1. Introduction

1.1 Project description

The main objective of the Student Management System is to manage the details of Profiles, Logins. It manages all the information about Profiles, Student, password. The project is totally built at administrative end and thus only the administrator is guaranteed the access.

Now a Days everyone required a software to manage student details. In this you can see all the information of the students and the student can also login their profile.

Student management system using PHP and MySQL is a web-based application. Student Management Project is software that is helpful for students as well as the school authorities. In the current system, all the activities are done manually. It is very time-consuming and costly. Our online Student Management System in PHP deals with the various activities related to the students.

1.2 Project Profile

Project Title:	Student Management System
Definetion:	Student Management System is the system where all the aspects related to the proper management of student information is these aspects involve managing information about the student details.
Developed For:	SDJ International College, Vesu, Surat
Project Guide(s):	
Front End:	Visual studio Code, HTML
Scripting language :	PHP, JavaScript
Back End :	Xampp Server
Operating System:	Microsoft Windows 10
Tools used for ERD & DFD :	Microsoft Word
Submitted By:	Ashish Sarvaiya

2. Environment Description

2.1 Hardware and Software Requirements

A. Hardware Requirements :

- System : **Intel core i5 Processor**
- Hard Disk : **520GB**
- Ram : **4GB**

B. Software Requirements :

- Operating system : **Microsoft Windows 10**
- Coding Language : **PHP**
- Data Base: **MySQL**

2.2 Technologies Used

- Front End **: HTML**
- Designing **: CSS , Bootstrap , jQuery**
- Scripting Language **: JavaScript**
- Back-End **: Xampp Server**

3. System Analysis and Planning

3.1 Existing System and its Drawbacks

- The student result management system is prone to hacks.
- Administration can edit or modify scores after the deadline.
- Extensive modules and features make it difficult for a user to utilize the application.
- Minor technical glitches and issues.

3.2 Feasibility Study

The purpose of the feasibility study is to determine whether the problem can be solved with minimum cost as soon as possible. Nowadays, the price of the computer has been very low, while the performance has made considerable progress.

3.3 Requirement Gathering and Analysis

- Functional Requirement :
  - Creationof the new record for the new student.
  - Deletion of the record which already exists in the system based on the requirement of the institute.
  - Update in the record which is present in the system as per the need.
  - Generate the report on the attendance of the student as per his/her record.
  - Admin's handle of the department, this function eases the process of management.

- Non-Functional Requirement :
  - The security of the system is maintained by providing a login interface to the user. Only those who have the login is and password can enter the system.
  - User-Friendly as the system is very interactive and can be easily operated.
  - Maintainability and reliability if the system is kept very thoroughly as all the records kept in the database have the backups and system can restore if there is power loss.

4. Proposed System

4.1 Scope

The proposed system will affect or interface with the user and admin.

The system works and fulfills all the functionalities as per the proposed system. It will provide reduced response time against the queries made by different users. This project is based on PHP language with MySQL database which manage the detailsof the students because it is a tedious job for any organization. Student information system will store all the details of the students including their background information.

All possible features such as verification, validation, security, user friendliness etc. have been considered.

4.2 Project modules

- The two types of modules present in this project are
  1. Admin
  2. User

4.3 Module vise objectives/functionalities Constraints

1. Admin :

- Dashboard: In this section, admin can see all detail in brief like Total Classes, Total Students, Total Class Notices and Total Public Notices.
- Class: In this section, admin can manage class (Add/Update/Delete).
- Students: In this section, admin can manage the students (Add/Update/Delete).
- Notices: In this section, the admin can manage notices (Add/Update/Delete).
- Public Notices: In this section, the admin can manage public notices.
- Pages: In this section admin, can manage about us and contact us page of administration.
- Search: In this section admin, can search students by their student id.
- Reports: In this section admin, can view how much students has been register in particular period.
- Admin can also update his profile, change the password and recover the password.

2. **User** (Students) **:**

- Dashboard: It is welcome page for students.
- View Notices: In this section, user can view notices which are announced by administrator.
- Student can also view his profile, change the password and recover the password.

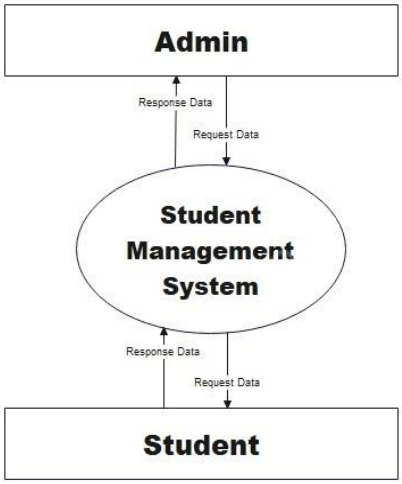
User (Non-Register):

- Home: It is welcome page for user.
- About: User can view about us page.
- Contact: User can view contact us page.

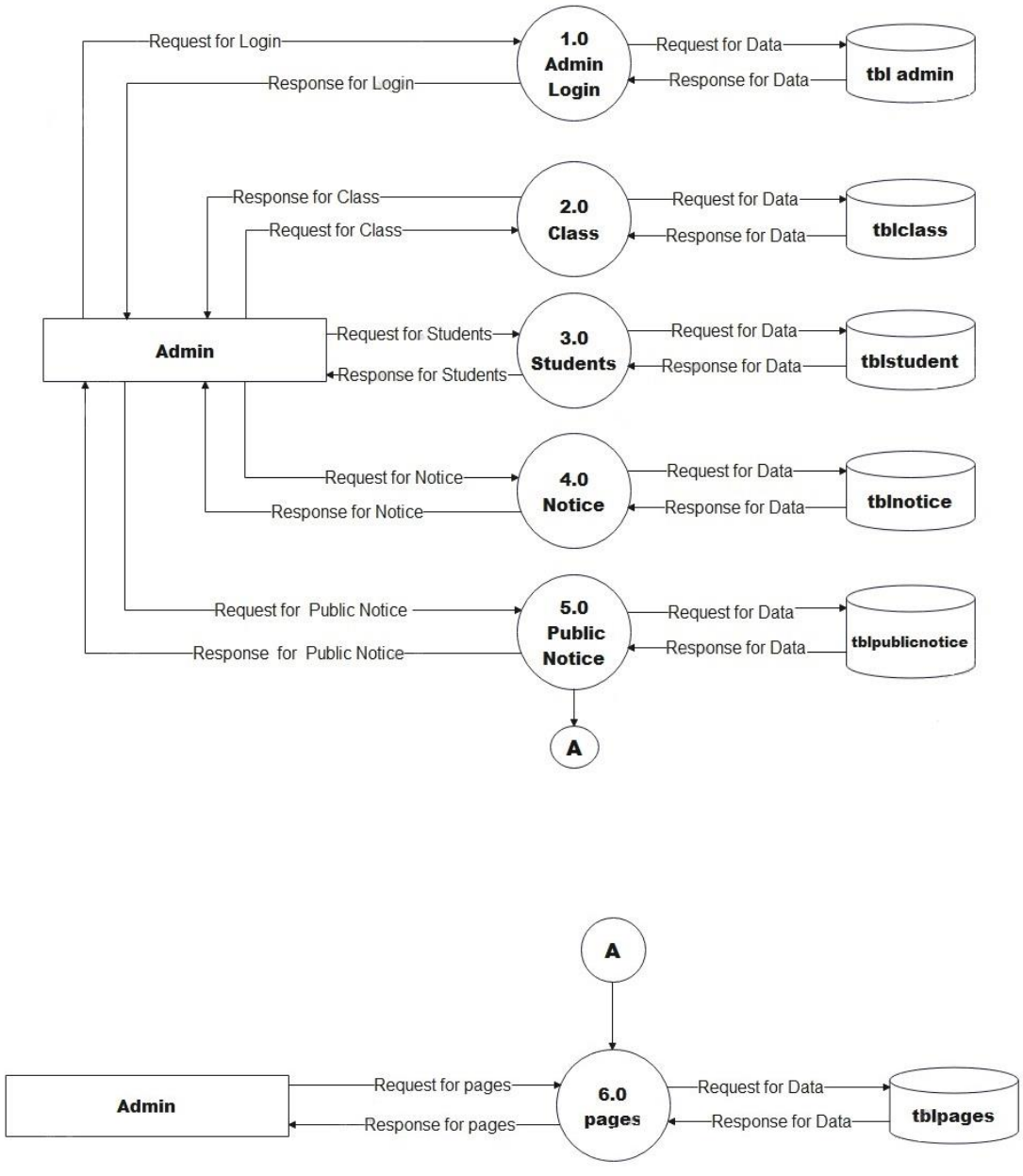
5. Detail Planning

5.1 Data Flow Diagram / UML

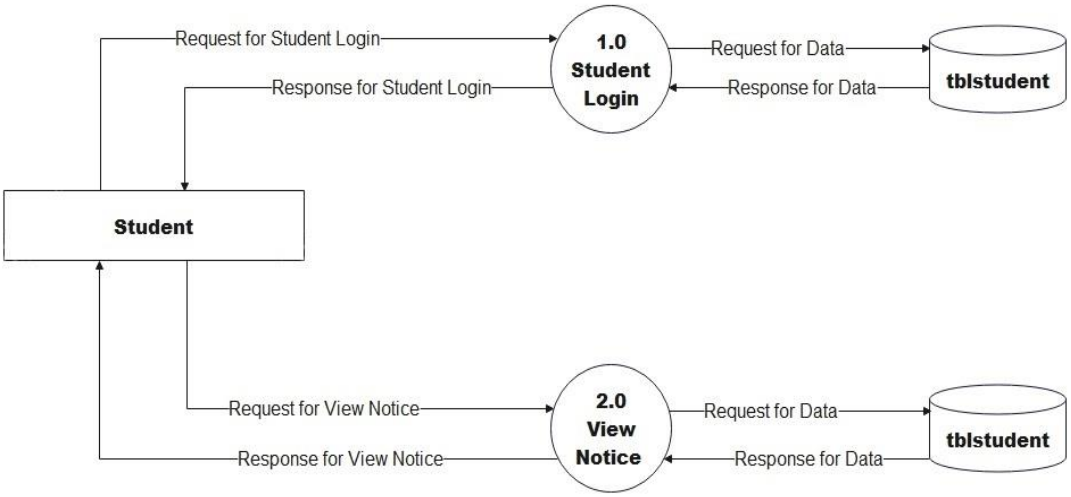
1) Context Level :



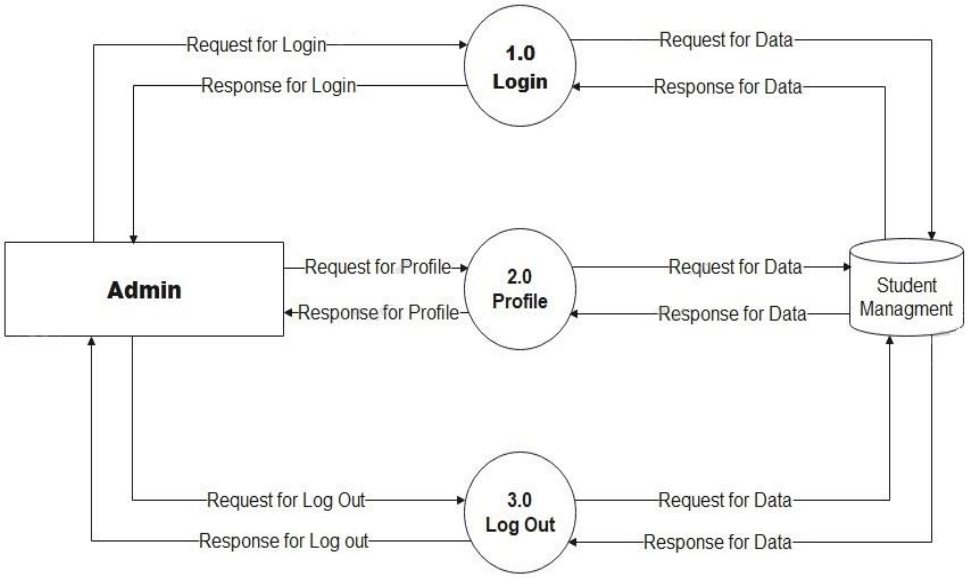
2) First Level DFD of Admin Side :



3) First Level DFD of Student Side :

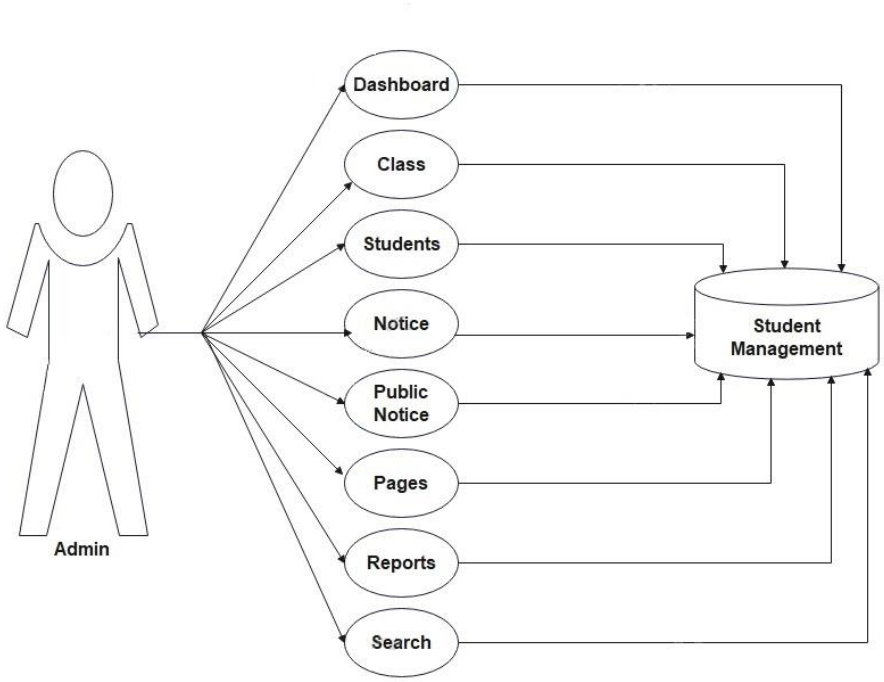


4) Second Level DFD of Admin Side :

