	varchar(25)	utf8mb4_general_ci		No	None		
7	email	varchar(100)	utf8mb4_general_ci		No	None		
8	password	varchar(100)	utf8mb4_general_ci		No	None		
9	remark	varchar(250)	utf8mb4_general_ci		No	None		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	s_no	10	A	No	

7.3 Techers Table :

#	Name	Type	Collation	Attributes	Null	Default	Comments	Extra
1	s_no	int(11)			No	None		AUTO_INCREMENT
2	t_id	int(11)			No	None		
3	name	varchar(100)	utf8mb4_general_ci		No	None		
4	mobile	varchar(12)	utf8mb4_general_ci		No	None		
5	courses	varchar(250)	utf8mb4_general_ci		No	None		

Indexes

Keyname	Type	Unique	Packed	Column	Cardinality	Collation	Null	Comment
PRIMARY	BTREE	Yes	No	s_no	3	A	No	

5. Implementation Details

The system consists of 3 basic modules namely

- Authentication
- Manage Student
- Manage Teacher

1. Authentication Module

Users are able to login themselves. System logs user in, then and only then user can use other functionalities of system.

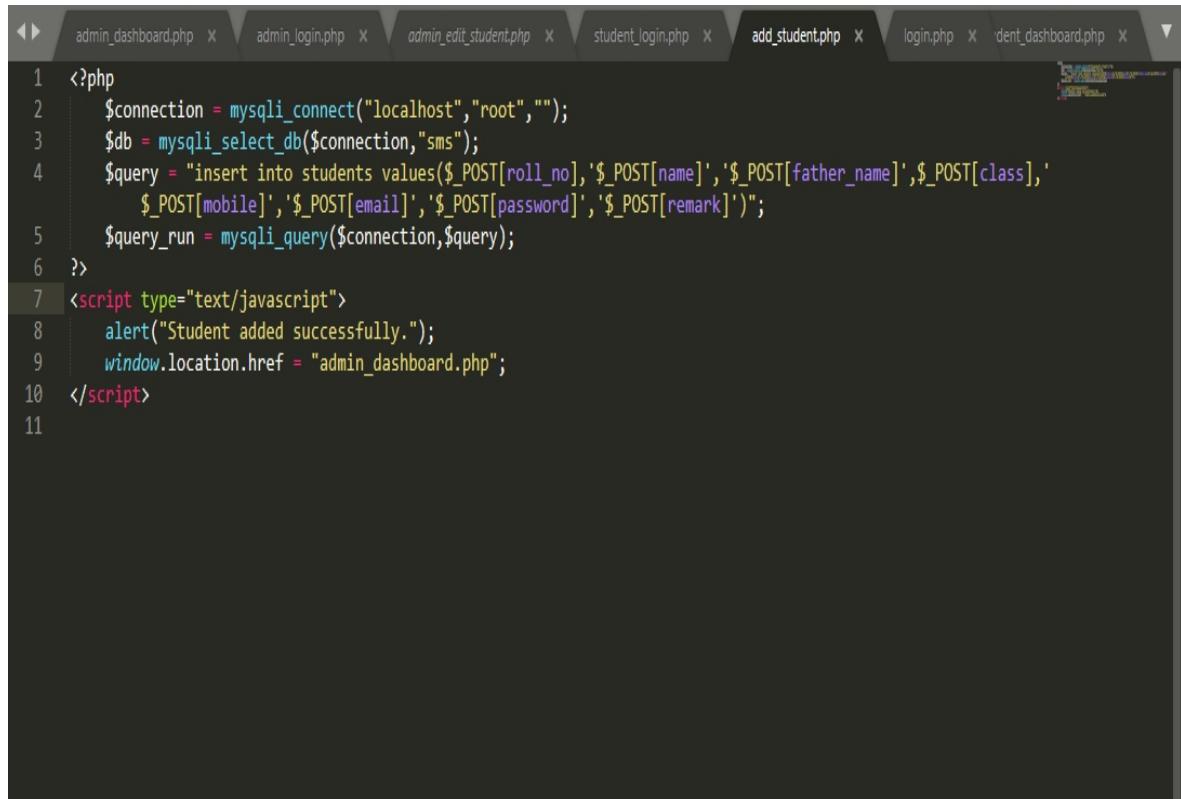
2. Manage Student Module

In Manage student modules admins have access to add students . If a student wants to changes any other detail of a student is changed then the teacher can update it also.Admin are able to delete student

