

**PROJECT REPORT**  
**ON**  
**SOLO: Web-Based Educational Resources**



**DAFFODIL INTERNATIONAL UNIVERSITY**  
**DHAKA, BANGLADESH**  
**OCTOBER 2021**

**SOLO: Web-Based Educational Resources**

**BY**

**SHAHINUR ISLAM**

**ID: 171-15-9140**

**MEHEDI HASAN SHAHED**

**ID: 181-15-966**

**MOHAMMAD FAHIM FIROZ**

**ID: 181-15-978**

This Report Presented in Partial Fulfillment of the Requirements for the  
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

**Ms. Naznin Sultana**

Assistant Professor

Department of CSE

Daffodil International University

Co-Supervised By

**Mohammad Monirul Islam**

Senior Lecturer

Department of CSE

Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY**

**DHAKA, BANGLADESH**

**OCTOBER 2021**

## **APPROVAL**

This Project titled “**SOLO: Web-Based Educational Resources**”, submitted by SHAHINUR ISLAM ID: 171-15-9140, MEHEDI HASAN SHAHED ID: 181-15-966 and MOHAMMAD FAHIM FIROZ ID: 181-15-978 to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on 30<sup>th</sup> October 2021.

## **BOARD OF EXAMINERS**

---

**(Name)**

**Chairman**

**Designation**

Department of CSE <sup>[Font-12]</sup>

Faculty of Science & Information Technology

Daffodil International University

---

**(Name)**

**Internal Examiner**

**Designation**

Department of CSE

Faculty of Science & Information Technology

Daffodil International University

---

**(Name)**

**External Examiner**

**Designation**

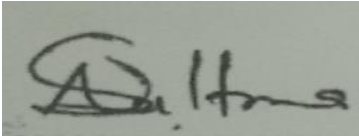
Department of -----

Jahangirnagar University

## DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Ms. Naznin Sultana, Assistant Professor, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

### Supervised by:



---

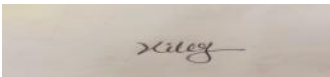
**Ms. Naznin Sultana**  
Assistant Professor  
Department of CSE  
Daffodil International University

### Co-Supervised by:

---

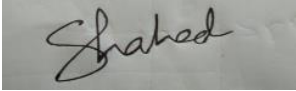
**Mohammad Monirul Islam**  
Senior Lecturer  
Department of CSE  
Daffodil International University

### Submitted by:



---

**(Shahinur Islam)**  
ID: 171-15-9140  
Department of CSE  
Daffodil International University



---

**(Mehedi Hasan Shahed)**

ID: 181-15-966

Department of CSE

Daffodil International University



---

**(Mohammad Fahim Firoz)**

ID: 181-15-978

Department of CSE

Daffodil International University

## ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty Allah for His divine blessing makes us possible to complete the final year project/internship successfully.

We really grateful and wish our profound our indebtedness to **Ms. Naznin Sultana, Assistant Professor**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Research and Development*” to carry out this project. Her endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Professor Dr. Touhid Bhuiyan**, and Head, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

## **ABSTRACT**

We are going to develop a web application named “SOLO: Web-Based Educational Resources”. It’s an online platform for an offline educational institution. It provides the institute to run almost all main functionalities of study purposes online. Sometimes, unexpected things happen which causes offline education interrupted. Like last year (2020), every institution including the education system had been closed their physical activities because of the COVID-19 pandemic. This project provides services in these kinds of circumstances. There are fixed courses and subjects under some departments. Teachers can maintain these courses enrolling their students. A new department, subject, and course can be added anytime. This project maintains communication between teachers and students. Teachers and students can discuss with each other the topic of lectures about given courses via a social group. Teachers can share course content with their students. These contents can be any type of files like pdf, document, audio, or video. Students can view and if wish, can download them. They can take quizzes through this system. Classes can be held online through video meetings. Recording of these classes is also available for the students. They can submit the assignment on this system. This system also helps the institute to maintain its functionality online parallel along with offline which will save time and expenses for all. Also, will be easy for both the teacher and student.

## TABLE OF CONTENTS

<b>CONTENTS</b>	<b>PAGE</b>
Board of Examiners	iii
Declaration	iv
Acknowledgement	v
Abstract	vi
<b>CHAPTERS</b>	
<b>CHAPTER 1: INTRODUCTION</b>	<b>1-3</b>
1.1 Introduction	1
1.2 Motivation	1
1.3 Objective	2
1.4 Expected Outcome	3
1.5 Report Layout	3
<b>CHAPTER: BACKGROUND</b>	<b>4-6</b>
2.1 Introduction	4
2.2 Related Works	5
2.3 Comparative Studies	5
2.4 Scope of the problems	5
2.5 Challenges	6



<b>CHAPTER 3: REQUIREMENT SPECIFICATION</b>	<b>6-9</b>
3.1 Requirement Collection and Analysis	6
3.2 Logical Data Model	7
3.3 Design Requirement	9
3.4 Server Requirement	9
<b>CHAPTER 4: DESIGN SPECIFICAITON</b>	<b>10-55</b>
4.1 Front-end Design	10
4.2 Back-End Design	26
4.2.1 Back-End for Admin	27
4.2.2 Back-End for Course	27
4.2.3 Back-End for Department	28
4.2.4 Back-End for File upload	29
4.2.5 Back End for Student	30
4.2.6 Back-End for Subject	31
4.2.7 Back-End for Subject Wise Student	32
4.2.8 Back End for Teacher	33
4.2.9 Back-End for Teacher wise Student	33
4.2.10 Back-End for Teacher Subject	34
4.2.11 Back-End for Class	35
4.3 Code Segment	36
4.3.1 Class Code	36

4.3.2 Class of Student Code	37
4.3.3 Data base Connect with Php	38
4.3.4 Delete class	38
4.3.5 Delete File code	39
4.3.6 Delete Student	40
4.3.7 Delete Teacher	40
4.3.8 Delete By Teacher a Student	41
4.3.9 Student Home code	41
4.3.10 Student_ class	42
4.3.11 Student.php code	43
4.2.12 Student portal Code	43
4.3.13 Student Modal	44
4.3.14 Teacher Add Class Code	45
4.3.15 Teacher Add Student	46
4.3.16 Teacher Add Subject code	47
4.3.17 Teacher Home code	47
4.3.18 Teacher.php code	48
4.3.19 Teacher Subject code	49
4.3.20 Upload Save php	49
4.3.21 Upload File Code	50
4.3.22 Admin Add Student	51
4.3.23 Logout Code	51

4.3.24 Contact Code	52
4.3.25 Header Code	53
4.3.26 Index	53
4.3.27 Responsive View	54
<b>CHAPTER 5: TESTING AND IMPLEMENTATION</b>	<b>56-59</b>
5.1 Testing and Implementation	56
5.1.1 Unit Testing	56
5.1.2 Integration Testing	56
5.1.3 Functional Testing	57
5.1.4 System Testing	57
5.1.5 Usability Testing	57
5.2 Test Result	57
5.2.1 Unit Test Result	58
5.2.2 Integration Test Result	58
5.2.3 Functional Test Result	58
5.2.4 System Test Result	59
5.2.5 Usability Test Result	59
<b>CHAPTER 6: CONCLUSION AND FUTURE SCOPE</b>	<b>59-60</b>
6.1 Conclusion & Limitations	59
6.2 Future scope	60



## List of Figures

Figure 3.2.1: ER Diagram of the system	8
Figure 4.1.1: Home page of the website	10
Figure 4.1.2: Team of teachers	11
Figure 4.1.3: Services provided	12
Figure 4.1.4: Goal of the project	12
Figure 4.1.5: Student progress ratio	13
Figure 4.1.6 Admin Home page	14
Figure 4.1.7: File uploaded	14
Figure 4.1.8: Admin add course	15
Figure 4.1.9: Admin add subject page	16
Figure 4.1.10: Department Section of admin	17
Figure 4.1.11: Admin's Student Section	17
Figure 4.1.12: Student Registration form	18
Figure 4.1.13: Teacher Section of Admin	19
Figure 4.1.14: Teacher registration	20
Figure 4.1.15: Teacher Home Page	20
Figure 4.1.16: Teacher's class	21
Figure 4.1.17: Teacher's Subject section	22
Figure 4.1.18: Teacher's Student section	23
Figure 4.1.19: Student's classes	24
Figure 4.1.20: Student's class content	25
Figure 4.1.21: Contact option	26
Figure 4.2.1 Back-end Admin	27
Figure 4.2.2 Back-End Course	28
Figure 4.2.3 Back-End Department	29
Figure 4.2.4 Back-End File upload	30
Figure 4.2.5 Student Database	31
Figure 4.2.6 Back-End Subject	32
Figure 4.2.7 Back-End SWS	32
Figure 4.2.7 Teacher Database	33
Figure 4.2.9 Teacher Wise Student Database	34
Figure 4.2.10 Back-End Teacher Subject	35
Figure 4.2.11 Back-End Class	36
Figure 4.3.1 Student Class Code	37
Figure 4.3.2 Student php Code	37
Figure 4.3.3 Database connect with php Code	38
Figure 4.3.4 Delete Code	39