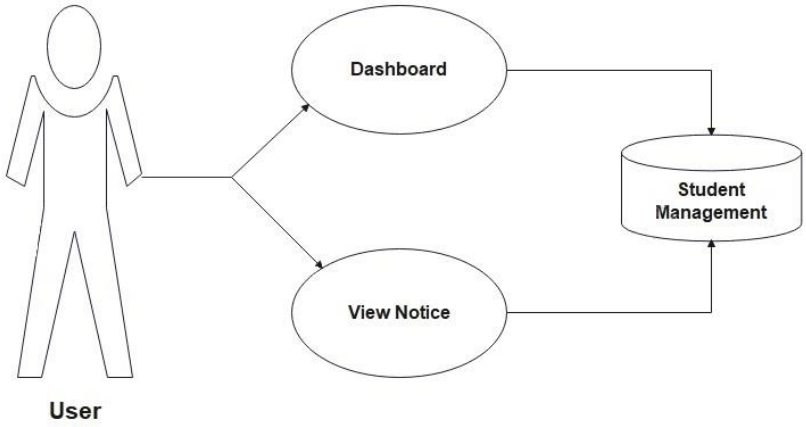


5.2 Process Specification / Activity Flow Diagram

1) Use Case Diagram :

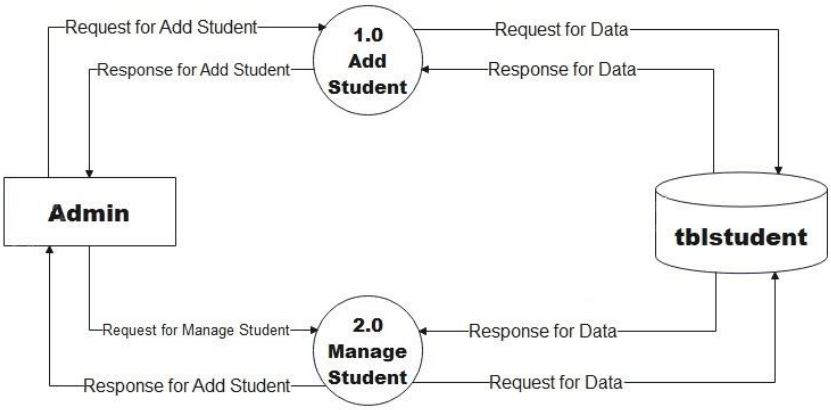


2) Student :

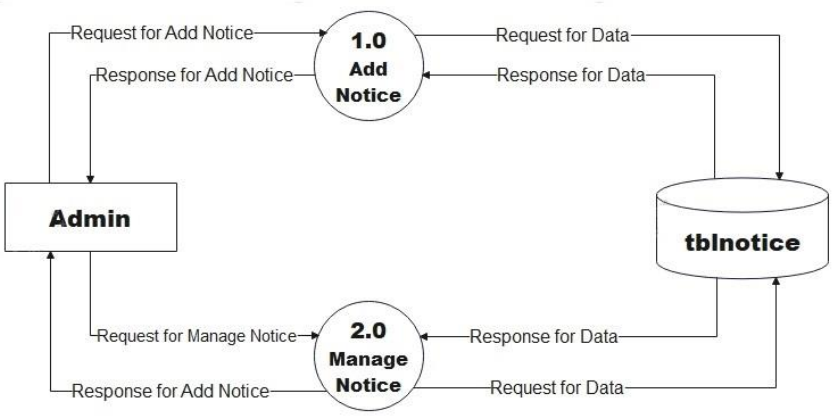


5.3 Data Dictionary

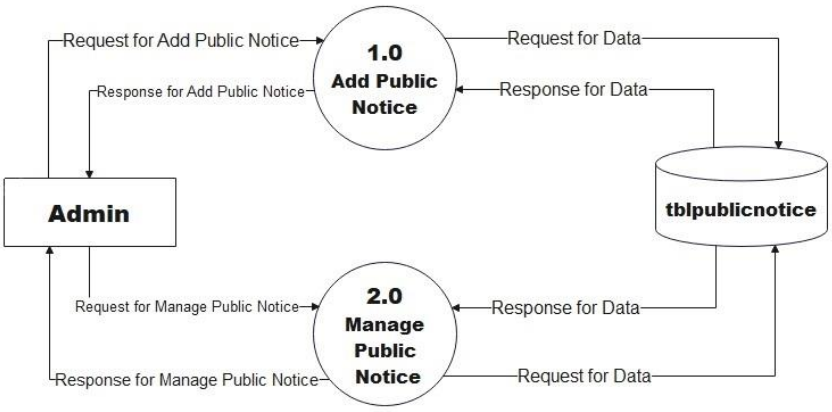
1) Student Management :



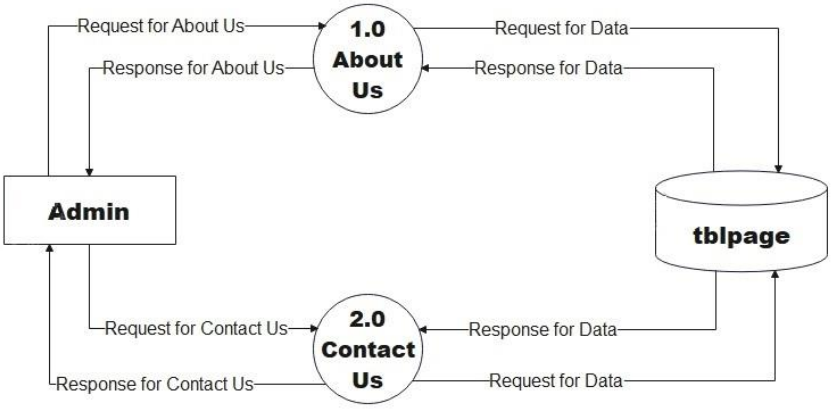
2) Notice Management :



3) Public Notice Management :

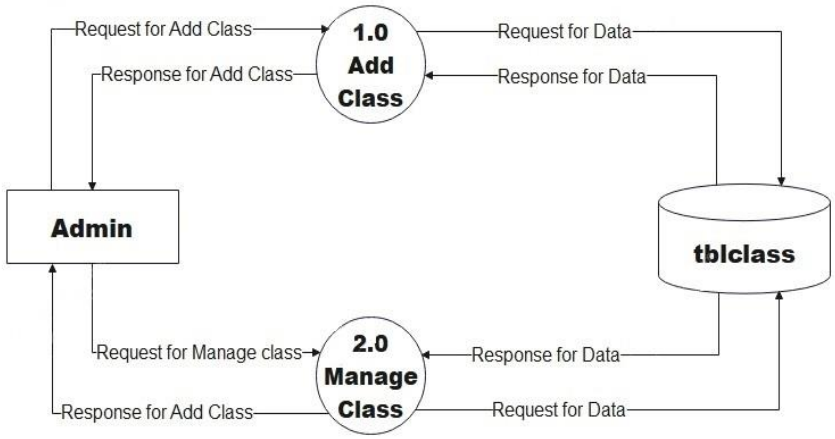


4) Pages Management :



5.4 Entity-Relationship Diagram / Class Diagram

1) Class Management :



6. System Design

6.1 Database Design

Database Handling Table	
No.	Table_Name
1.	tbladmin
2.	tblclass
3.	tblnotice
4.	tblpage
5.	tblpublicnotice
6.	tblstudent

1. tbladmin		
Column	Datatype	Constraint
Id	int(10)	AUTO_INCREMENT PRIMARY_KEY
AdminName	varchar(120)	Admin name required
UserName	varchar(120)	Username required
MobileNumber	bigint(10)	Phone no. required
Email	varchar(200)	Email required
Password	varchar(200)	Password required
AdminRegdate	timestamp	

2. tblclass		
Column	Datatype	Constraint
Id	int(5)	AUTO_INCREMENT PRIMARY_KEY
ClassName	varchar(50)	class name required
Section	varchar(20)	Section required
CreationDate	timestemp	

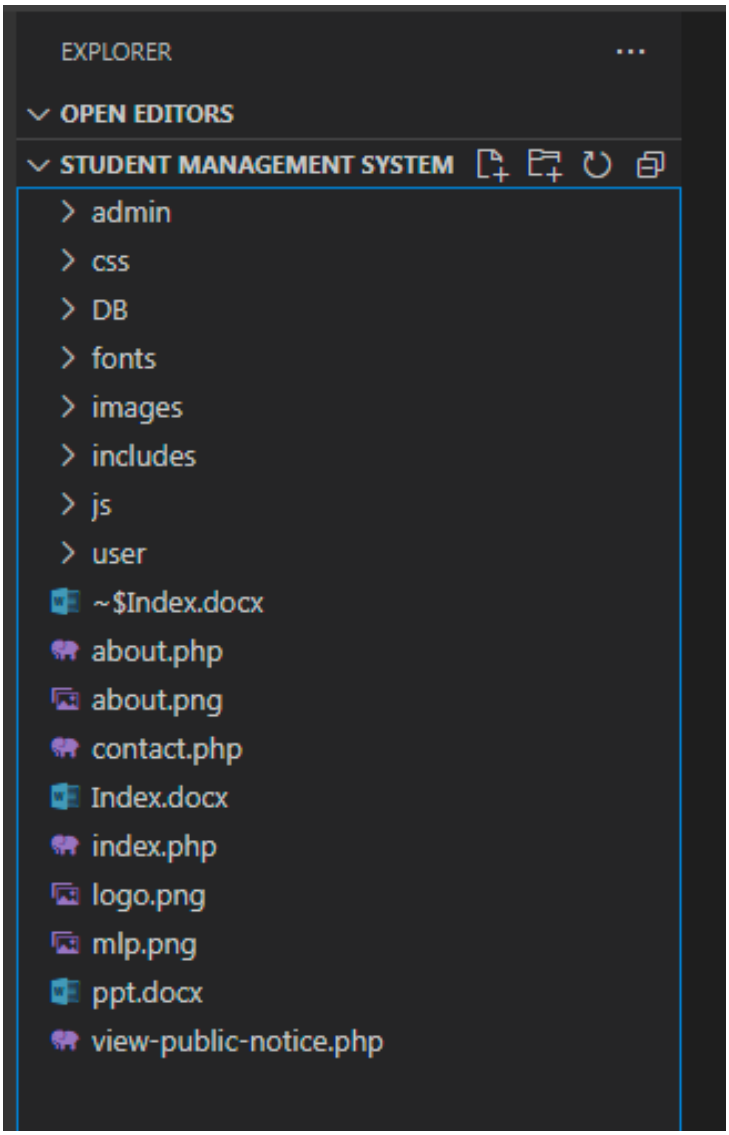
3. tblnotice		
Column	Datatype	Constraint
Id	int(5)	AUTO_INCREMENT PRIMARY_KEY
NoticeTitle	mediumtext	Notice title required
ClassId	int(10)	required
NoticeMsg	mediumtext	required
CreationDate	timestemp	

4. tblpage		
Column	Datatype	Constraint
Id	int(10)	AUTO_INCREMENT PRIMARY_KEY
PageType	varchar(200)	
PageTitle	mediumtext	required
PageDescription	mediumtext	required
Email	varchar(200)	Email required
MobileNumber	Bigint(10)	required
UpdationDate	date	

5. tblpublicnotice		
Column	Datatype	Constraint
Id	int(10)	AUTO_INCREMENT PRIMARY_KEY
NoticeTitle	varchar(200)	required
NoticeMessage	mediumtext	required
CreationDate	timestamp	

6. tbladmin		
Column	Datatype	Constraint
Id	int(10)	AUTO_INCREMENT PRIMARY_KEY
StudentName	varchar(200)	required
StudentEmail	varchar(200)	required
StudentClass	varchar(100)	required
Gender	varchar(50)	required
DOB	Date	required
StuID	Varchar(200)	required
FatherName	mediumtext	required
MotherName	mediumtext	optinal
ContactNumber	bigint(10)	required
AltenateNumber	bigint(10)	required
Address	mediumtext	required
UserName	Varchar(200)	required
Password	Varchar(200)	Required
Image	Varchar(200)	Null
DateofAdmission	timestamp	