3. Manage teacher Module

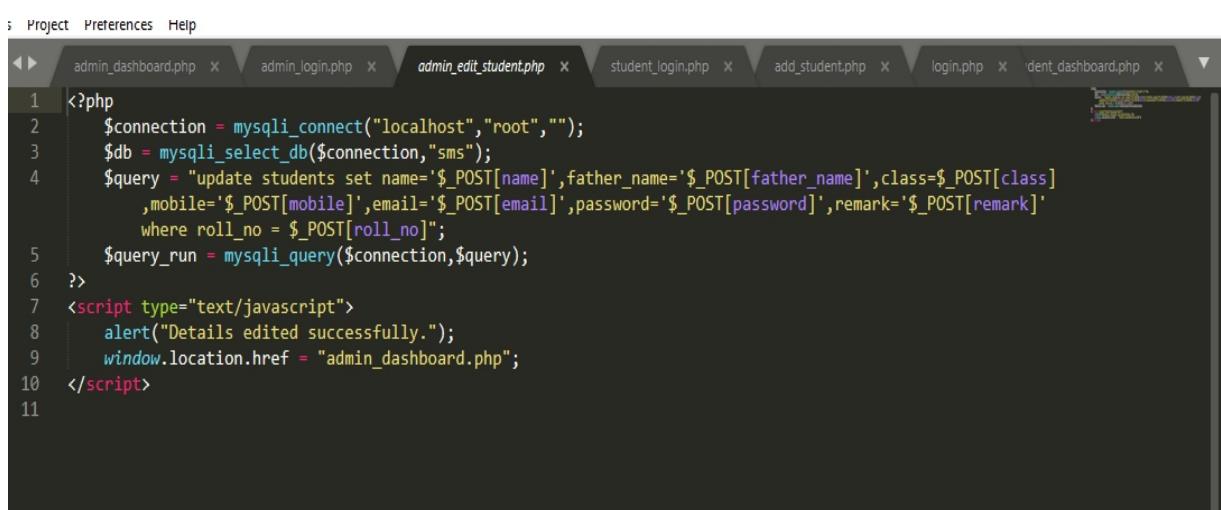
Manage teacher modules appear in the admin side. Admin have all access to add teachers. And can perform crud operations (add ,update,delete)on teacher details also.but here admin only see teachers list

Function prototypes:



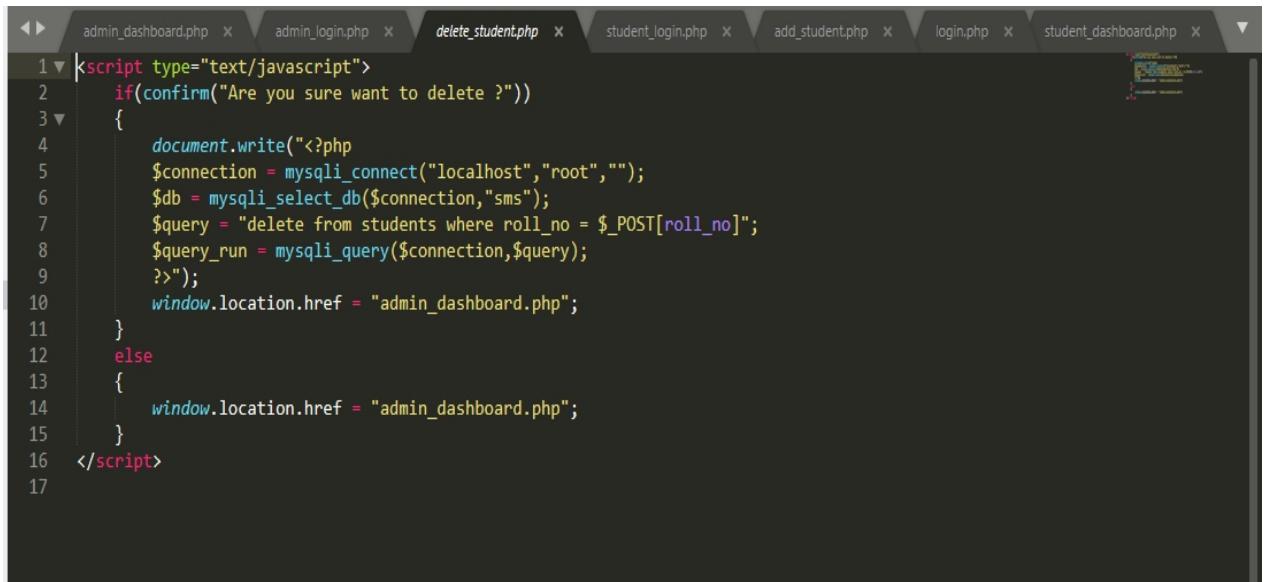
```
1 <?php
2     $connection = mysqli_connect("localhost","root","");
3     $db = mysqli_select_db($connection,"sms");
4     $query = "insert into students values($_POST[roll_no],$_POST[name],$_POST[father_name],$_POST[class],'
5         $_POST[mobile]','$_POST[email]','$_POST[password]','$_POST[remark]')";
6     $query_run = mysqli_query($connection,$query);
7 >
8 <script type="text/javascript">
9     alert("Student added successfully.");
10    window.location.href = "admin_dashboard.php";
11 </script>
```

Add student



```
1 <?php
2     $connection = mysqli_connect("localhost","root","");
3     $db = mysqli_select_db($connection,"sms");
4     $query = "update students set name='$_POST[name]',father_name='$_POST[father_name]',class=$_POST[class]
5             ,mobile='$_POST[mobile]',email='$_POST[email]',password='$_POST[password]',remark='$_POST[remark]'
6             where roll_no = $_POST[roll_no]";
7     $query_run = mysqli_query($connection,$query);
8 >
9 <script type="text/javascript">
10    alert("Details edited successfully.");
11    window.location.href = "admin_dashboard.php";
12 </script>
```

Update student

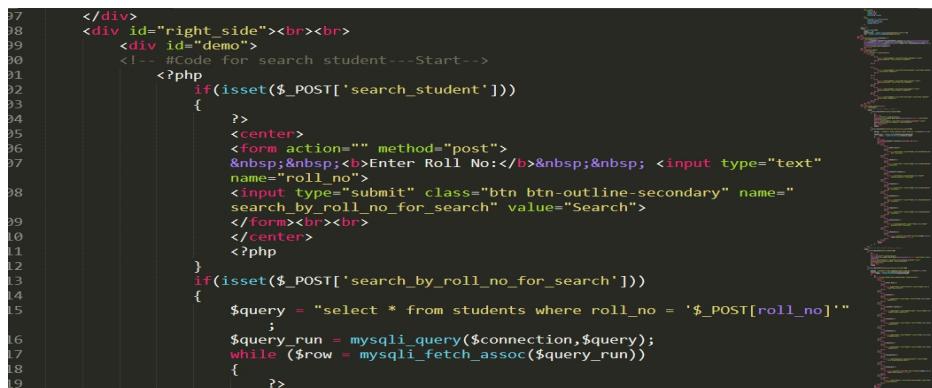


```

1 <script type="text/javascript">
2     if(confirm("Are you sure want to delete ?"))
3     {
4         document.write("<?php"
5             $connection = mysqli_connect("localhost","root","");
6             $db = mysqli_select_db($connection,"sms");
7             $query = "delete from students where roll_no = $_POST[roll_no]";
8             $query_run = mysqli_query($connection,$query);
9             ?>");
10        window.location.href = "admin_dashboard.php";
11    }
12    else
13    {
14        window.location.href = "admin_dashboard.php";
15    }
16 </script>
17