Figure 4.3.5 Delete file Code	39
Figure 4.3.6 Delete student Code	40
Figure 4.3.7 Delete teacher Code	40
Figure 4.3.8: Delete teacher _student Code	41
Figure 4.3.9: Student home php Code	42
Figure 4.3.10: Student class php Code	42
Figure 4.3.3: Student php Code	43
Figure 4.3.12: Student portal Code	44
Figure 4.3.13: Student Modal Code	45
Figure 4.3.14: Teacher Add class Code	46
Figure 4.3.15: Teacher Add Student Code	46
Figure 4.3.16: Teacher Add Subject Code	47
Figure 4.3.17: Teacher Home Code	48
Figure 4.3.18: Teacher. Php Code	48
Figure 4.3.19: Teacher subject Code	49
Figure 4.3.20: Upload File save Code	50
Figure 4.3.21: Teacher Upload File Code	50
Figure 4.3.22: Admin Add Student Code	51
Figure 4.3.23: Logout Code	52
Figure 4.3.24: Contact Code	52
Figure 4.3.25: header file Code	53
Figure 4.3.26: Index html Code	54
Figure 4.3.27.1: Responsive View Code	55
Figure 4.3.27.2: Responsive View Code	55

# CHAPTER 1

## INTRODUCTION

### **1.1 Introduction:**

Education is one of the basic needs of a human being. Without this none can develop himself. For proper and formative learning an institution is the best choice. Most of the educational institutes give their services physically. If somehow, for a short time physical connection becomes banned, all of the activities of these institutions become paused. But if it happens for a long time it becomes a great problem. In 2020, the COVID-19 corona pandemic had caused this problem all over the world. Doors of all the schools over the world had been closed for this pandemic. Then, only those who had an online backup could keep their pace unchanged. If there is a platform where students can communicate with their teachers and classmates, they can take tests, teachers can share class contents such as audio, video, doc or pdf files and have class conducted through video meeting it will serve all basic needs to continue study process. If they discuss with each other, then learning will be more easy & effective online. This project will serve all these requirements.

### **1.2 Motivation:**

When the pandemic situation had started and our online classes started then there were many lacking in the online platform. As we are students, partially take classes of our students for pocket money. But we could not find out a good way or system how to take classes on an online platform in this pandemic situation. Then an idea came to our mind that why we did not create a project in this unique platform of an online academy where the administration can connect teachers & students. Students can join live classes, discuss with each other, submit their quizzes, assignments & the system will be enlarged with such helpful resources. Then we talked to each other with our group members and we agreed to make such type of system as project & we start our implementation. Although it was a new concept for us, this motivation led us to develop a project like this.

### **1.3 Objective:**

It's painful to have a thirst for knowing something but having no resources on that. One of our objectives is to satisfy this hunger for knowledge. The main objective is to provide different types of educational resources on different topics. It will be starting from class one to the university level. Another objective is to make a platform for those who are experts in some topic but their experience is not in use to help others.

Features:

- Offered subject wise courses
- Every course is taken by a fixed teacher
- Students are eligible by the admin
- Students can join live classes
- Students can discuss with the teacher & other class members
- Students can give quiz & online examination
- Students can view and download course items
- Teachers can add students to their courses
- A teacher can upload a file
- A teacher can create a meeting & discussion group



- A new department, subject, and courses can be added by the administration
- It is an Administration based system so the administration can be added or removed everything.

#### **1.4 Expected Outcome:**

The student will be very pleased after the successful launch of the project. Our students will be very excited & glad to use this system. This is mainly a student-friendly web base system where they will find their desired system. All of the design & feature was made by thinking about the student advantage. By providing resources on the fixed topic and making complex things clear then goodness will be spread to the learners. Students can use it as like game. So education will be more enjoyable & efficient. By creating this awesome project we will make our debt in the development career. This project will be a reference if the project is built successfully.

#### **1.5 Report Layout:**

In our report, there are six chapters. They are Introduction, Background, Requirement Specification, Design Specification, Implementation and Testing, Conclusion, and future scope. The contents of these six chapters are described shortly:

1. In chapter one, we have given the introduction, objective, motivation, expected outcome of our project that a simple description of our project can be gained.
2. In chapter two, we focused on the detailed background of our project as well as related work. Also, we have added comparative studies and challenges.
3. In chapter three, we have presented the detailed diagram of the Requirement collection and analysis, Logical Data Model, Design Requirements, and Server Requirements.

4. In chapter four we have detailed Design Specifications of the front end, back end interaction, and UX.
5. In chapter five we have described the testing and implementation process of our project.
6. In the last chapter, chapter six we have discussed the conclusion and future scope of the project.

## **CHAPTER 2**

### **BACKGROUND**

#### **2.1 Introduction:**

The year 2020 was historical. This year the pandemic COVID-19 coronavirus has stopped everything included education and closed the doors of schools. It impeded students so much for more than one year. But some institutions like our university had not been stopped by the pandemic. The Daffodil International University has run its activities on the Blended Learning Center (BLC) at full pace. The project “SOLO: Web-Based Educational Resources” is a web-based system kind of that platform. This system is an online center of an offline education platform. Here various kinds of resources are provided such as audio, video, images, pdf, documents, hand notes, etc. Through this system, students can be connected to the teachers and their classmates by chatting and social networking system. Teachers can take classes on zoom, Google meets, or other mediums through this system. In this system, courses are fixed and students are enrolled by authorization. Authority can select the best teachers for the selected courses. To keep the pace of study flawless, a quiz system is available for every lesson. It is developed in a way that if offline interconnection becomes interrupted somehow the education will not stop. And obviously, the parallel of the offline education process will help keep records smartly and save time and communication gap.

## **2.2 Related Works:**

On the internet, there are many web-based systems quite similar to this. Most of them are school management systems or college management systems or both in one place. Ours' is a website that will cover both school and college. And if we want we can use it for the university level too. One of the similar web base systems is the Blended Learning Center of Daffodil International University. Along with these, there are so many web-based educational resource systems available online. Some of them we can mention as

- (i) Unique School Management System [1]
- (ii) Eskooly [2]
- (iii) Robi 10-minute School [3]

## **2.3 Comparative Studies:**

We had to brainstorm with the students who had been fallen into difficulties during the COVID-19 corona pandemic. We had to find out exactly which of the problems they had faced. The class has been closed and for health caution, getting in physical touch is restricted so that communication became limited. For lockdown, most of the people left cities that distance becomes a great matter. A precise platform was required to share notes, documents, content, etc. One of the most important things to speed up the learning process is test-taking. So, we have included a quiz system for every lesson. And all other problems that could be faced which are described above have been solved in our system.

## **2.4 Scope of the problems:**

In our system, we won't take enormous user traffic at the first moment because of cloud storage and bandwidth. If 5000+ users are using this app at the same time, then the server is going to be down and an application not responding (ANR) will occur. The size of files can be a problem that it could not be more than 20 MB. Primarily it has been seen that the media link of live video classes was not working which is solved. Also, the linking of the communication platform was having some problems we have solved.

## **2.5 Challenges:**

To develop this system we have to face some challenges as it's a kind of user-friendly system. Its' management is difficult. Admin has to do almost all the maintaining and registration types of tasks. So the admin section will be quite tough if there is more user traffic. Though this project is simple on the student site, the teachers have to be well trained. As with the flow of time user requirement changes so updating is a must for any online application. It is hard to update the site. Here, we have to maintain many users and their information, so privacy is an important topic here. Another challenge of this project will be getting popularity. Like many people, the project reaches so many the new demand from this project will be discovered. So then, we can make it awesome and more excited.

## **CHAPTER 3 REQUIREMENT SPECIFICATION**

### **3.1 Requirement Collection and Analysis:**

Requirement engineering which is also called requirement analysis is the method of determining user requirements for new or updated systems. It is often referred to as requirements gathering and requirement capturing in the sense of software engineering. The tasks that go into determining the needs or conditions to meet for a new or altered product or project, taking into account the potentially conflicting requirements of various stakeholders are all included in requirement analysis. We have collected requirements from our classmates and other students around us. We have discussed this with the teachers of various institutions. We gathered information about the world environment of the present education system also. We find out that what problems the students are facing, how they manage these problems, what their daily tasks are hampered. The teachers had to take classes online from home. So, now they don't have a whiteboard and marker to explain to the students. They needed to take attendance. One main problem was students' not being present at school and for these students become inattentive to study. To hold them up in study quizzes or class tests was a must. At this time, students couldn't meet with their teacher and classmates physically. So any information gathering, sharing notes, or any discussion was hampered. For this, a platform was needed where all of them can meet. As the education system become online-based, it had many dependencies such as electricity, internet, devices, etc. So, resources needed to be stored somewhere. These requirements we had collected from several sources.

### **3.2 Logical Data Model:**

A logical data model is a diagram that shows how data is processed in a database. This diagram is known as the E-R diagram because it includes all entities and relationships between tables.