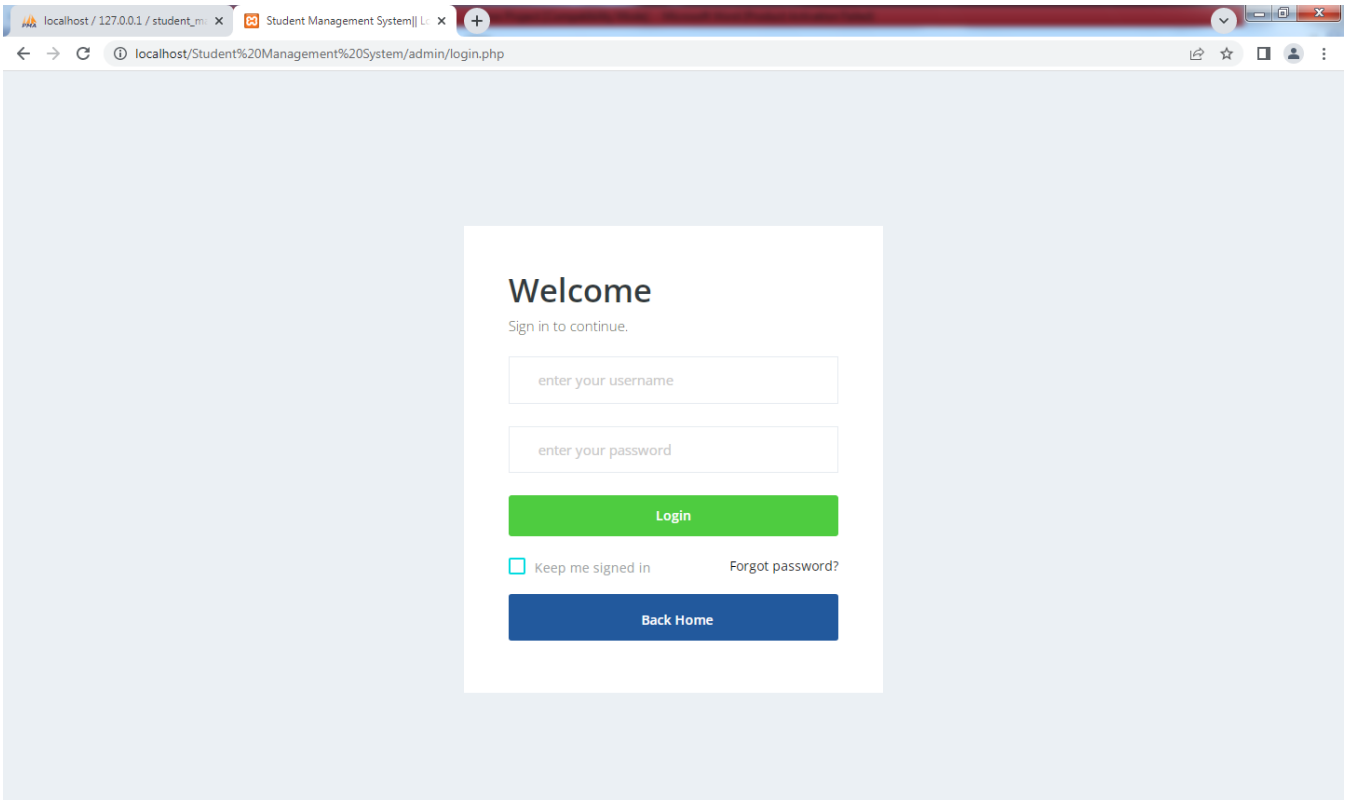
6.2 Directory Structure



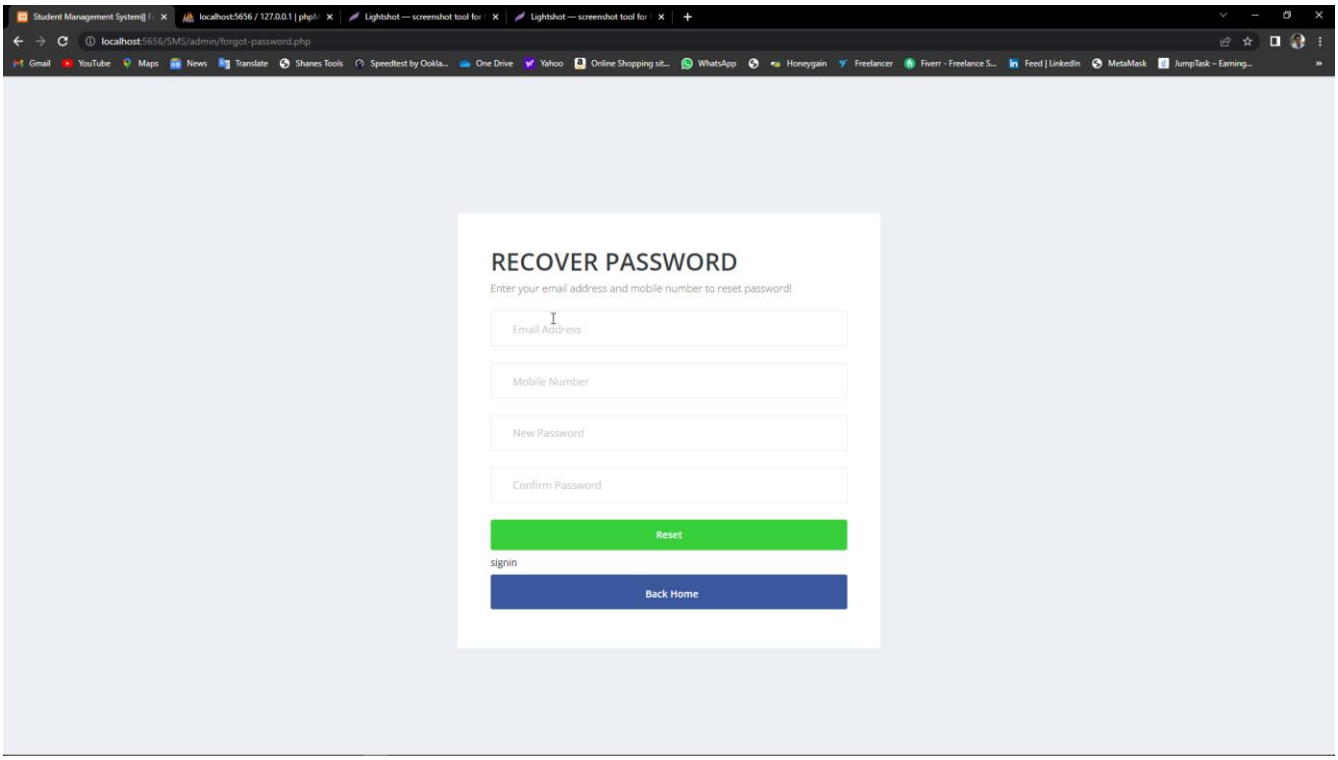
6.3 Input Design

A. Admin Side :

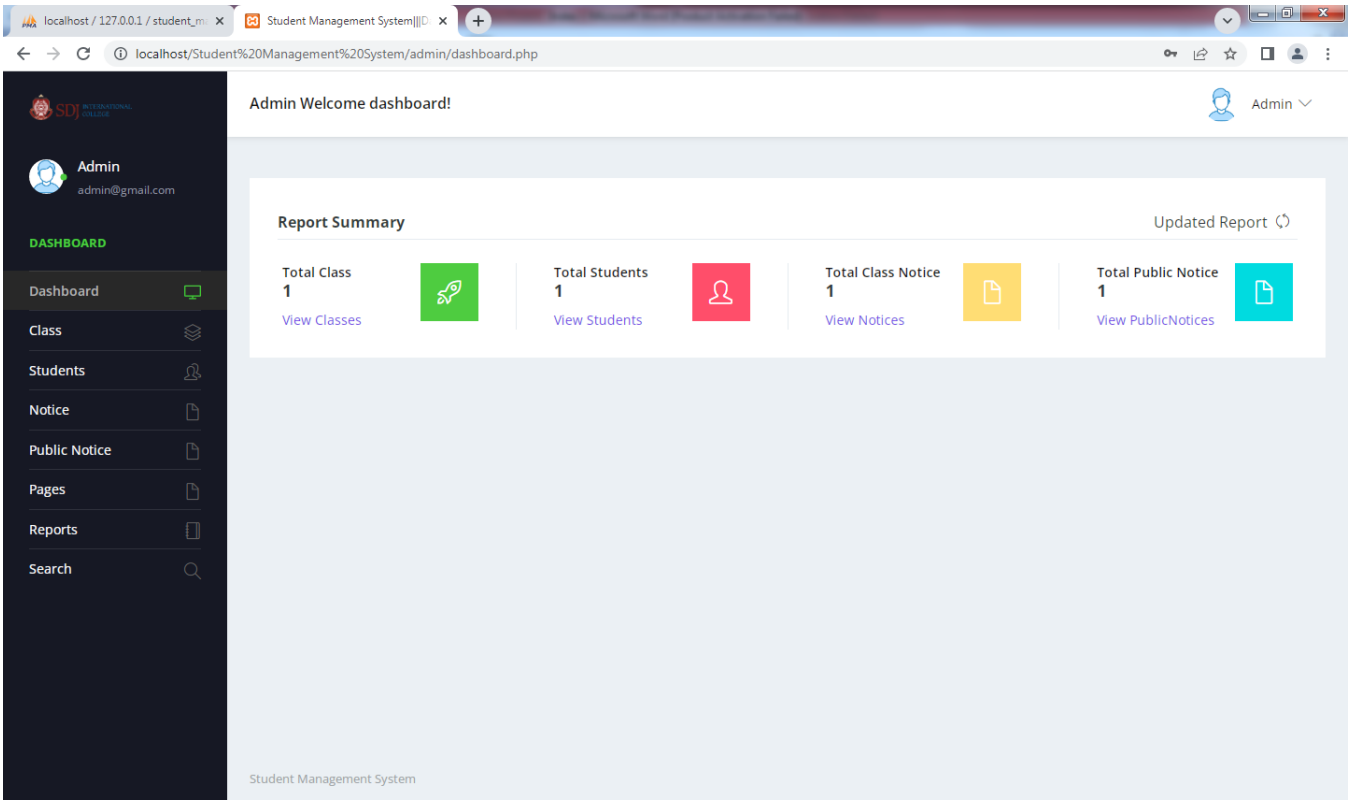
1. Admin Login :



2. Admin Recover Password :

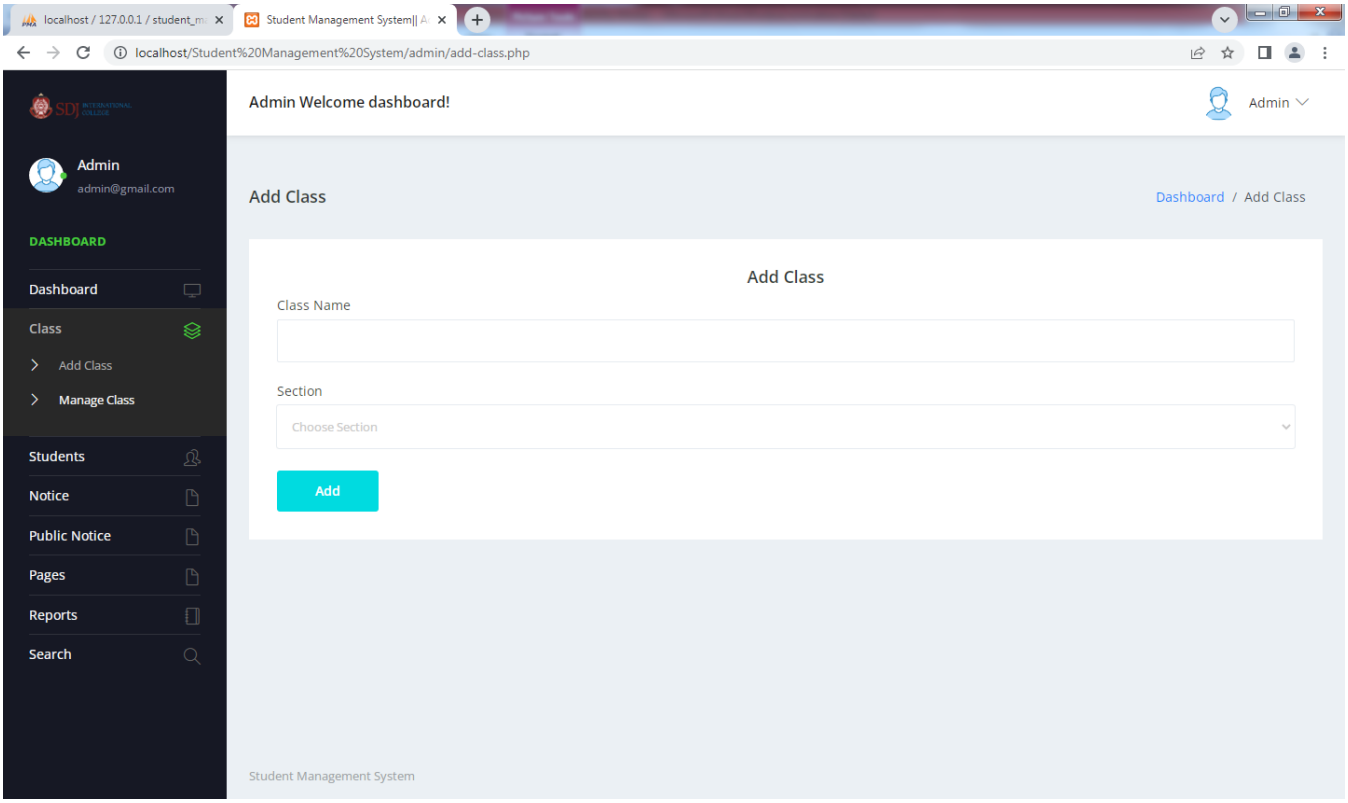


3. Admin Dashboard :

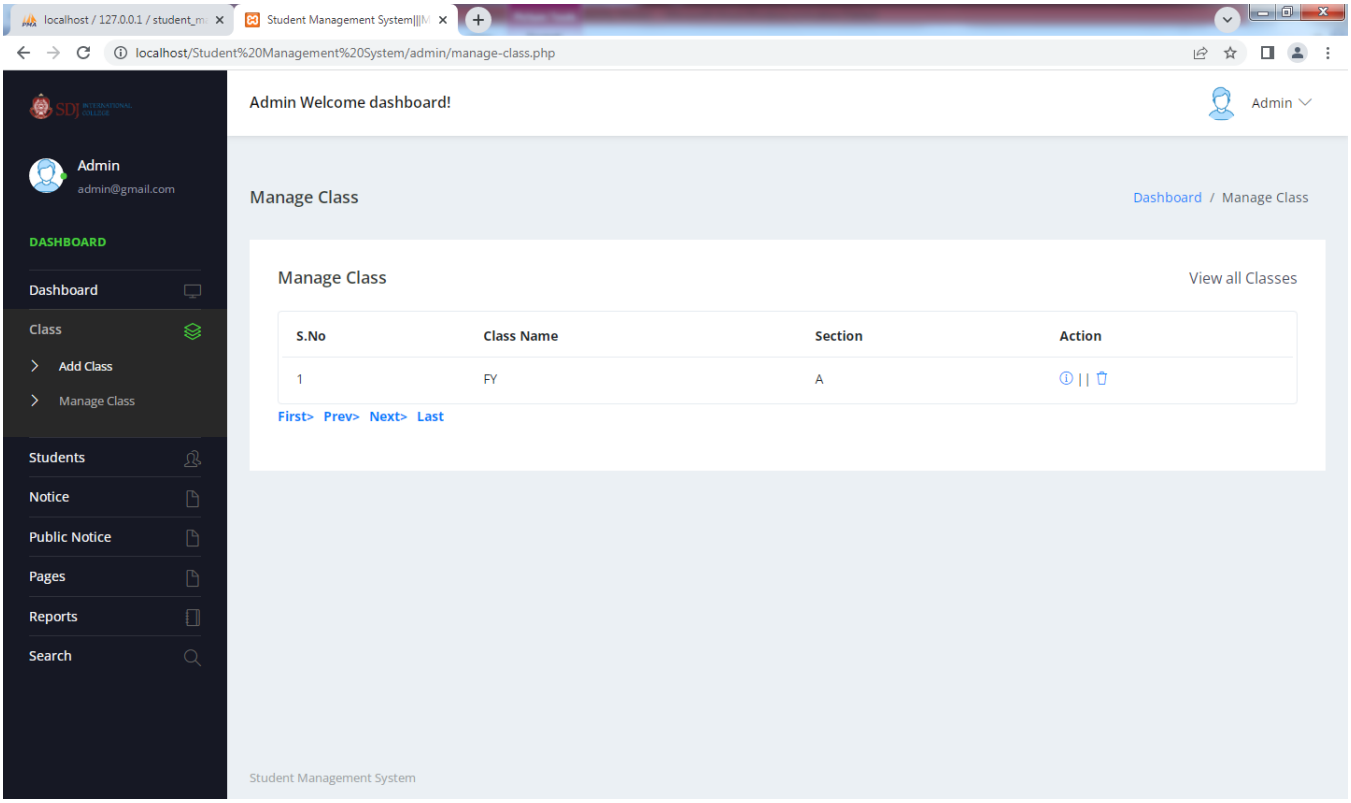


4. Class Management :

   - Add Class :