```

Delete student



```

97
98     </div>
99     <div id="right_side"><br><br>
100        <div id="demo">
101            <!-- #Code for search student---Start-->
102            <?php
103                if(isset($_POST['search_student']))
104                {
105                    ?>
106                    <center>
107                        <form action="" method="post">
108                            &nbsp;&nbsp;<b>Enter Roll No:</b>&nbsp;&nbsp; <input type="text"
109                            name="roll_no">
110                            <input type="submit" class="btn btn-outline-secondary" name=""
111                            search_by_roll_no_for_search" value="Search">
112                        </form><br><br>
113                    </center>
114                <?php
115                if(isset($_POST['search_by_roll_no_for_search']))
116                {
117                    $query = "select * from students where roll_no = '".$_POST[roll_no]."'"
118                    ;
119                    $query_run = mysqli_query($connection,$query);
120                    while ($row = mysqli_fetch_assoc($query_run))
121                    {
122                        ?>

```

Search student



```

88
89     ?>
90     <center>
91         <h5>Teacher's Details</h5>
92         <table>
93             <tr>
94                 <td id="td"><b>ID</b></td>
95                 <td id="td"><b>Name</b></td>
96                 <td id="td"><b>Mobile</b></td>
97                 <td id="td"><b>Courses</b></td>
98             </tr>
99         </table>
100     </center>
101     <?php
102     $query = "select * from teachers";
103     $query_run = mysqli_query($connection,$query);
104     while ($row = mysqli_fetch_assoc($query_run))
105     {
106         ?>
107         <center>
108             <table style="border-collapse: collapse; border: 1px solid black
109             ;">
110                 <tr>
111                     <td id="td"><?php echo $row['t_id']?></td>
112                     <td id="td"><?php echo $row['name']?></td>
113                     <td id="td"><?php echo $row['mobile']?></td>
114                     <td id="td"><?php echo $row['courses']?></td>
115                 </tr>
116             </table>
117         Go to Settings to activate Windows

```

View teacher

6. Testing

Testing method : manual testing

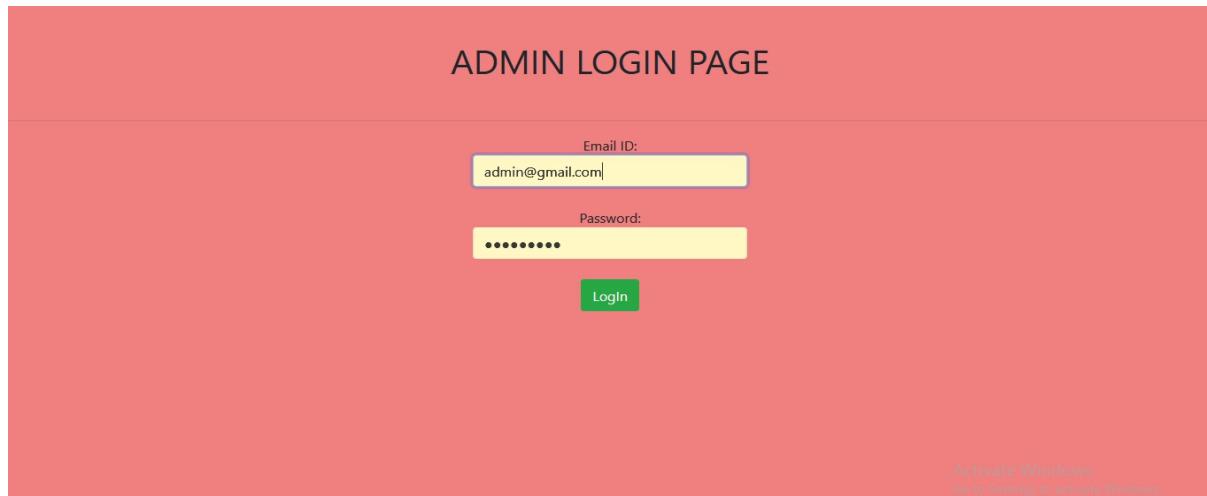
All test cases executed by us manually according to end user's perspective . our motto to test project is that it ensures matching with requirement document or not.