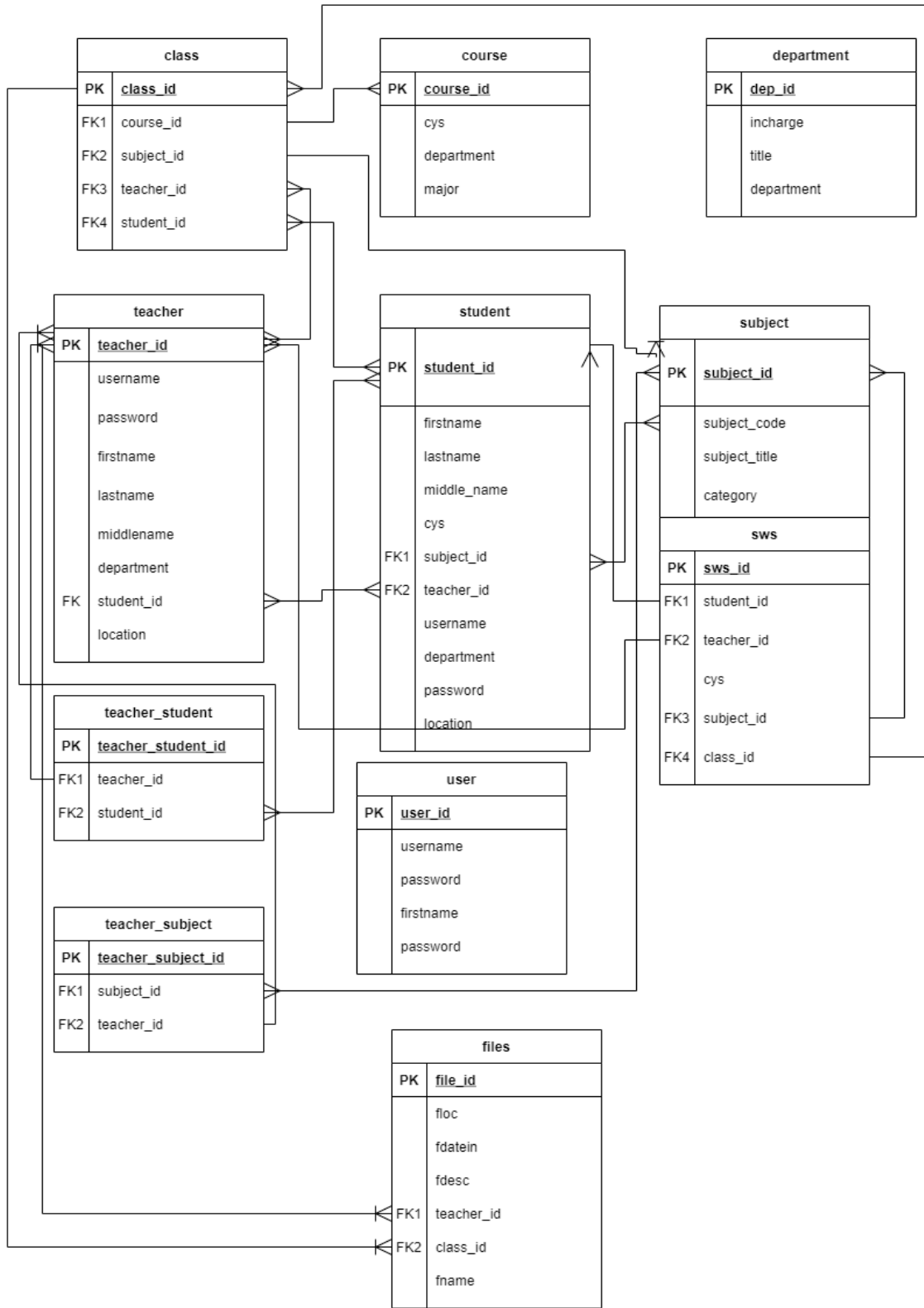


Figure 3.2.1: ER Diagram of the system

### **3.3 Design Requirement**

The specification of the design of the step is designing an application. Since all the processes are dependent on the design of the requirement. Following are the user-specific requirement of the system:

- Admin panel
- Admin log in
- Student/teacher log in
- Course list
- Department
- Class
- Subject list
- Attempt Quiz
- Join Live Classes
- Group Discussion
- File Upload

### **3.4 Server Requirement**

To deploy this system server need the following system.

- 1GB RAM
- 1 CPU Core
- 25GB SSD Storage
- 2 TB Transfer
- 40 Gbit Network in
- 125 Mbit Network Out

# CHAPTER 4

## DESIGN SPECIFICATION

### 4.1 Front-end Design:

The front-end design is the most crucial part of a system to develop. Because it is the presentation layer where users interact with the system. Through this part, the user could communicate with each other and have service from the platform. It must be easy to work with and have to be user-friendly. So we tried to make it as simple as it could be.

The following figure 4.1.1 shows the homepage of our system. In this page there is a navigation to view all of the features and also the login option for the students.

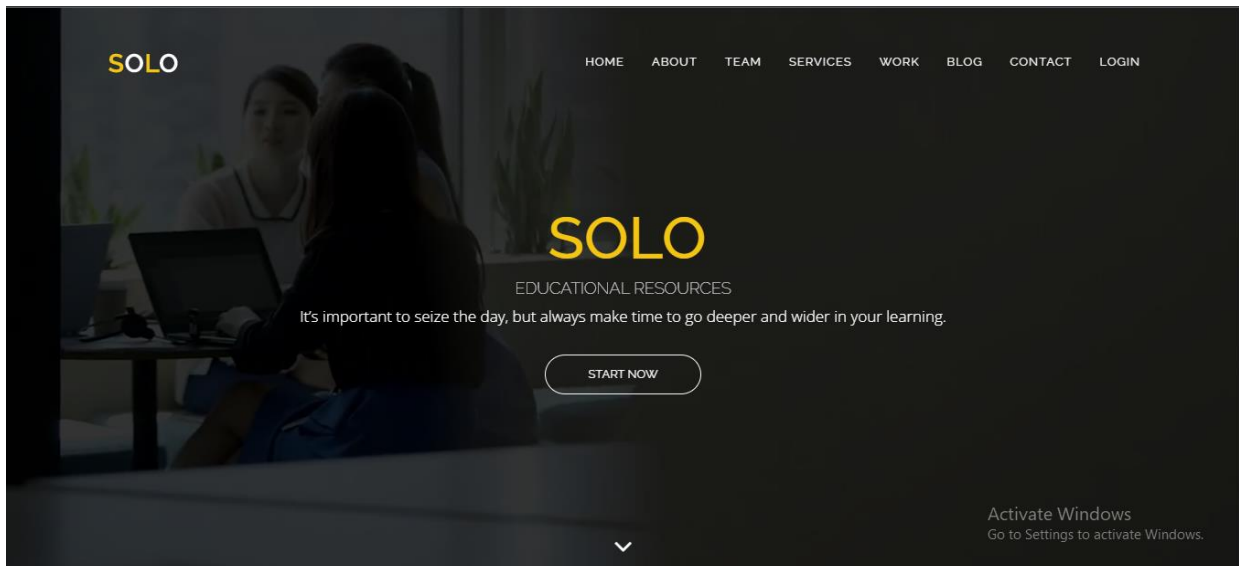


Figure 4.1.1: Home page of the website



This system depends on expert teachers. So visitors can view the teachers, and if they hover on any teacher's image then they can see which courses the teachers are taking. Figure 4.1.2 below is about the team of teachers.

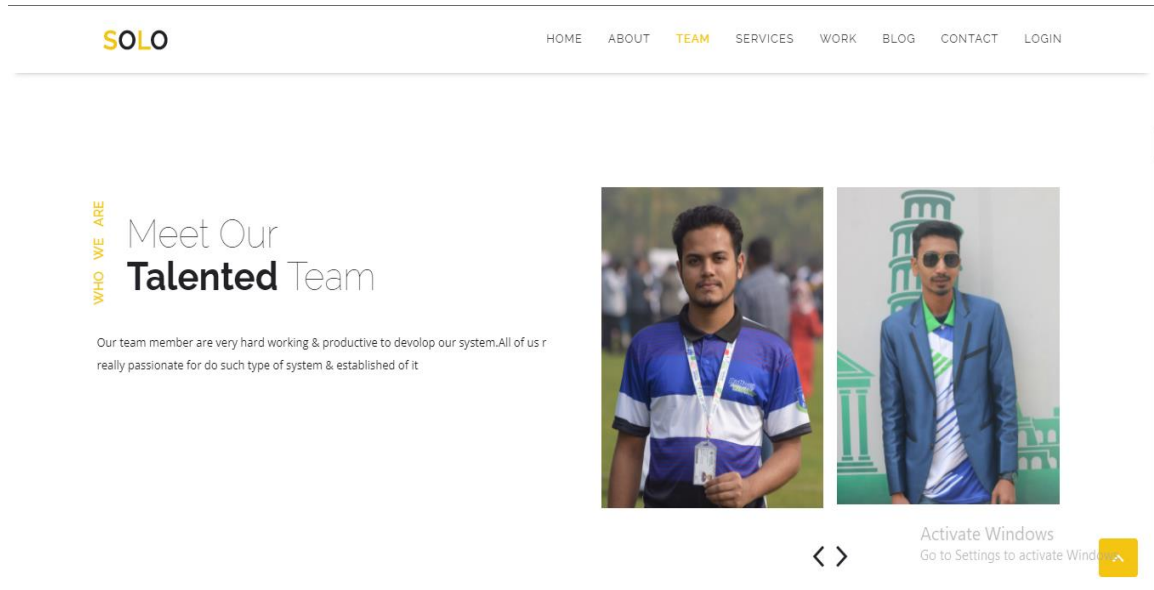


Figure 4.1.2: Team of teachers

Figure 4.1.3 below shows the services we give through this system. This page will shortly describe what types of resources students can get from the system. Also, how their communication will build up and what types of courses this system provides.