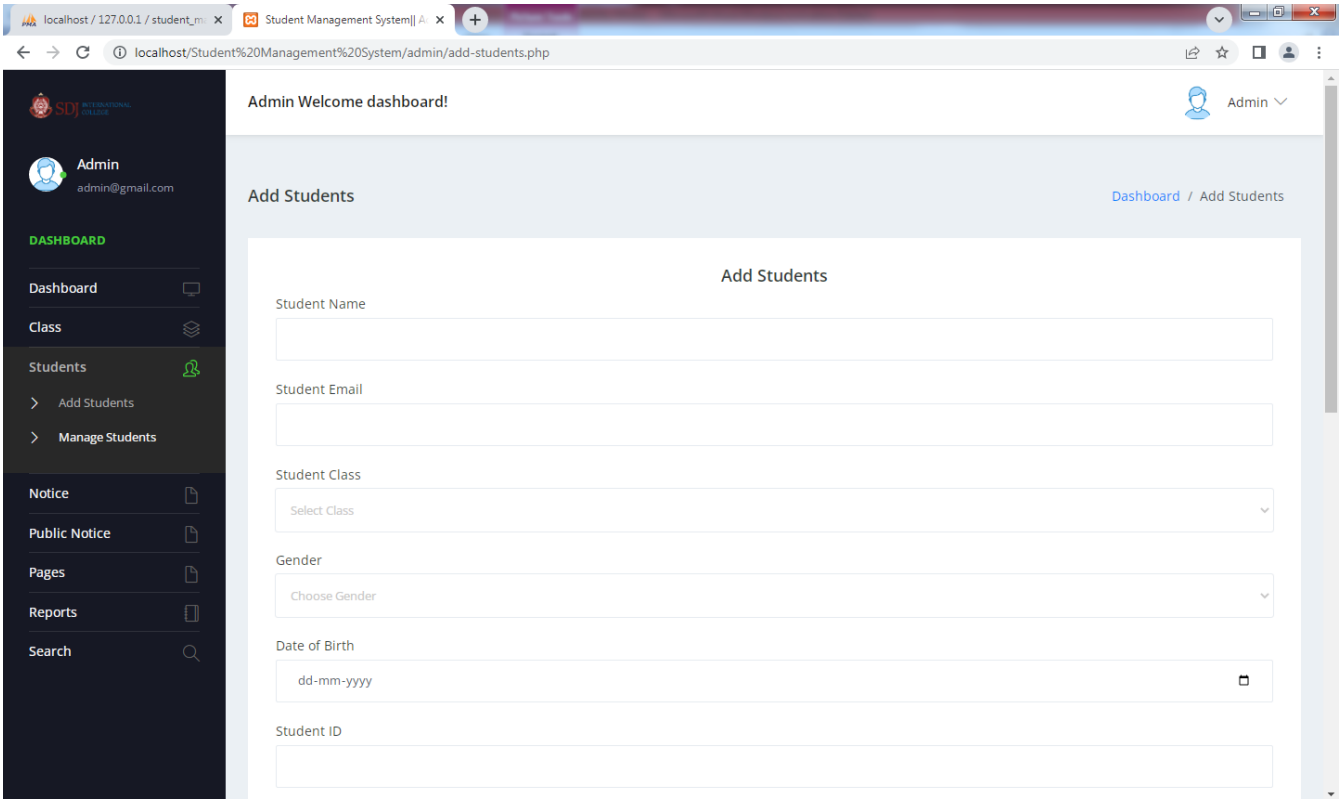


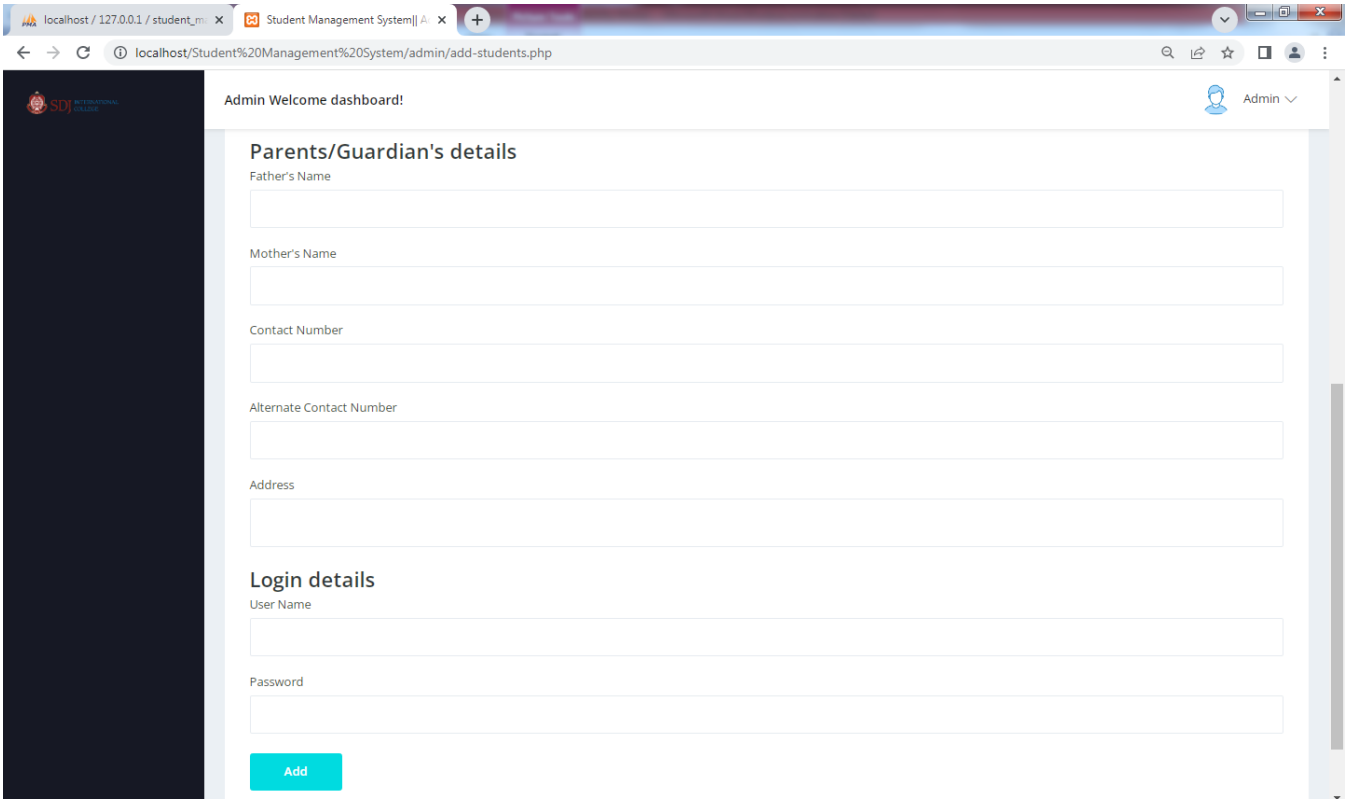
- Manage Class :



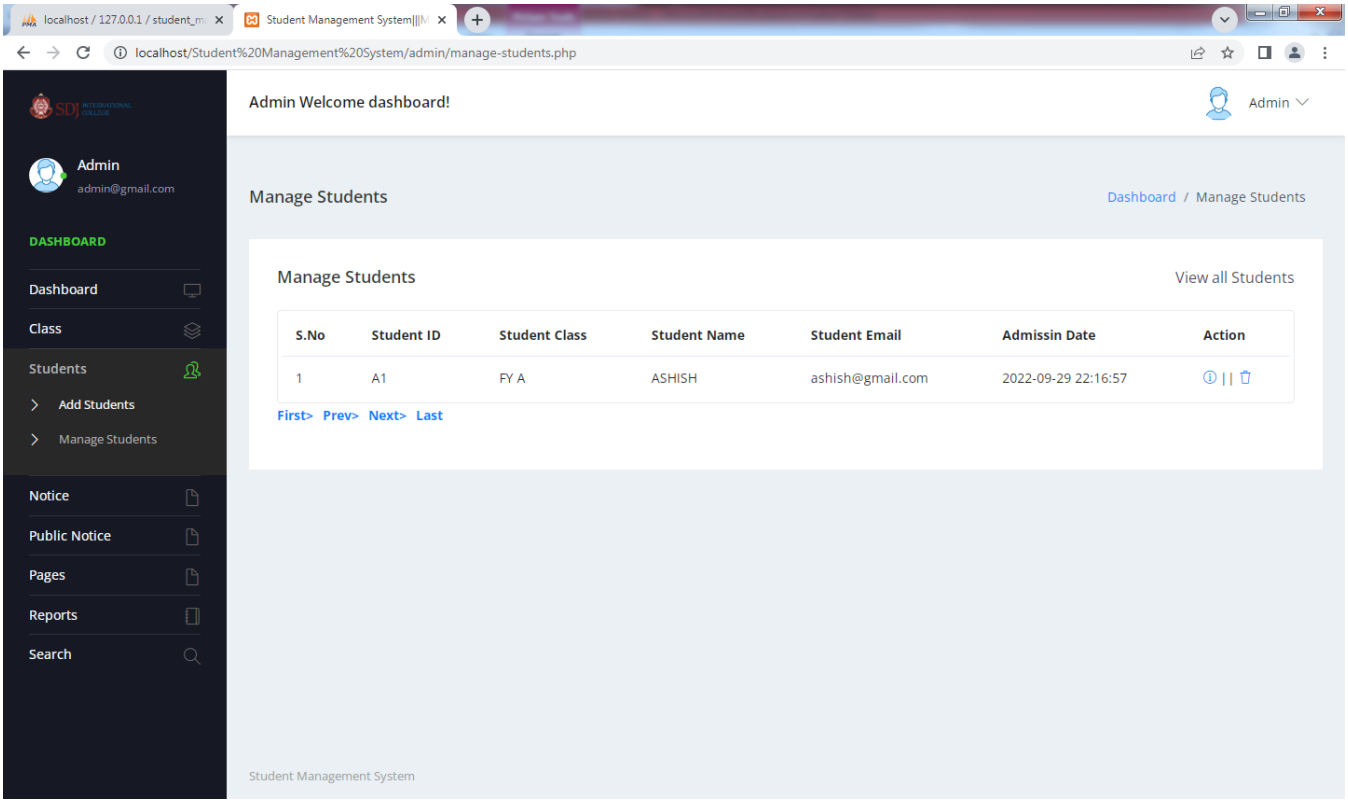
5. Student Management :

   - Add Student :



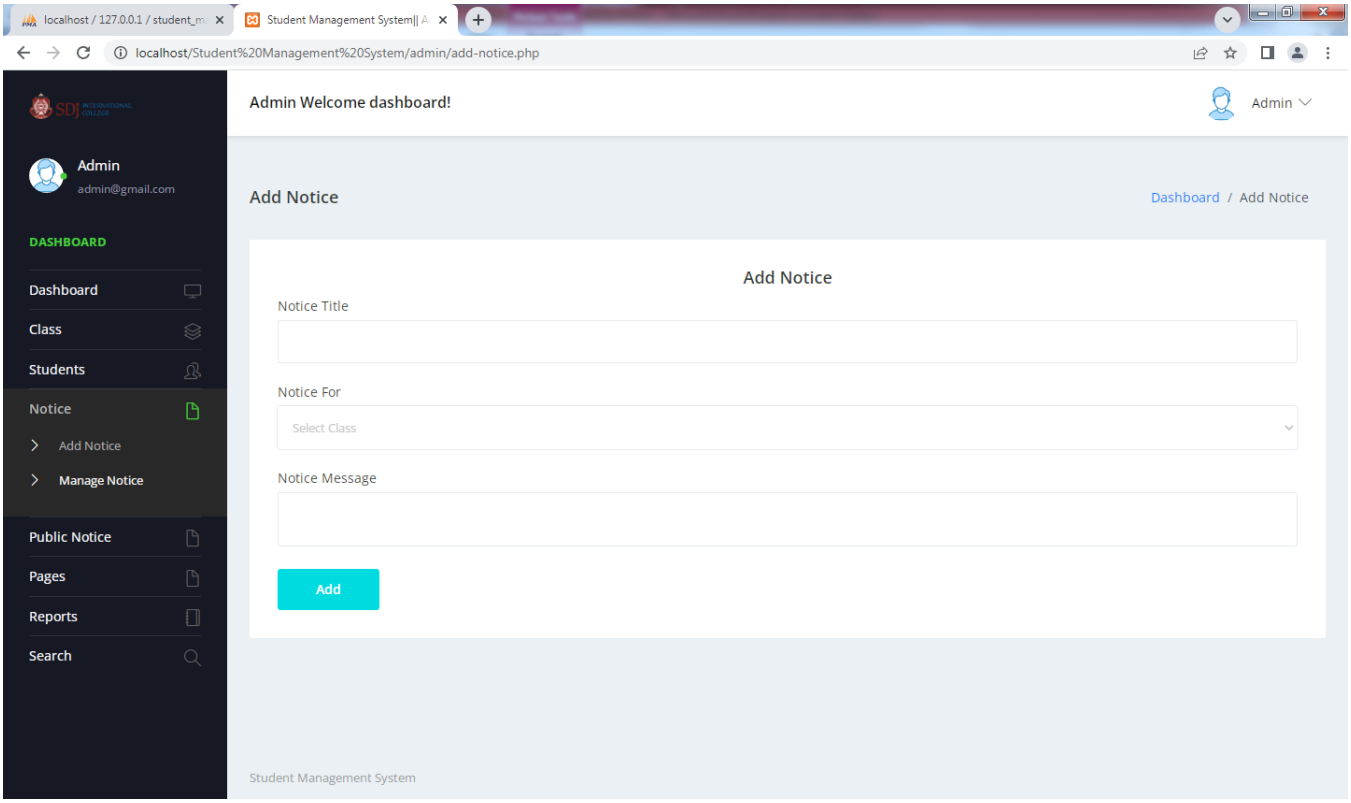


- Manage Student :