Sr. No.	Test Scenario	Expected Result	Actual Result	Status
TC_1 (Admin + student)	Login with incorrect credentials	User should not able to log in.	User is given error message. And redirected to login page.	Pass
TC_2 (Admin + student)	Login with correct credentials	User should be able to log in.	User is logged in and shown the dashboard according to input credentials.	Pass
TC_3 (Admin)	Search student (if exists)	Roll no matching Student's details are displayed .	Admin can see details of student.	Pass
TC_4 (Admin)	Delete Student (if exists)	Roll no matching Student record removed from db.	Admin can see confirmation message. And record are not in db.	Pass
TC_5 (Admin)	Add student	Roll no matching Student record Added into db.	Admin can see confirmation message & record added into db	Pass
TC_6 (Admin)	Update student (if exists)	Roll no matching Student record updated into db.	Admin can see confirmation message & record updated into db	Pass

7. Workflow/Layouts



Home Page



Admin Login Page

The screenshot shows a web browser window with a blue header bar containing various links like 'Import bookmarks...', 'SEM-4', 'dharma_158 | CodeChef', 'Classes', 'Folder - Google Drive', 'Inbox - Gmail', 'MYSY', 'My Drive', 'GITHUB', 'Settings | Django docu...', 'Blank board: Lucidspark', and 'Documents'. Below the header, a black navigation bar displays 'STUDENT INFORMATION MANAGEMENT SYSTEM', 'Email: admin@gmail.com', 'Name: Admin', and a 'Logout' link. On the left, a sidebar menu includes 'Search Student' (grey), 'Edit Student' (yellow), 'Add New Student' (green, highlighted), 'Delete Student' (red), and 'Show Teachers' (dark grey). The main content area has a red background and features a search form with a text input 'Enter Roll No: 1001' and a 'Search' button. In the bottom right corner of the main area, there is a watermark: 'Activate Windows Go to Settings to activate Windows'.

Search student

The screenshot shows a web browser window with multiple tabs open: 'localhost / 127.0.0.1 / sms / stu', 'Report.pdf', 'Admin Dashboard', and 'localhost / 127.0.0.1 | phpMyA...'. The browser's address bar shows 'localhost/SMS Project/admin_dashboard.php'. The page layout is identical to the previous screenshot, with the same blue header, black navigation bar, and sidebar menu. The main content area now displays a table titled 'Student's details' showing the following information:

Roll No:	1001
Name:	Vinod Meena
Father's Name:	chetanbhai
Class:	12
Mobile:	1234567789
Email:	vinod@gmail.com
Password:	*****
Remark:	very Fineee!!!!

In the bottom right corner of the main area, there is a watermark: 'Activate Windows Go to Settings to activate Windows'.

Displaying Details

STUDENT INFORMATION MANAGEMENT SYSTEM Email: admin@gmail.com Name: Admin [Logout](#)

[Search Student](#)
[Edit Student](#)
[Add New Student](#) (Current)
[Delete Student](#)
[Show Teachers](#)

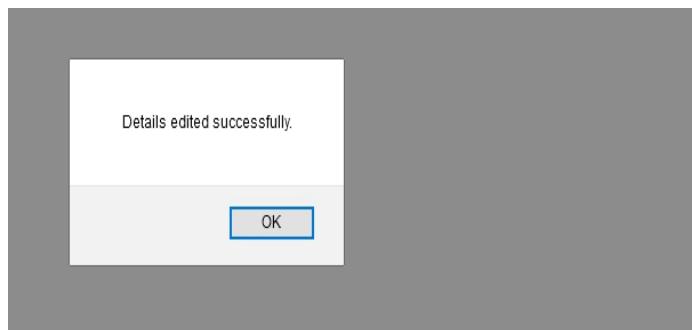
Roll No:	1001
Name:	Vinod Meena
Father's Name:	chetanbhai
Class:	12
Mobile:	1234567789
Email:	vinod123@gmail.com
Password:	*****
Remark:	very Finee!!!!