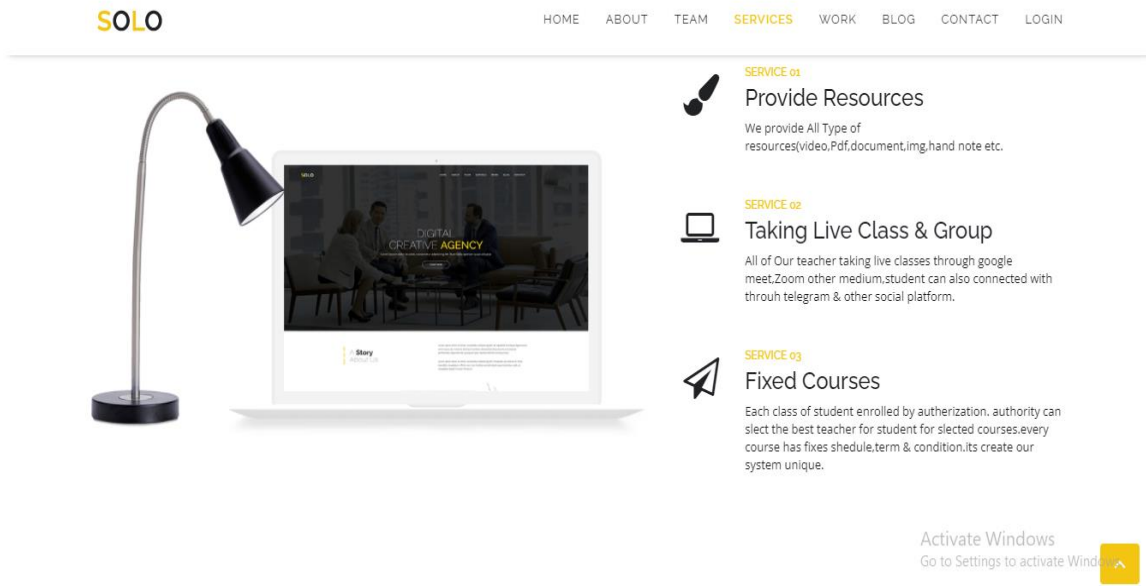


Figure 4.1.3: Services provided

Figure 4.1.4 presents below the page where the goal of this project is described. It contains Mission, vision, and passion of the project.

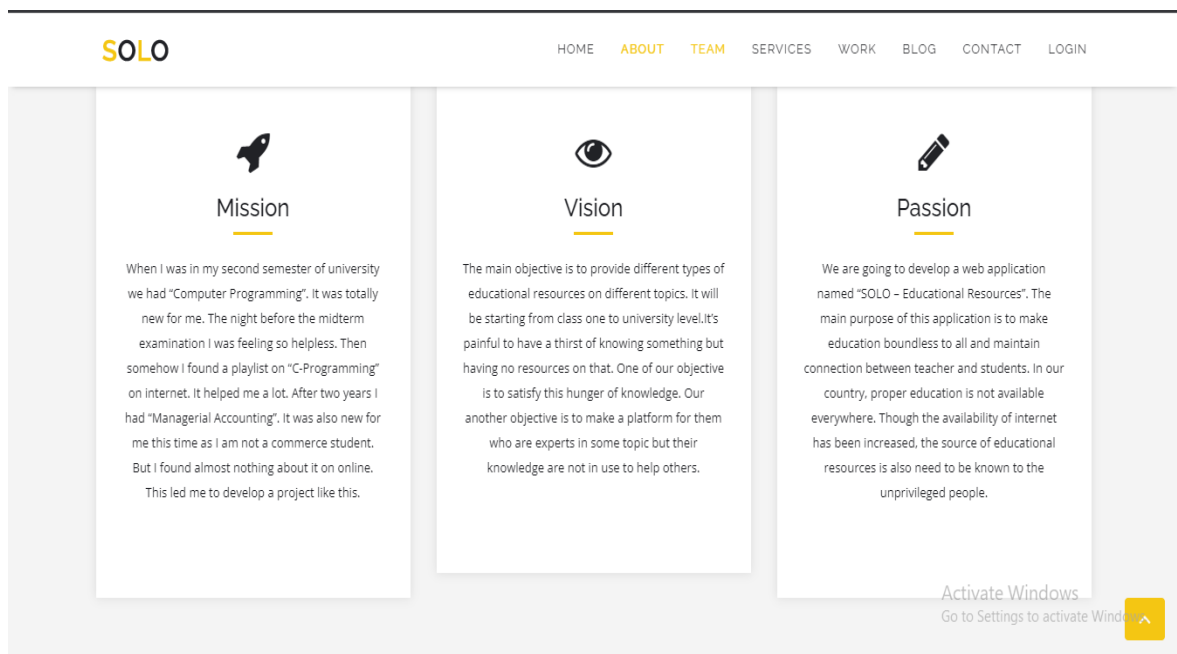


Figure 4.1.4: Goal of the project

Figure 4.1.5 Student Ratio can be identified as a progress report. The percentage ratio of students' progress in every class will be shown like the figure below:

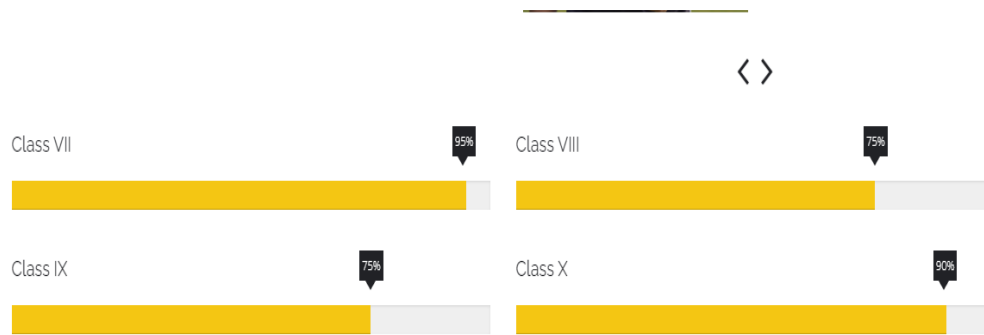


Figure 4.1.5: Student progress ratio

This project is dependent on the admin for maintenance. So the admin has the largest features. Figure 4.1.6 below shows the admin's homepage where all of the maintenances features are presented. From different features, the admin can do various work. Here, the admin can work with file, course, subject, department, student, teacher, and user.



Figure 4.1.6 Admin Home page

Figure 4.1.7 shows below the File feature of the admin panel. Here admin can view the files that are uploaded by the teachers with the date and time of upload. Admin can also delete a file from the list.

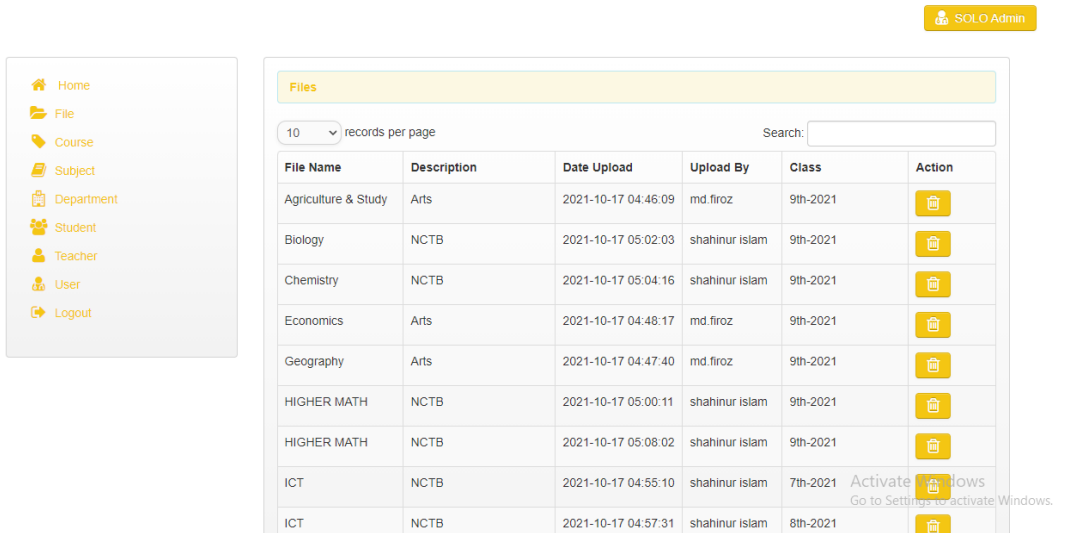


Figure 4.1.7: File uploaded

One of the important tasks of the admin is to add courses to the system. By clicking on the “Add Course” admin can add new classes to the definite department. Figure 4.1.8 below shows that panel of course