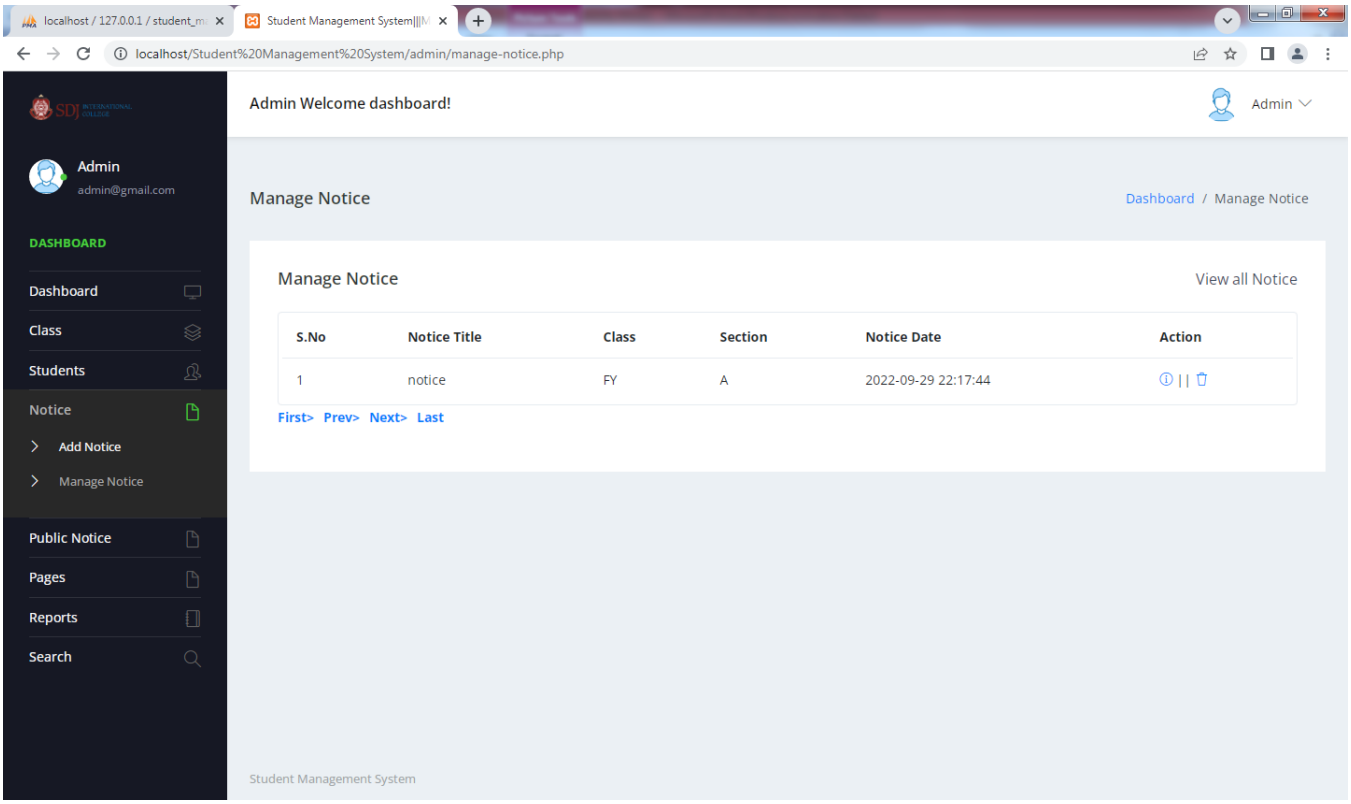


6. Notice Management :

   - Add Notice :

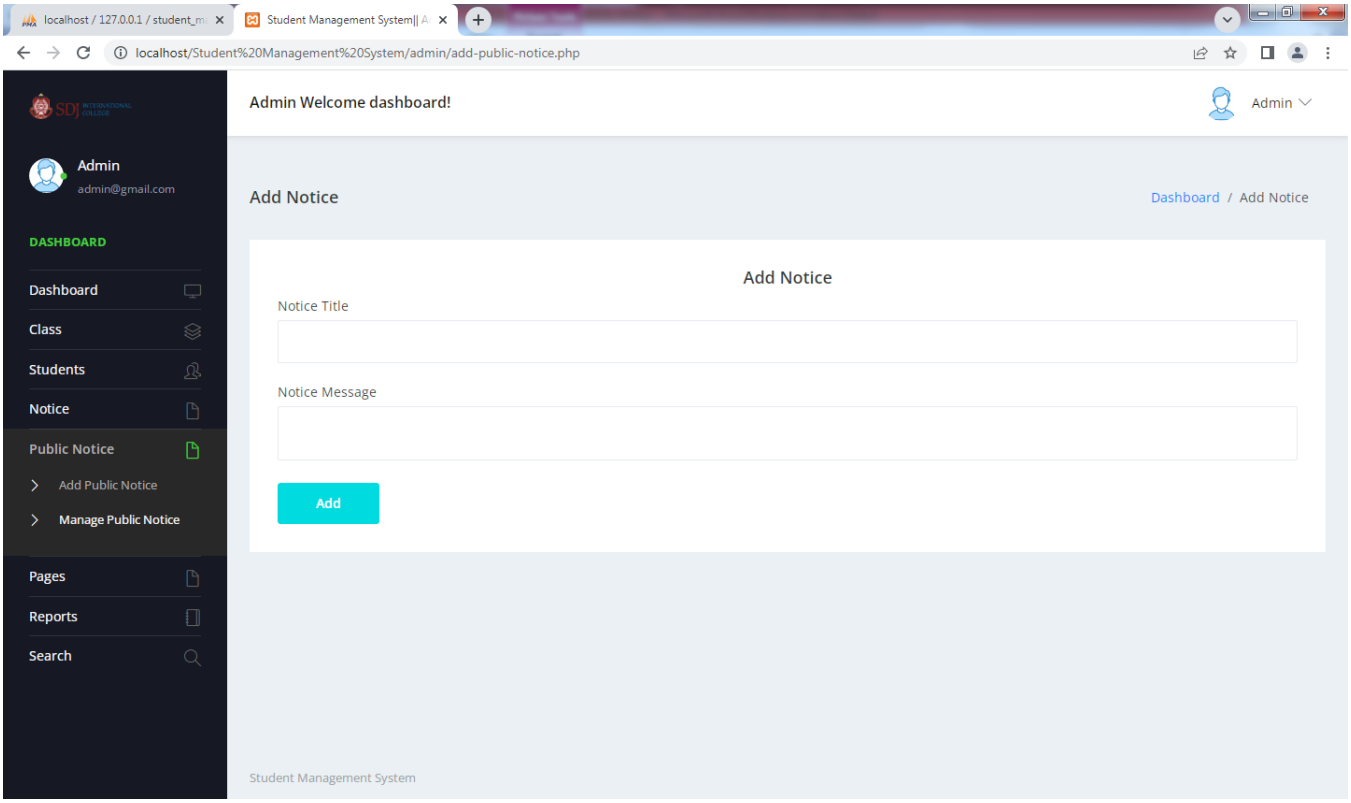


- Manage Notice :

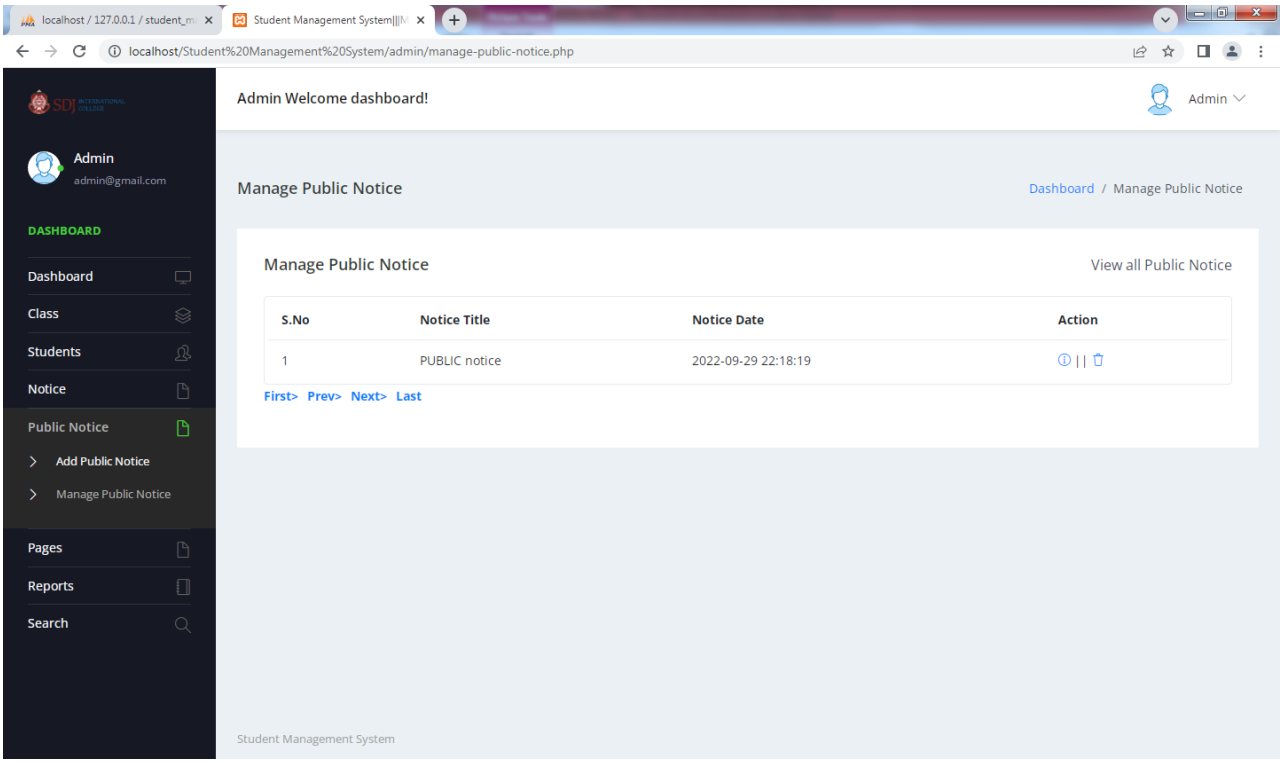


7. Public Notice Management :

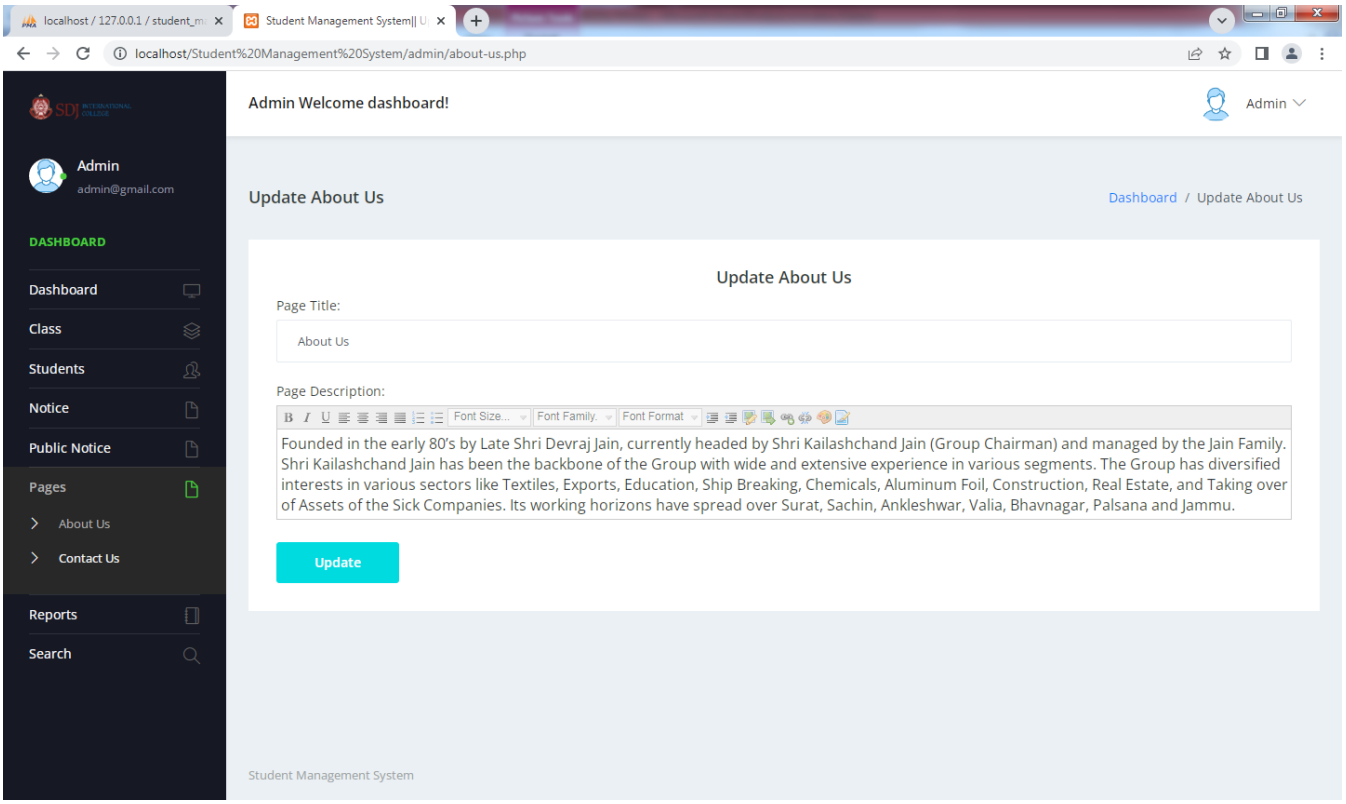
   - Add Public Notice :