[Save](#)

Activate Windows
Go to Settings to activate Windows

Edit student

Change = email : **vinod@gmail.com** instead of vinod123@gmail.com



Confirm msg

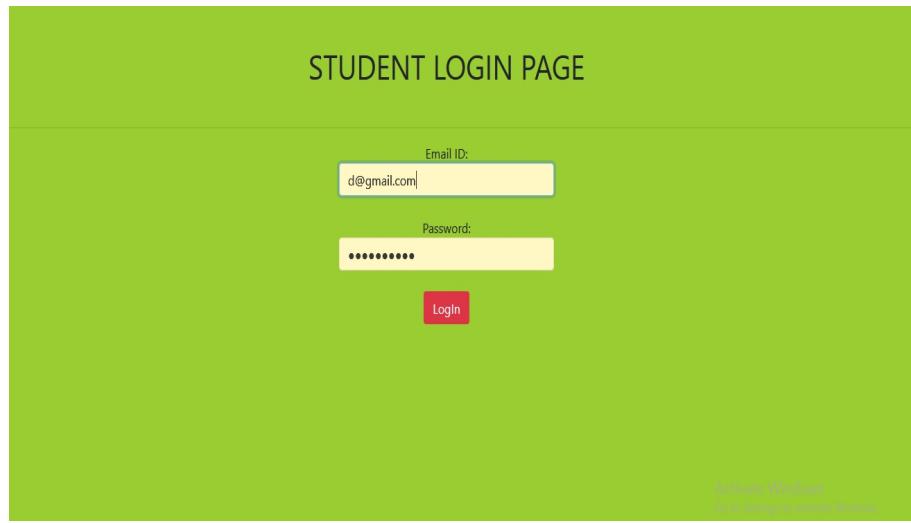
STUDENT INFORMATION MANAGEMENT SYSTEM Email: admin@gmail.com Name: Admin [Logout](#)

[Search Student](#)
[Edit Student](#)
[Add New Student](#) (Current)
[Delete Student](#)
[Show Teachers](#)

Teacher's Details			
ID	Name	Mobile	Courses
101	Shivam Yadav	5484654878	Physics, Maths
102	Gopal Sharma	9878452484	English, Reasoning, History
103	Hariom sain	7887451254	Politics, History, Biology

Activate Windows
Go to Settings to activate Windows

Show Teacher



Student Login Page

A screenshot of a student detail view page. The top navigation bar is black with white text: "STUDENT INFORMATION MANAGEMENT SYSTEM", "Email: d@gmail.com", "Name: dharmi", and "Logout". Below the navigation bar is a blue sidebar on the left with a yellow button labeled "Show Detail". The main content area has a pink background. It displays student information in a table format:

Roll No:	1005
Name:	dhari
Father's Name:	smitbhai
Class:	1
Mobile:	123457891
Email:	d@gmail.com
Password:	*****

very good girl..

Remark Given By Teacher :

Activate Windows
Go to Settings to activate Windows

Show deatails

8. Conclusion

We have successfully implemented the Admin-side & Student-side functionality. Admin will add the new student &edit student deatails. Admin can show all the students and can also delete students record whoever is left from school admin also can see teachers list which contain (id,name ,mob-no ,courses).student who login from student login page he/she can also see his/her details.

9. Limitation and Future Extension

Limitations :

Students are not able to register yourself but once admin added into database then he/she can able to show their details.

Future extensions :

As a beginners we tried to implement admin & student side pannel . also we can implement to manage teachers details with crud operations in next extension.Teachers side module is also not implemented so they can not realate with any assignments realated stuff. Also fees realated modules yet not implemented.

10. Bibliography

References/resources used for developing project:

- Head First PHP & MySQL by Lynn Beighley and Michael Morrison(O'Reilly)
- <https://getbootstrap.com/docs/4.0>
- www.w3schools.com
- www.google.com
- www.stackoverflow.com
- www.wikipedia.com
- <http://localhost/phpmyadmin/index.php>(database)