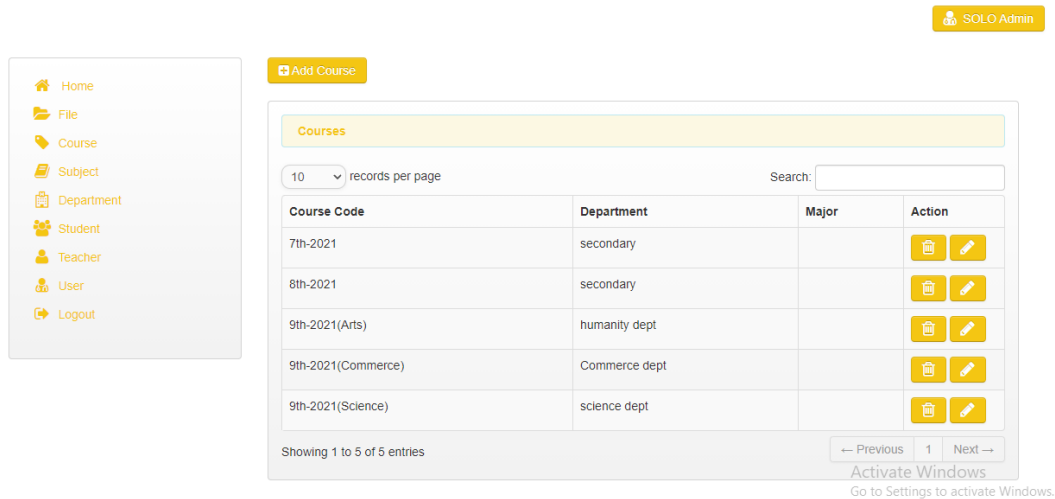


Figure 4.1.8: Admin add course

Another task of the admin is to add subjects to different classes. Only these subjects are offered to the students. Here the information of subjects such as subject code, their title, and category is displayed. Admin can also add new subjects in this place and if wants can delete or edit existing subjects. The process is shown in figure 4.1.9 below.

SOLO Admin

Add Subject

Subjects

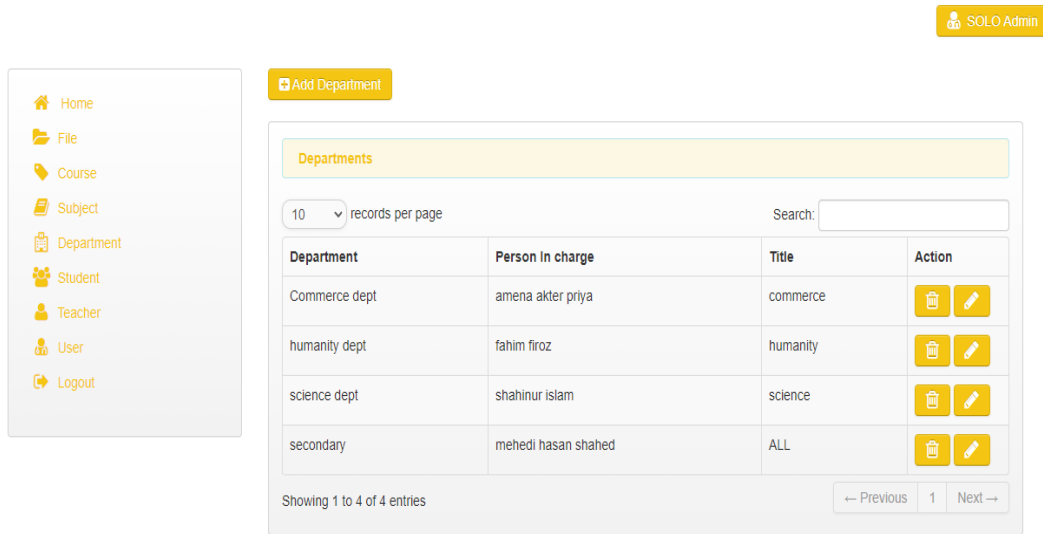
10 records per page Search:

Subject Code	Subject Title	Category	Action
accounting9th	Accounting	Major	
agriculture9th	Agricultural Education	Major	
biology9th	Biology	Major	
business9th	Business Venture	Major	
chemistry9th	Chemistry	Major	
economy9th	Economy	Major	
eng#7th	English Grammer-7	Major	
eng7th	English-7	Major	

Activation Windows  
Go to Settings to activate Windows.

Figure 4.1.9: Admin add subject page

The next feature of the admin is maintaining the department. This feature provides the admin with the information of departments. Here an admin can view the detailed information of existing departments including departments' names, the person in charge of that department, and the title. A new department can be opened any time. The existing department is also needed to be updated now and then. All of these tasks can be done in this section. Figure 4.1.10 shows the department section of the admin.



Activate Windows  
Go to Settings to activate Windows.

Figure 4.1.10: Department Section of admin

One of the most important tasks of the admin is to manage students' information. In the student section, the admin can view, delete or update students' information. Figure 4.1.11 shows below the student page of the admin.

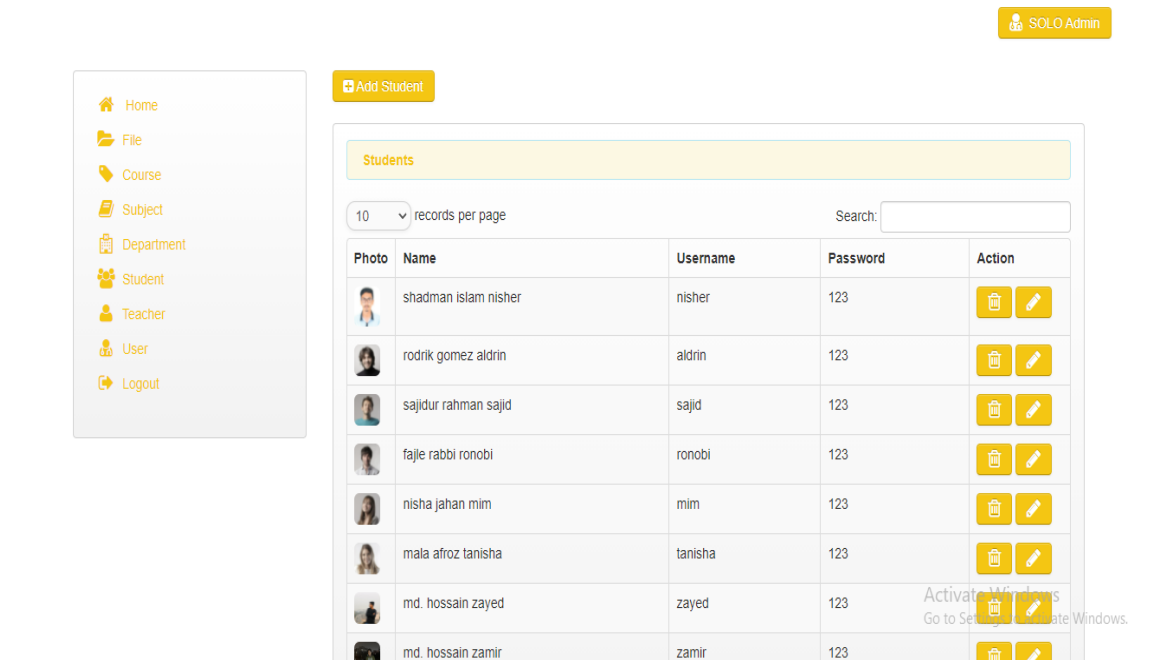
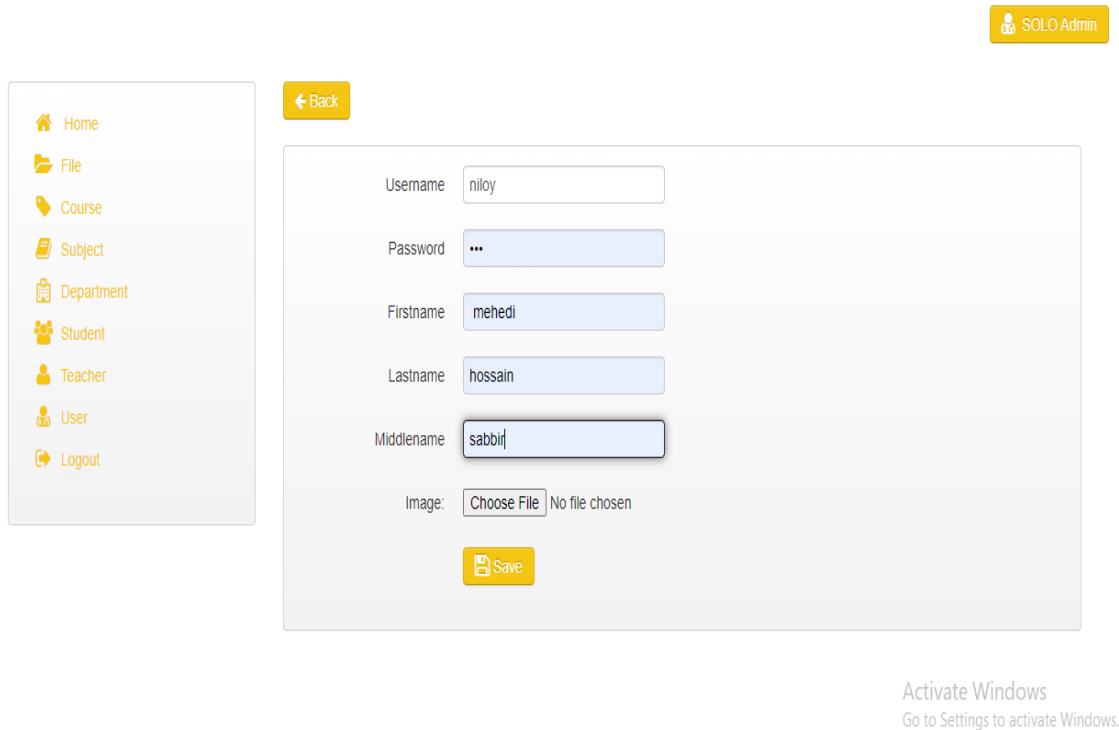