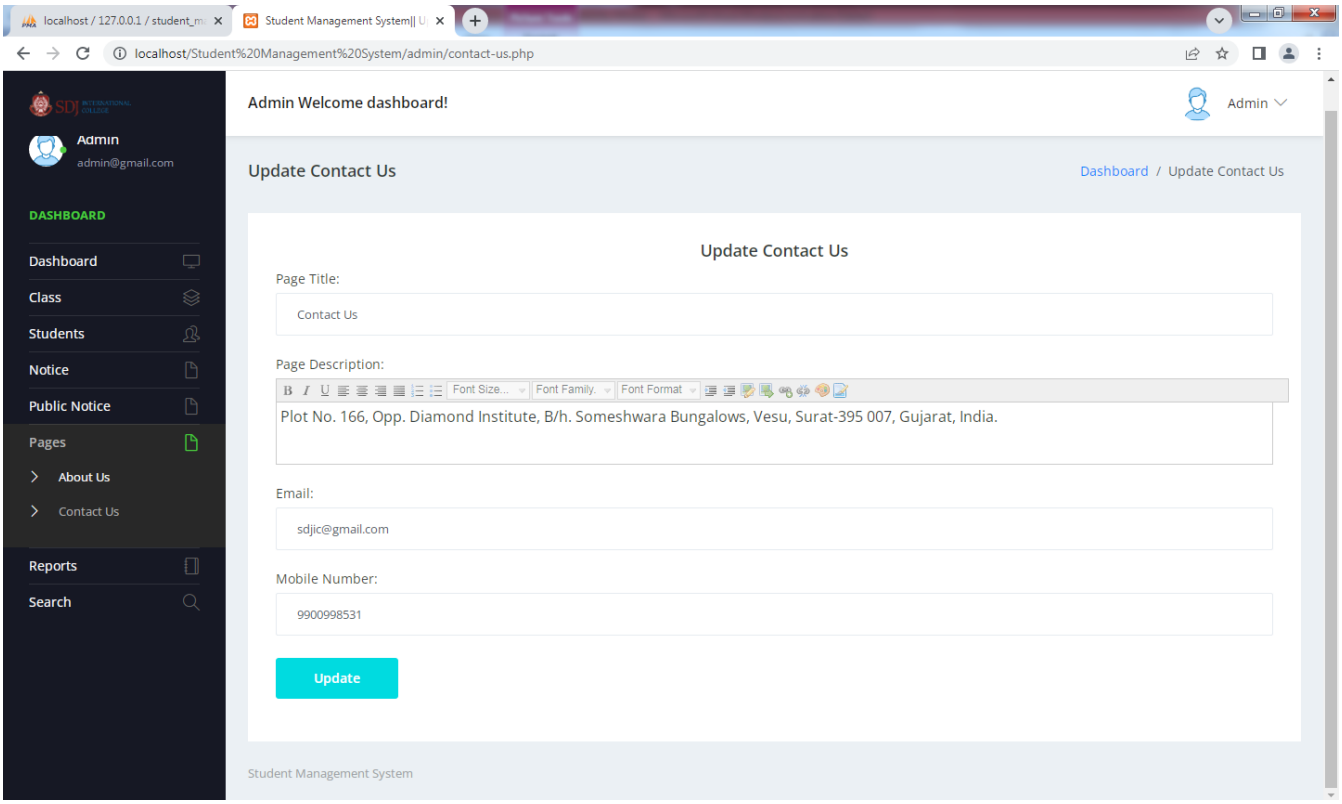
- Manage Public Notice :



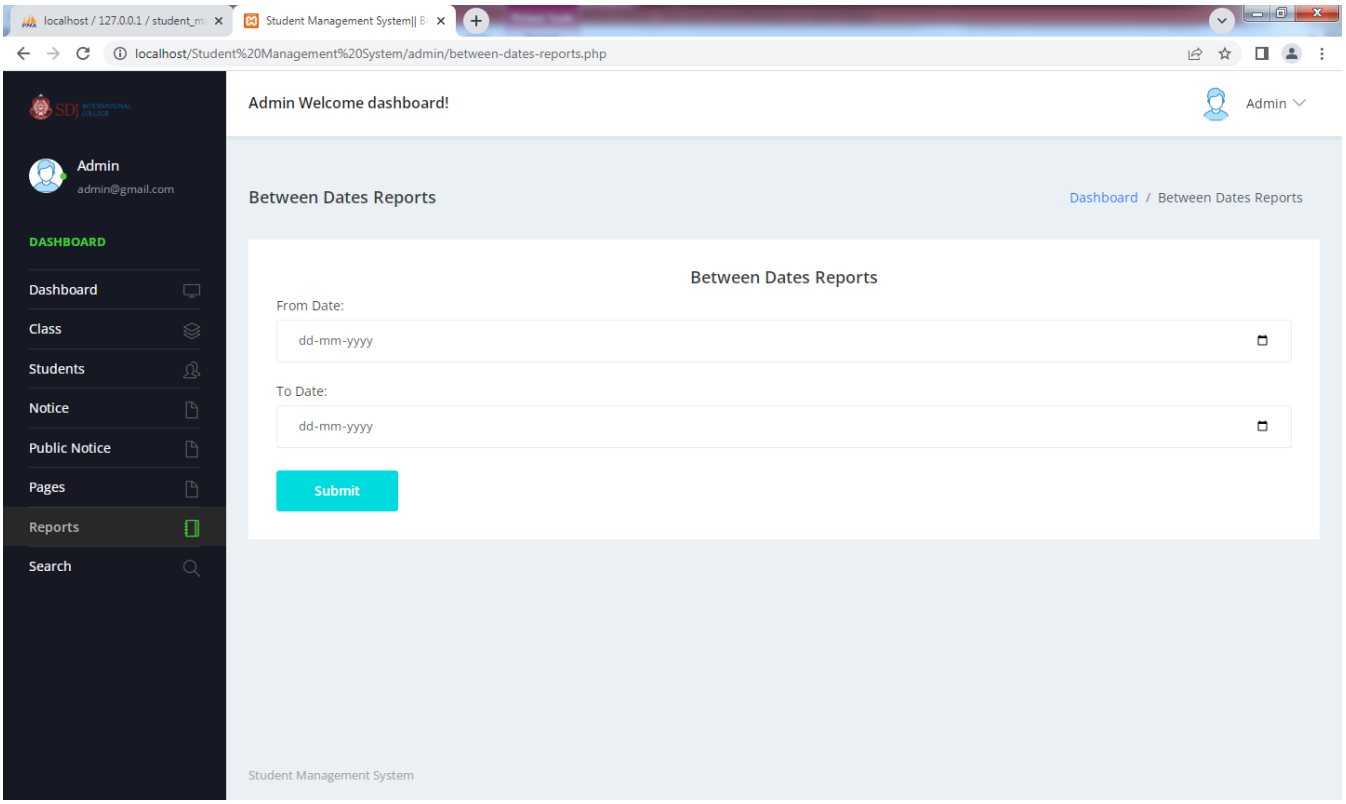
8. About Us page :



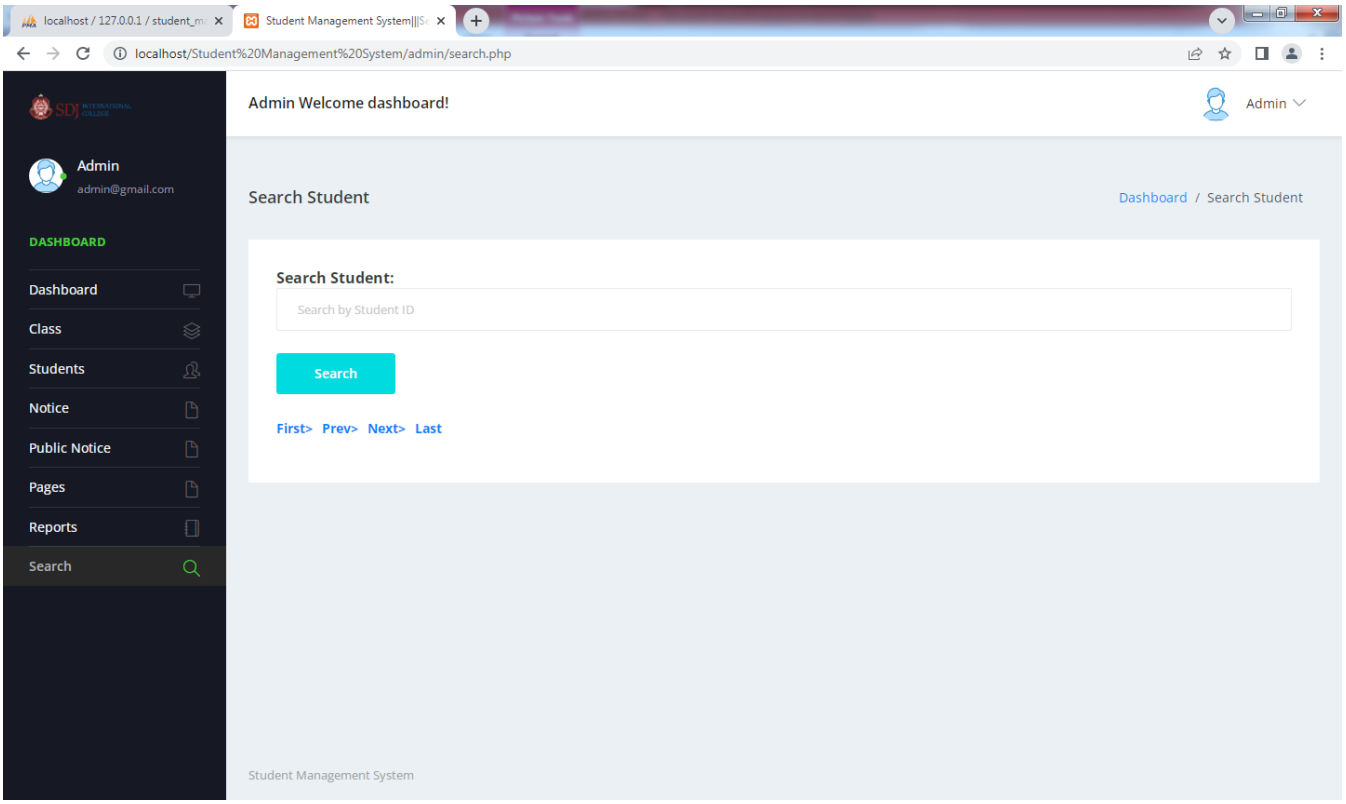
9. Contact Us page :