Figure 4.1.11: Admin's Student Section

Following the page displayed in figure 4.1.11 admin can add a new student to the system. Clicking Add student button admin will have a form like the figure 4.1.12 below. Here admin has to give a username, password, first name, last name, middle name, and an image of the student. Here, the registration of a student is done.



The screenshot shows a web application interface for adding a new student. In the top right corner, there is a yellow button labeled "SOLO Admin". On the left side, there is a vertical sidebar menu with icons and labels for: Home, File, Course, Subject, Department, Student, Teacher, User, and Logout. The main content area features a "Back" button and a registration form. The form contains the following fields and values:

- Username: niloy
- Password: (masked with three dots)
- Firstname: mehed
- Lastname: hossain
- Middlename: sabbir
- Image: Choose File No file chosen

At the bottom of the form is a yellow "Save" button. In the bottom right corner of the page, there is a watermark that reads "Activate Windows Go to Settings to activate Windows."

Figure 4.1.12: Student Registration form



Maintaining the information of teachers is done in the “Teacher” section of the admin. Here the admin can view the teachers’ username, password, their name, and department, and importantly the Telegram channel link which will be used as the forum where teacher and student can communicate. Figure 4.1.13 below shows the Teacher section of the admin.

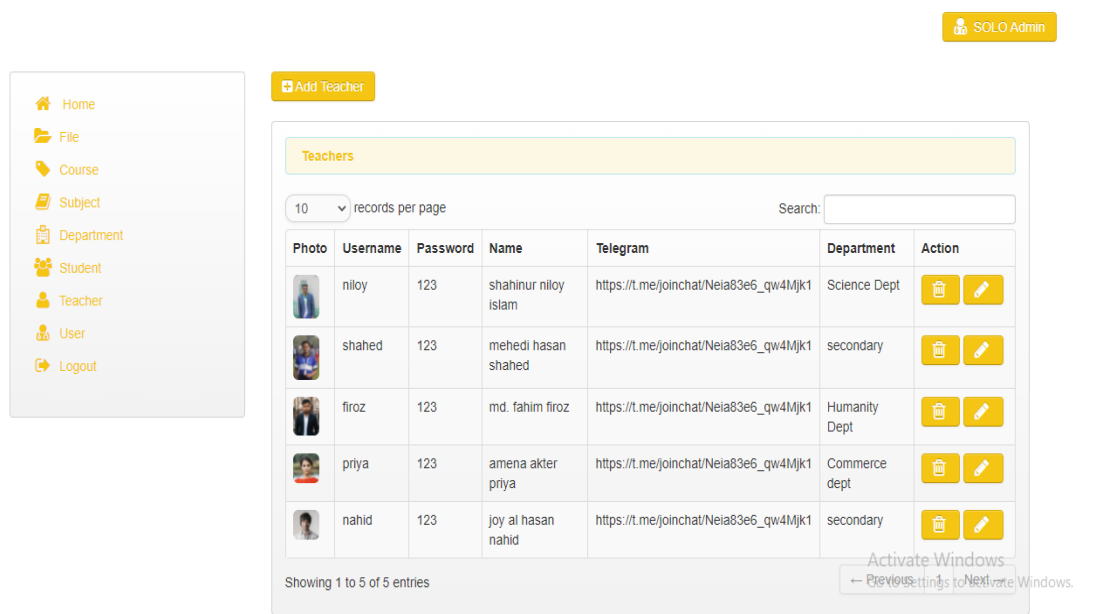


Figure 4.1.13: Teacher Section of Admin

If any new teacher is to be added then the admin will add him through a form. There is an option in the Teacher section of admin named “Add Teacher”. Clicking that button will get the teacher registration form. Figure 4.1.14 represents that below.

Figure 4.1.14: Teacher registration

If one log in to the system as a teacher he will get three feature to use. Figure 4.1.15 below shows the home page as a teacher.



Figure 4.1.15: Teacher Home Page

The teacher's class contains the courses he is taking. Clicking in the class option he can add materials to his courses. He also can add new courses in this portion. The option for adding a particular student to a specific course is also done here. The page looks like the following figure 4.1.16.

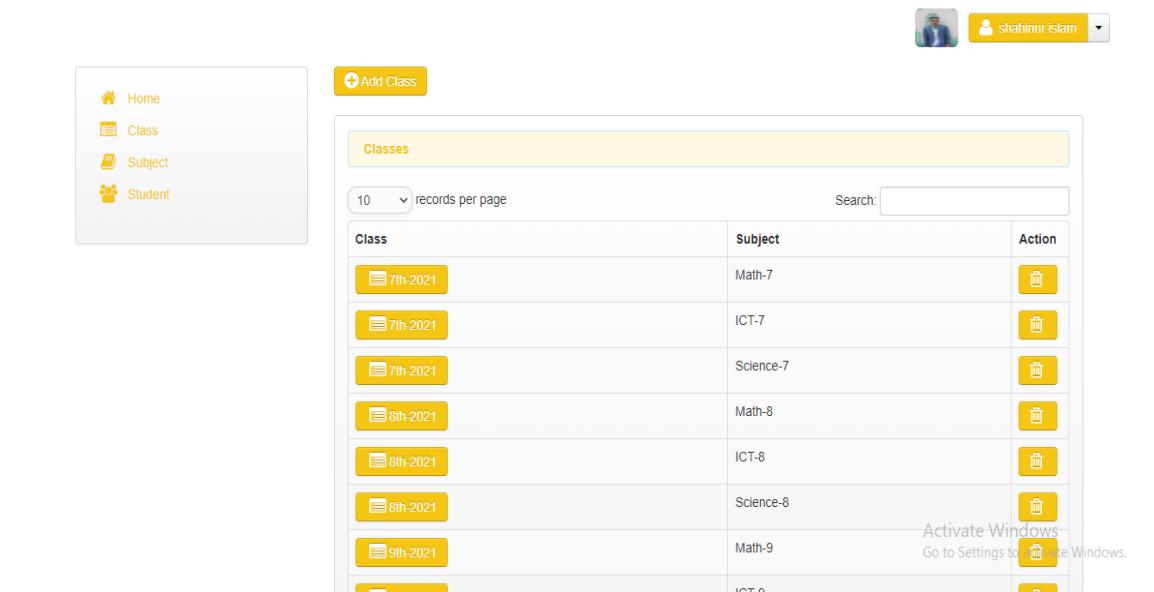


Figure 4.1.16: Teacher's class

The next feature of the teacher is the subject. A teacher can view which subjects he is taking. He can add a new subject in this portion and if wants, can delete an existing one. Figure 4.1.17 shows that section of the teacher.

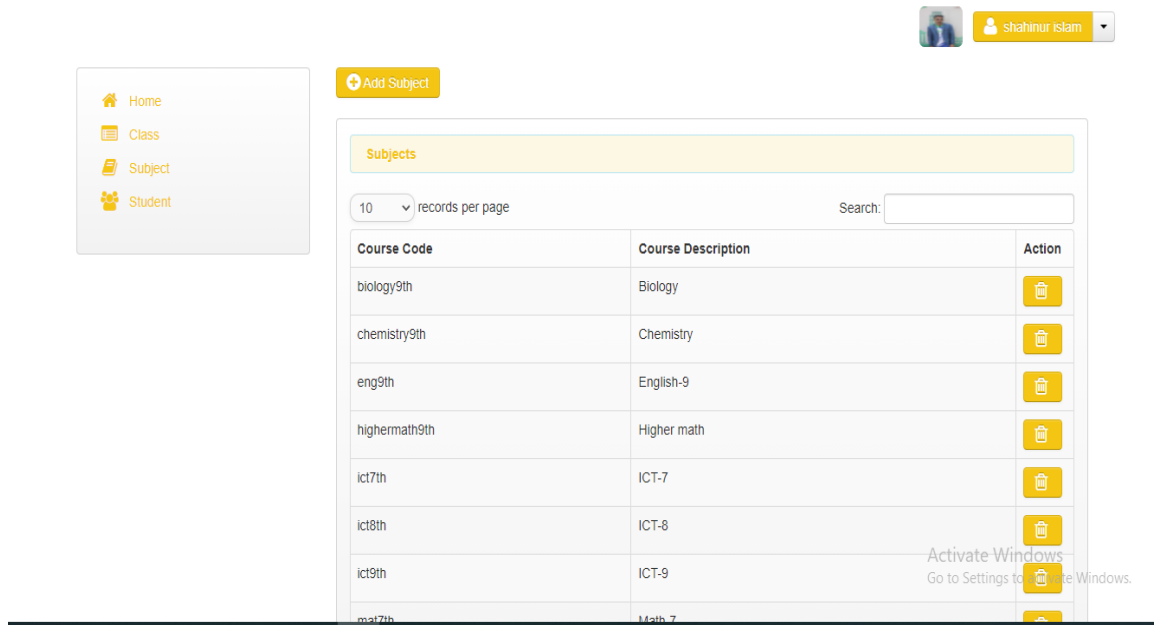


Figure 4.1.17: Teacher's Subject section

Another feature of the teacher is the student. A teacher can maintain his students in this portion. Figure 4.1.18 presents the teacher's Student segment.

- Home
- Class
- Subject
- Student

Add Student











Students		
10 records per page	Search:	
Name	Photo	Action
fajle rabbi ronobi	<input type="checkbox"/>	 
mala afroz tanisha	<input type="checkbox"/>	 
md. hossain zamir	<input type="checkbox"/>	 
md. hossain zayed	<input type="checkbox"/>	 
nisha jahan mim	<input type="checkbox"/>	 

Figure 4.1.18: Teacher's Student section

The third use of the system is the student. A student can log in from the Home page. Figure 4.1.19 shows the page after a student logging in. Here a student can view his/her courses along with that course's telegram link and online class link.

The screenshot shows a user interface for a student's classes. At the top right, there is a user profile icon and a dropdown menu labeled 'fajle ronobi'. Below this is a yellow header bar with the text 'My Classes'. Underneath the header, there is a search bar and a dropdown menu set to '10 records per page'. The main content is a table with the following data:

Class	Subject	Teacher	Telegram Group	Class Link
9th-2021	Science-9	md. firoz	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Math-9	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	ICT-9	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Biology	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Chemistry	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Physics	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Higher math	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>
9th-2021	Science-9	shahinur islam	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/ert-twzr-cvd">https://meet.google.com/ert-twzr-cvd</a>

At the bottom of the table, it says 'Showing 1 to 8 of 8 entries'. There are also navigation buttons: '← Previous', '1', and 'Next →'. A watermark 'Activate Windows Go to Settings to activate Windows.' is visible in the bottom right corner of the screenshot.

Figure 4.1.19: Student's classes

If the student selects any subject that will take him/her to that subject's page where he/she can view all details of that subject such as what contents his/her teacher uploaded and also there will be a quiz option available there for every lesson. Though it will be optional. Figure 4.1.20 below represents the page.

Lesson Name	Learning Topics	Quiz/Lesson Test	Date	Action
HIGHER MATH	NCTB	Check for Test	2021-10-17 05:00:11	
Lecture-01.1	number & intiger	Check for Test	2021-10-17 15:13:48	
Lecture-01.2	Real number	Check for Test	2021-10-17 17:55:36	
Lecture-01.3	Decimal Number	Check for Test	2021-10-17 17:57:32	
Lecture-01.4	determining rational	Check for Test	2021-10-17 23:19:56	
Lecture-01.5	Irrational Number	Check for Test	2021-10-17 23:22:12	
Lecture-02.1	Set	Check for Test	2021-10-17 23:22:58	
Lecture-02.2	Function	Check for Test	2021-10-17 23:24:33	

Figure 4.1.20: Student's class content

If any student requires to register to the system, he/she have to use the below-mentioned function. Here, he/she have to give his/her information to be registered as a student. There is an option for him/her in the last section of the homepage. Figure 4.1.21 shows that portion.

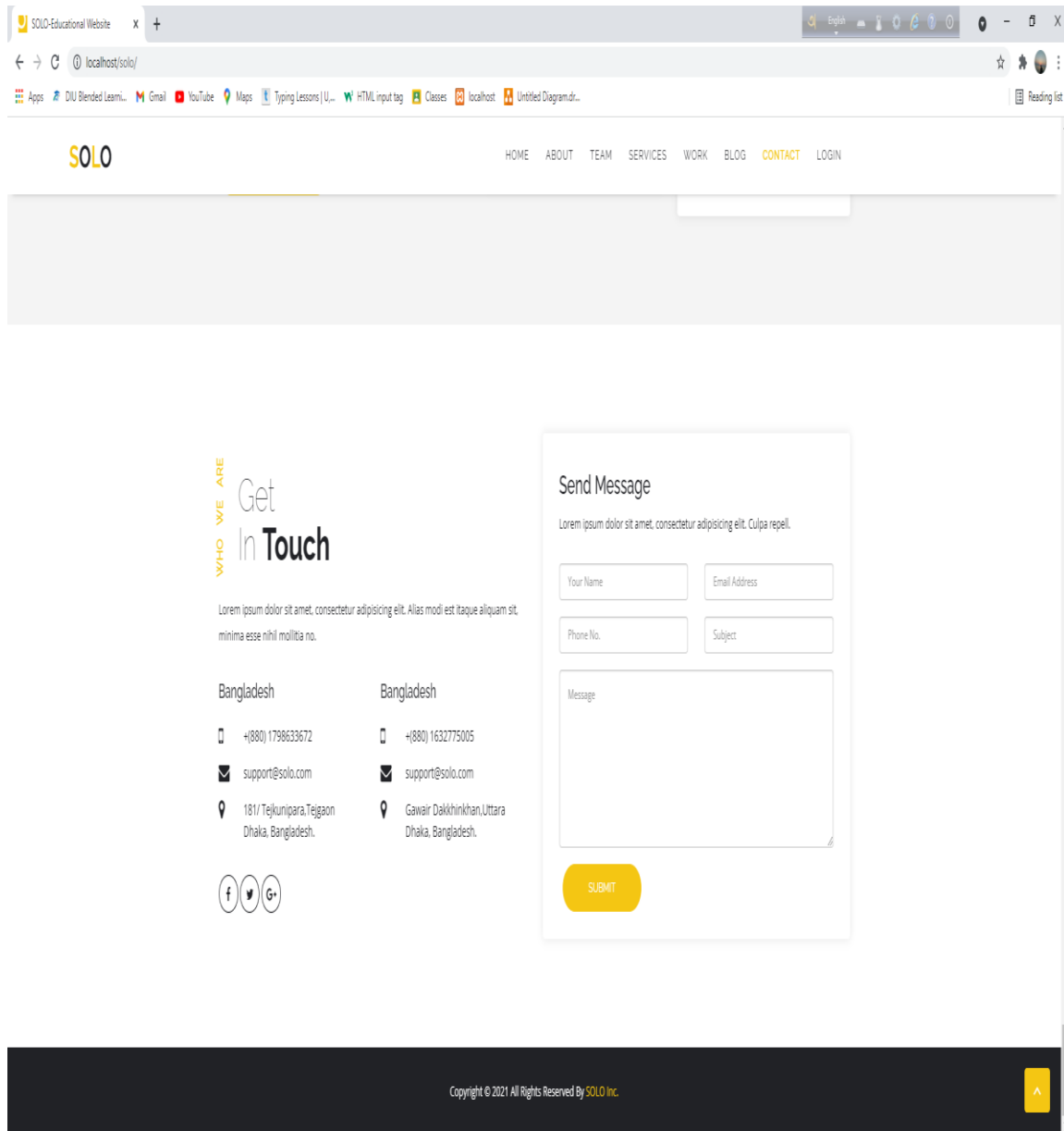


Figure 4.1.21: Contact option

## 4.2 Back-End Design

An application's main functionality works depending on the back-end. In this section, we will see our back-end database and some of the important codes. Firstly, we will show the database and later the codes.



## 4.2.1 Back-End for Admin

This design is the heart of the web design system. It is based on the server site & to store user data. Process and retrieve data when needed. Figure 4.2.1 shows that.