10. Reports Page :

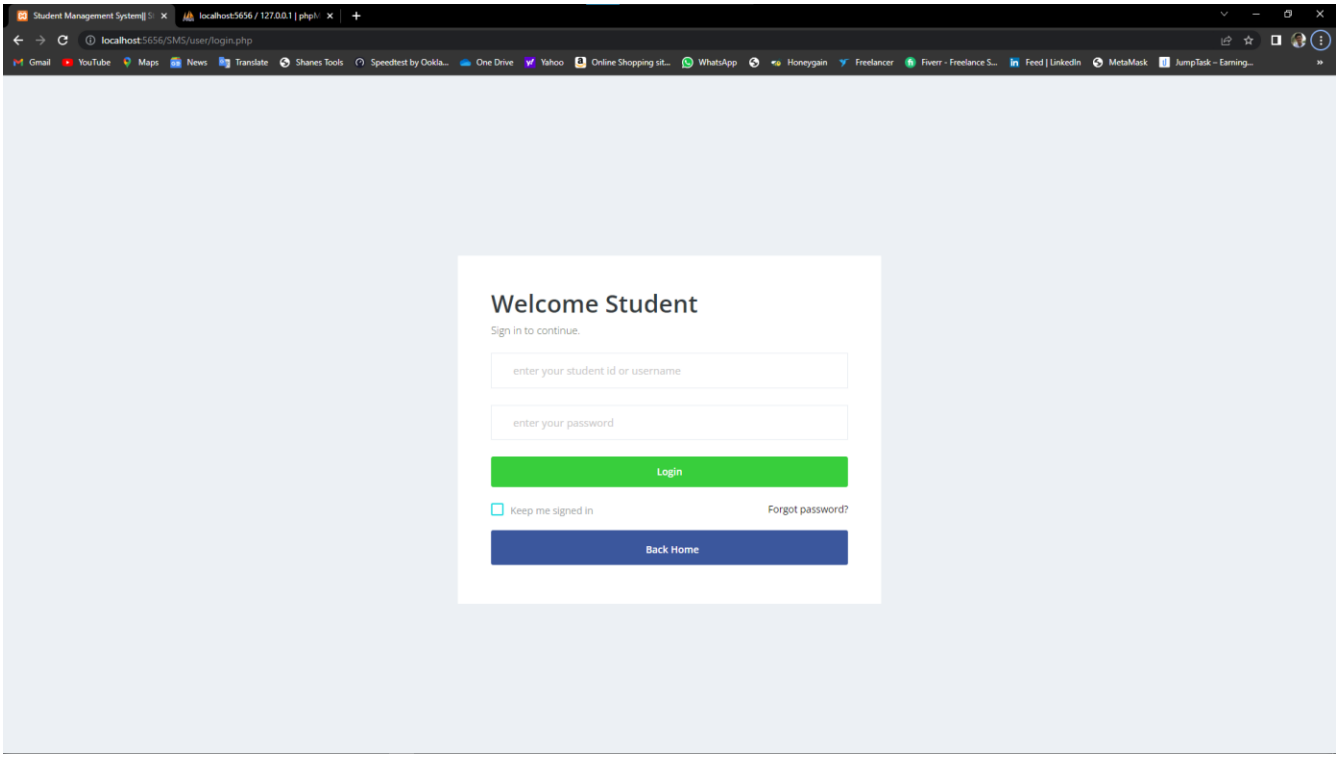


11. Search Student Details Page :

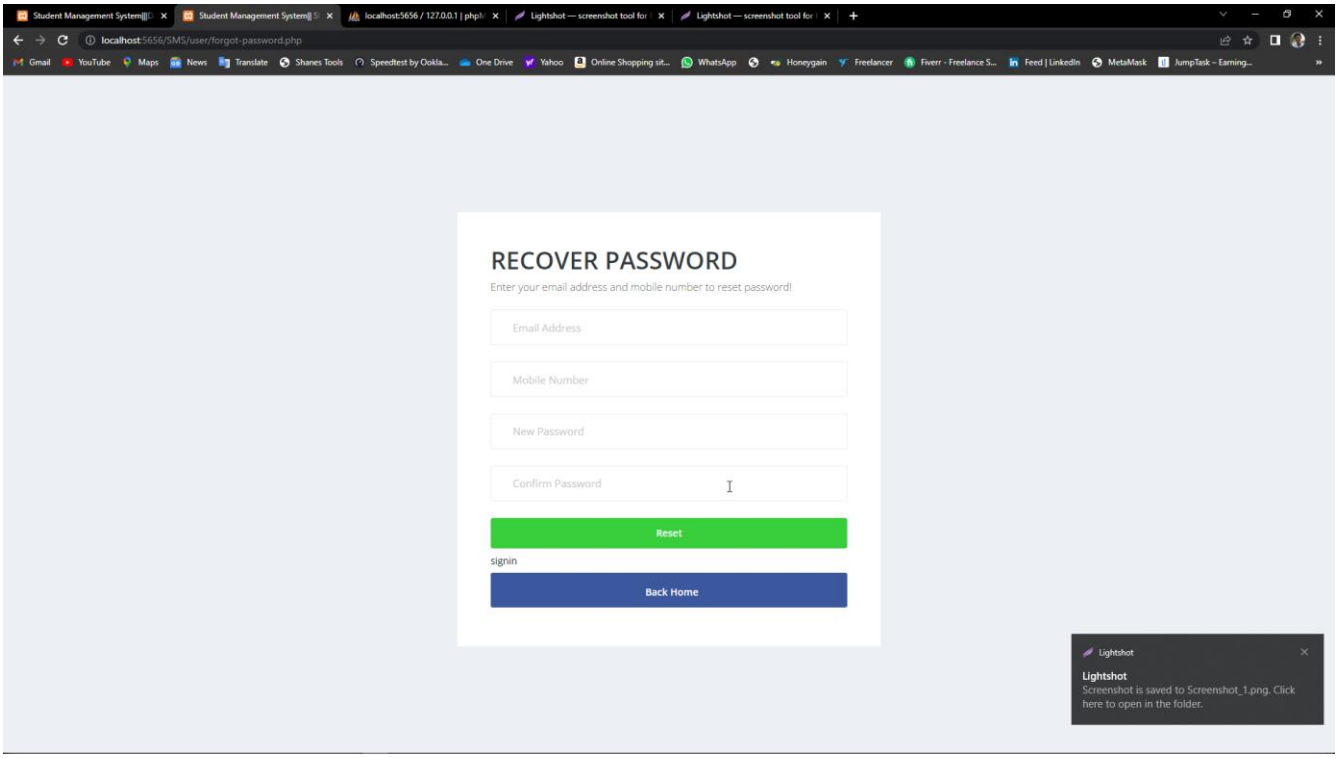


B. Student Side :

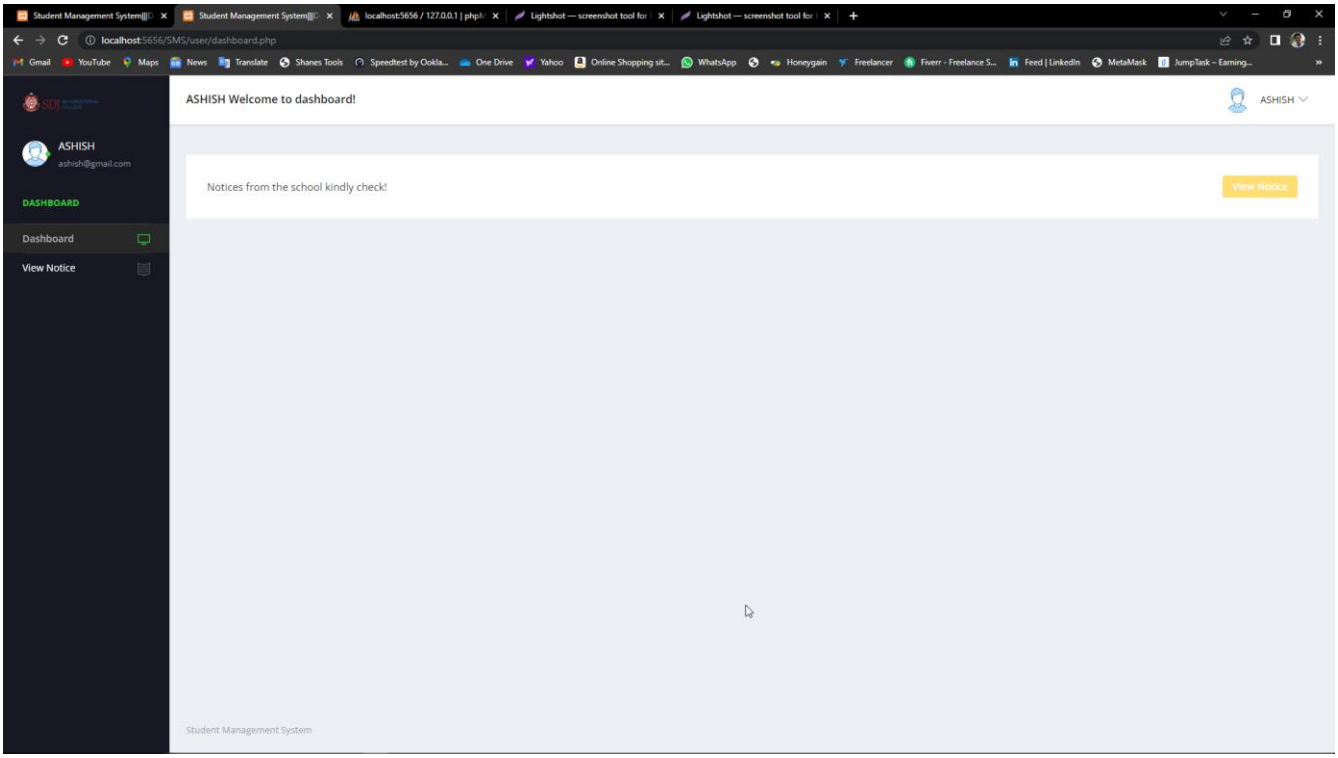
1. Student Login :



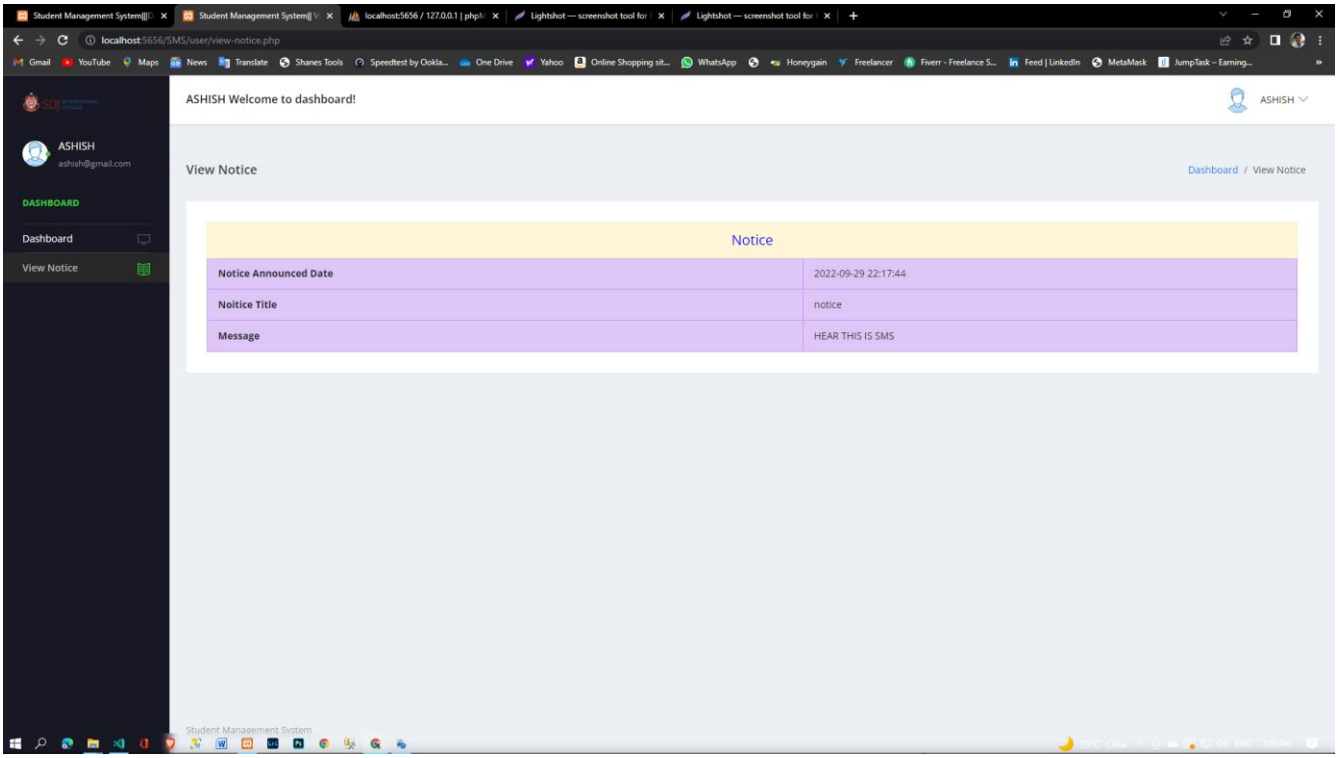
2. Student Recover Password :



3. Student Dashboard :

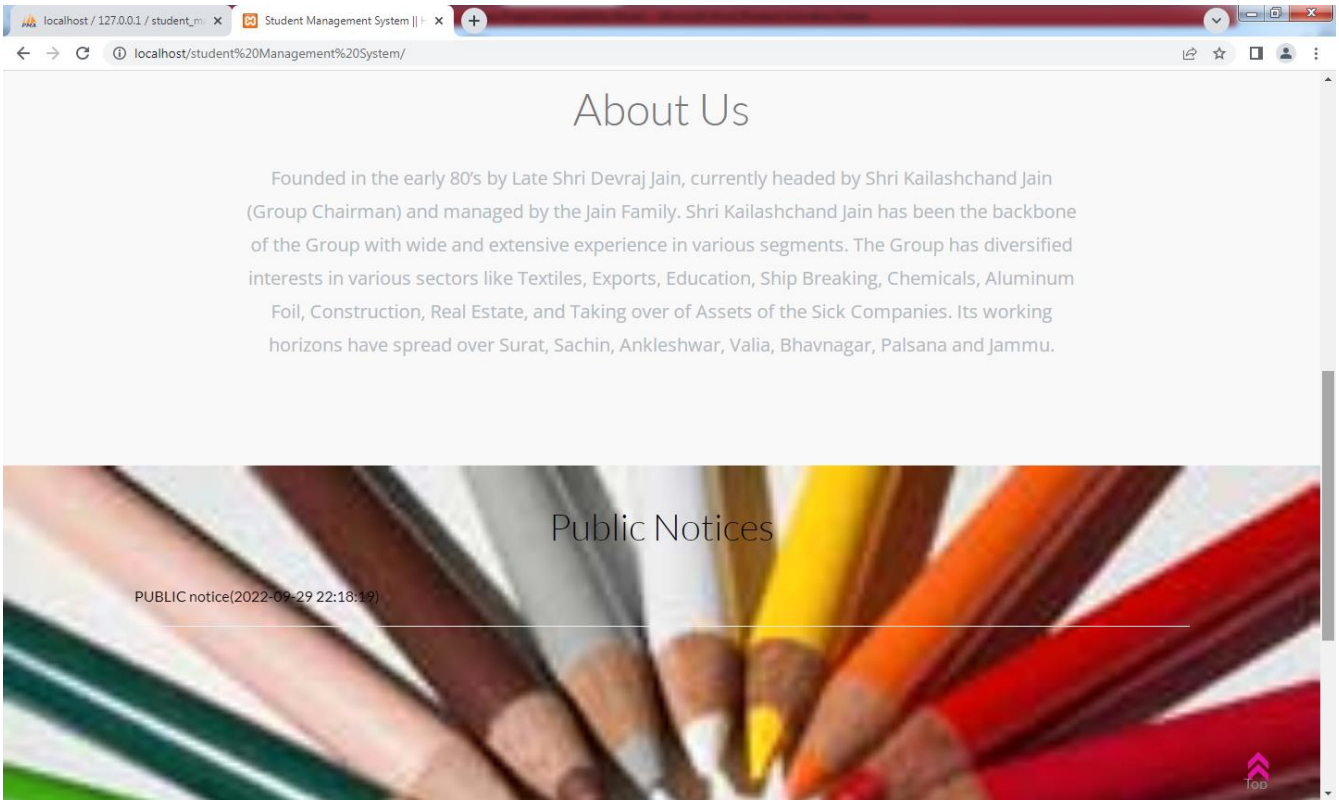


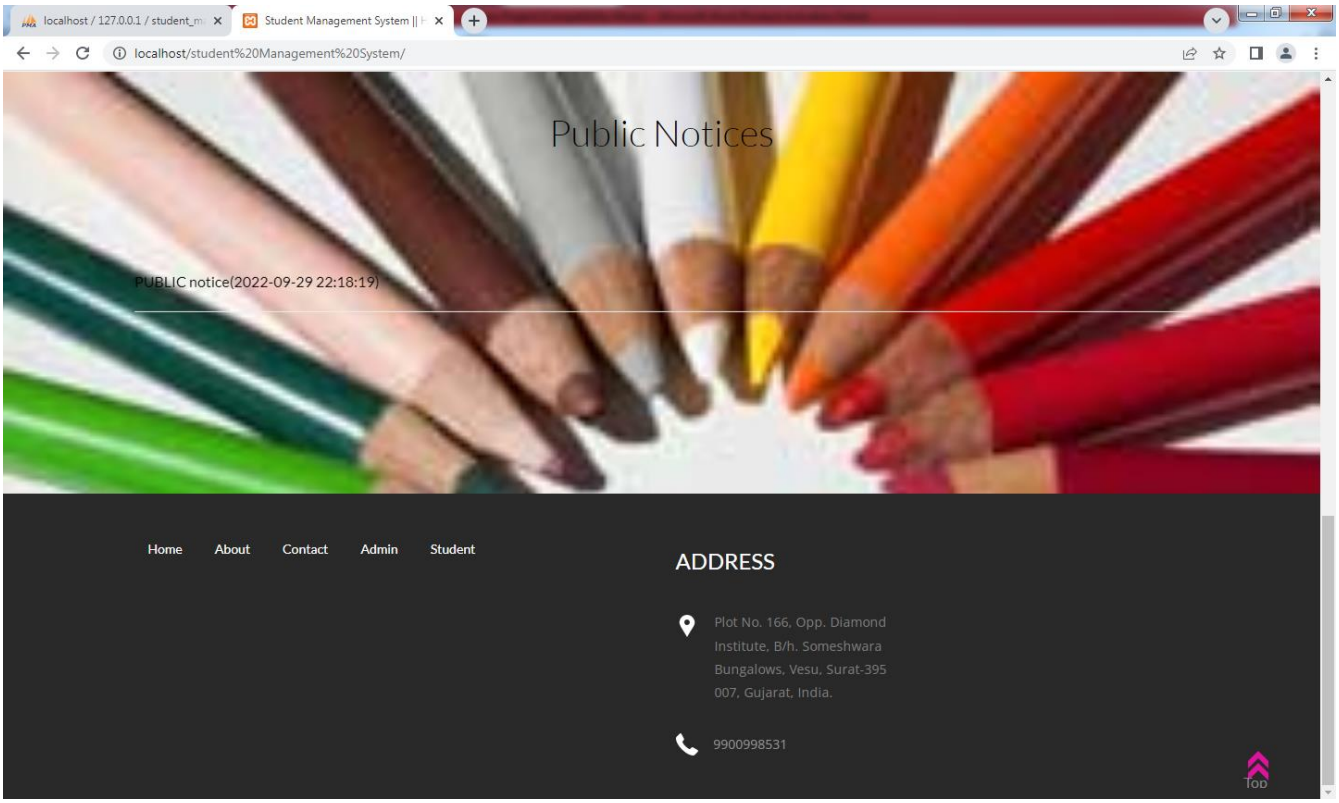
4. Student View Notice :



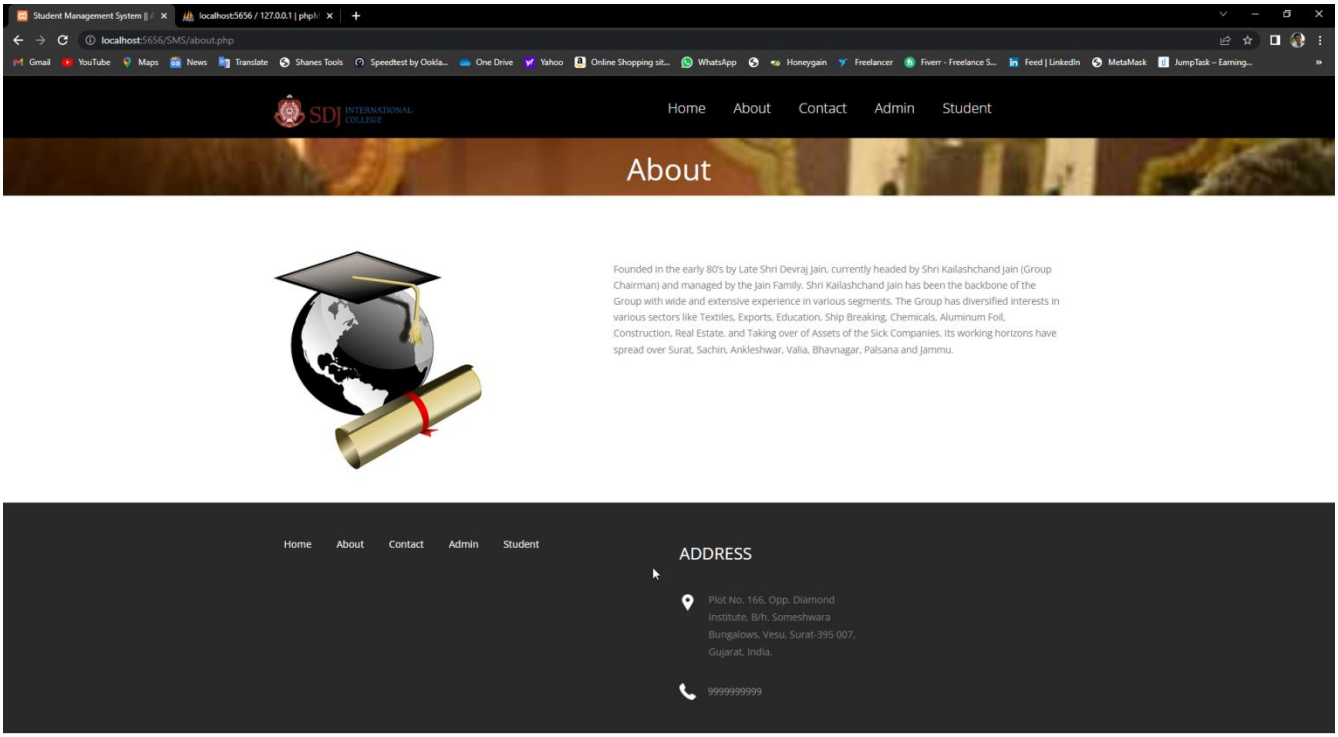
6.4 Output Design

1. Home Page :

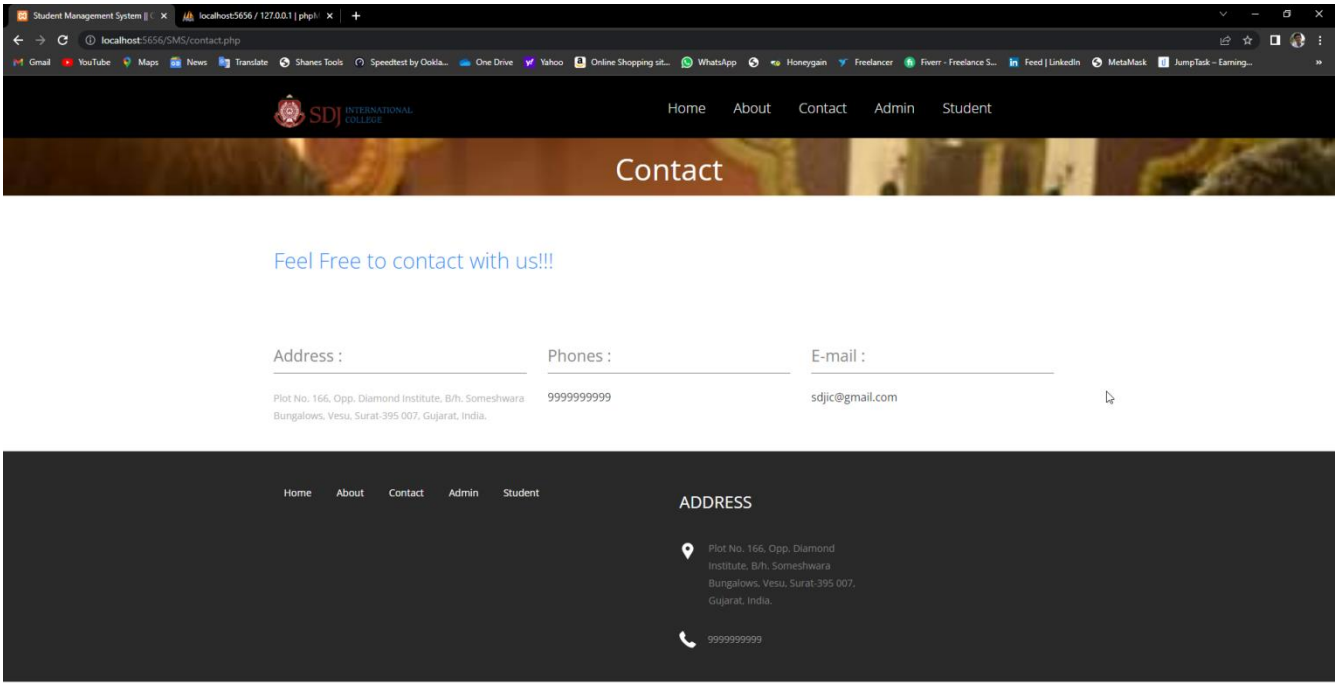



localhost / 127.0.0.1 / student_m
Student Management System ||
localhost/student%20Management%20System/
Public Notices
PUBLIC notice(2022-09-29 22:18:19)
Home
About
Contact
Admin
Student
ADDRESS
Plot No. 166, Opp. Diamond Institute, B/h. Someshwara Bungalows, Vesu, Surat-395 007, Gujarat, India.
9900998531
TOP

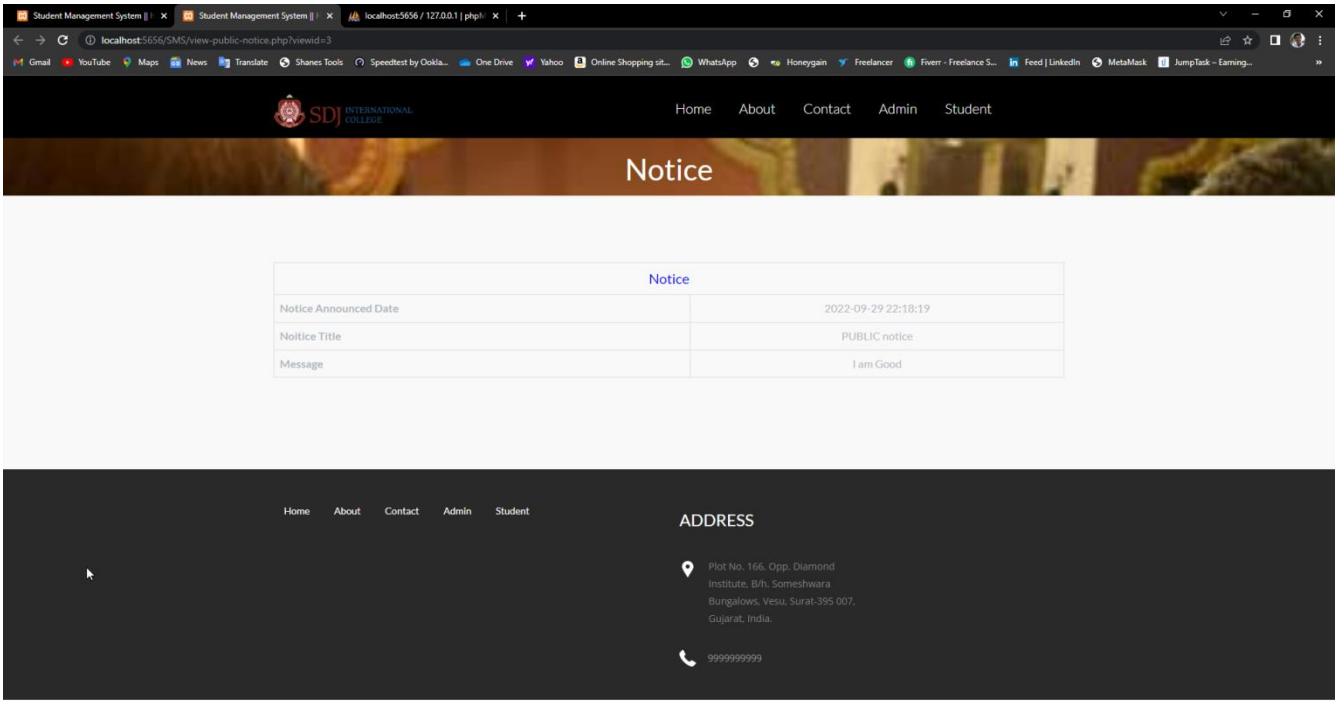
2. About Us Page :



3. Contact Page :



4. Notice View Page :



7. Software Testing

1. Unit Testing

- Test Case For Admin

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Username	Empty	“Please fill out this field”	As Expected
2	Username	Wrong	Invalid Details	As Expected
3	Password	Empty	“Please fill out this field”	As Expected
4	Password	Wrong	Invalid Details	As Expected

- Test Case For Admin Recover Password

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Email	Empty	“Please fill out this field”	As Expected
2	Email	Wrong	“Please enter a part followed by @”	As Expected
3	Mobile Number	Empty	“Please fill out this field”	As Expected
4	Mobile Number	Wrong	Invalid Details	As Expected
5	New Password	Empty	“Please fill out this field”	As Expected
6	New Password	Wrong	Invalid Details	As Expected
7	Confirm Password	Empty	“Please fill out this field”	As Expected
8	Confirm Password	Wrong	Invalid Details	As Expected

- Test Case For Admin Add Class

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Class Name	Empty	“Please fill in this field”	As Expected
2	Section	Empty	“Please select an item in the list”	As Expected

- Test Case for Admin Add Students

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Student Name	Empty	“Please fill in this field”	As Expected
2	Student Email	Empty	“Please enter a part followed by @”	As Expected
3	Student Class	Empty	“Please fill in this field”	As Expected
4	Gender	Empty	“Please select an item in the list”	As Expected
5	Date of Birth	Empty	“Please select an item in the list”	As Expected
6	Student ID	Empty	“Please fill in this field”	As Expected
7	Father's Name	Empty	“Please fill in this field”	As Expected
8	Mother's Name	Empty	“Please fill in this field”	As Expected
9	Contact Number	Empty	“Please fill in this field”	As Expected
10	Alternate Contact Number	Empty	“Please fill in this field”	As Expected
11	Address	Empty	“Please fill in this field”	As Expected
12	User Name	Empty	“Please fill in this field”	As Expected
13	Password	Empty	“Please fill in this field”	As Expected

- Test Case for Admin Add Notice

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Notice Title	Empty	"Please fill in this field"	As Expected
2	Notice For	Empty	"Please select an item in the list"	As Expected
3	Notice Message	Empty	"Please fill in this field"	As Expected

- Test Case for Admin Add Public Notice

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Notice Title	Empty	"Please fill in this field"	As Expected
2	Notice Message	Empty	"Please fill in this field"	As Expected

- Test Case for Admin Between Dates Reports

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	From Date:	Empty	"Please fill in this field"	As Expected
2	To Date:	Empty	"Please fill in this field"	As Expected

- Test Case For Student

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Username	Empty	"Please fill out this field"	As Expected
2	Username	Wrong	Invalid Details	As Expected
3	Password	Empty	"Please fill out this field"	As Expected
4	Password	Wrong	Invalid Details	As Expected

- Test Case For Student Recover Password

Test Id	Test field	Step Execute	Executed Result	Actual Result
1	Email	Empty	“Please fill out this field”	As Expected
2	Email	Wrong	“Please enter a part followed by @”	As Expected
3	Mobile Number	Empty	“Please fill out this field”	As Expected
4	Mobile Number	Wrong	Invalid Details	As Expected
5	New Password	Empty	“Please fill out this field”	As Expected
6	New Password	Wrong	Invalid Details	As Expected
7	Confirm Password	Empty	“Please fill out this field”	As Expected
8	Confirm Password	Wrong	Invalid Details	As Expected

2. Navigational Testing

- Admin

Link	Expected Page	Result Page
admin_login	Admin Login	Admin Login
dashboard	Dashboard Page	Dashboard Page
admin_manage_class	Manage Class Page	Manage Class Page
admin_manage_student	Manage Student Page	Manage Student Page
admin_manage_notice	Manage Notice Page	Manage Notice Page
admin_manage_public_notice	Manage Public Notice Page	Manage Public Notice Page
admin_password_reset	Reset Password Page	Reset Password Page

- Student

Link	Expected Page	Result Page
student_login	Student Login	Student Login
dashboard	Dashboard Page	Dashboard Page
student_password_reset	Reset Password Page	Reset Password Page

3. Environment Testing

- System Testing is a level of software testing where complete & integrated software is tested.
- The purpose of this test is to evaluate the systems compliance with the specifiedrequirements.
- Google Chrome & Firefox browser were considered as testing environment foroperability of Software.

a. Web Server : XAMPP Server.

b. Database : MySQL

c. OS : Windows 10.

d. Browser : Google Chrome/Firefox.

8. Limitations and Future Scope of Enhancements

- Student Management System can be used by educational institutions to maintain their student records easily.
- Achieving this objective is difficult using the manual system as the information is scattered, can be redundant, and collecting relevant information may be very time-consuming.
- All these problems are solved by this project.
- This system helps in maintaining the information of pupils of the organization.
- It can be easily accessed by the manager and kept safe for a long period of time without any changes.
- The project titled “Student Management System” is developed using Python Tkinter as front end and MYSQL database in back end to computerize the process of management of student records.
- This project covers only the basic features required.

9. References

- https://www.codewithharry.com/
- https://www.w3schools.com/
- https://phpgurukul.com/
- https://getbootstrap.com/
- https://www.codewithc.com/