Table	Action	Rows	Type	Collation	Size	Overhead
class	★ Browse Structure Search Insert Empty Drop	26	MyISAM	latin1_swedish_ci	2.8 KiB	-
course	★ Browse Structure Search Insert Empty Drop	6	MyISAM	latin1_swedish_ci	2.3 KiB	-
department	★ Browse Structure Search Insert Empty Drop	5	MyISAM	latin1_swedish_ci	2.2 KiB	-
files	★ Browse Structure Search Insert Empty Drop	32	MyISAM	latin1_swedish_ci	6.5 KiB	-
student	★ Browse Structure Search Insert Empty Drop	10	MyISAM	latin1_swedish_ci	2.6 KiB	-
subject	★ Browse Structure Search Insert Empty Drop	25	MyISAM	latin1_swedish_ci	2.9 KiB	-
sws	★ Browse Structure Search Insert Empty Drop	53	MyISAM	latin1_swedish_ci	4.1 KiB	-
teacher	★ Browse Structure Search Insert Empty Drop	6	MyISAM	latin1_swedish_ci	3.3 KiB	328 B
teacher_student	★ Browse Structure Search Insert Empty Drop	28	MyISAM	latin1_swedish_ci	2.4 KiB	-
teacher_subject	★ Browse Structure Search Insert Empty Drop	35	MyISAM	latin1_swedish_ci	2.4 KiB	-
user	★ Browse Structure Search Insert Empty Drop	1	MyISAM	latin1_swedish_ci	2.0 KiB	-
11 tables	Sum	227	InnoDB	utf8mb4_general_ci	33.5 KiB	328 B

Figure 4.2.1 Back-end Admin

## 4.2.2 Back-End for Course

Figure 4.2.2 is the back-end of the course database of “SOLO”.

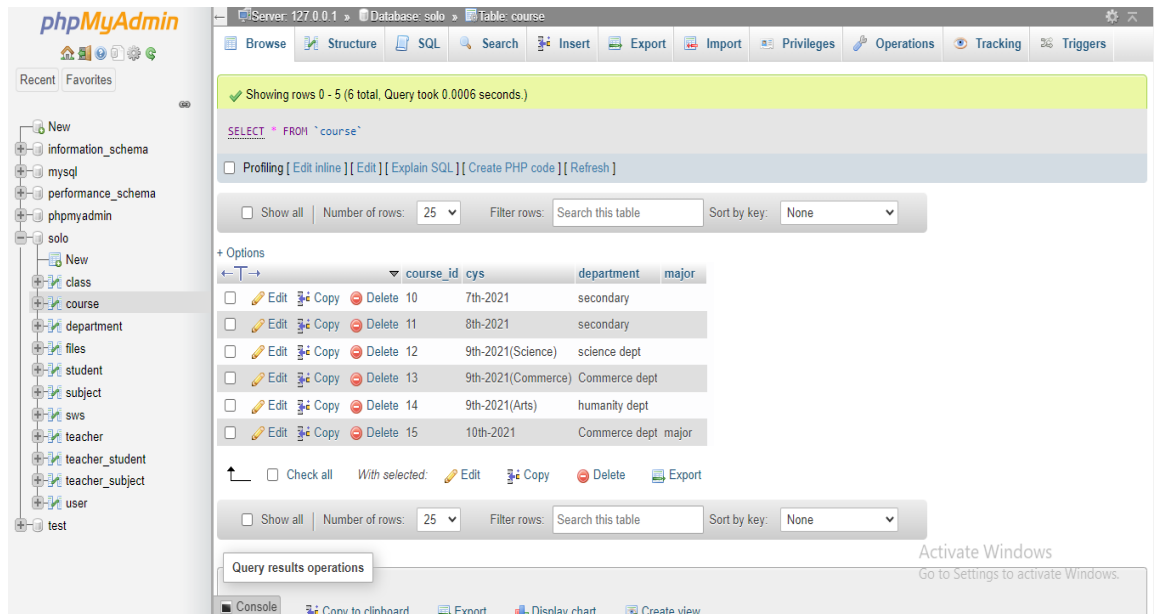


Figure 4.2.2 Back-End Course

### 4.2.3 Back-End for Department

Figure 4.2.3 is the back-end of the department database of our project.

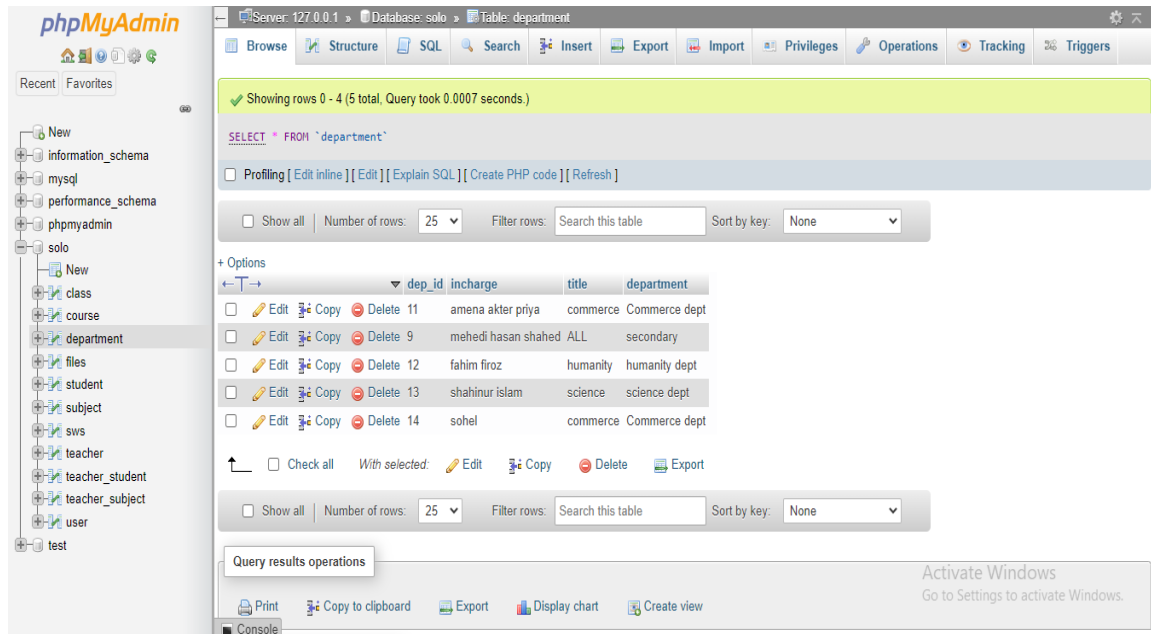


Figure 4.2.3 Back-End Department

## 4.2.4 Back-End for File upload

Figure 4.2.4 presents the database of the file upload of the project.

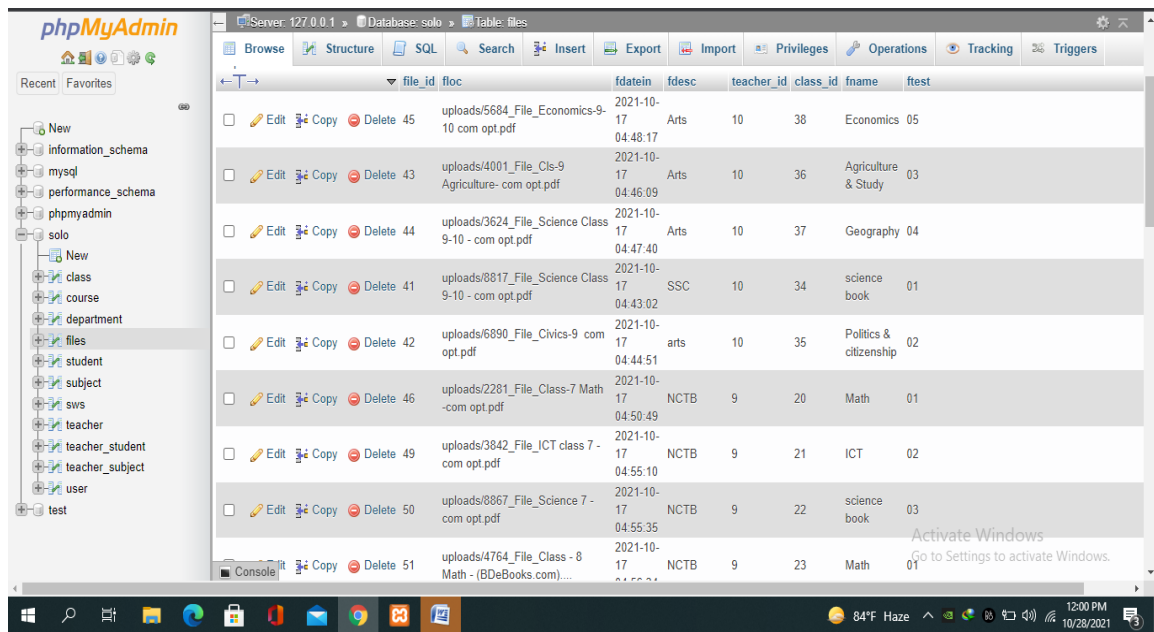


Figure 4.2.4 Back-End File upload

## 4.2.5 Back End for Student

Figure 4.2.5 show the database where the student will be added by system administration & student who have a unique username.

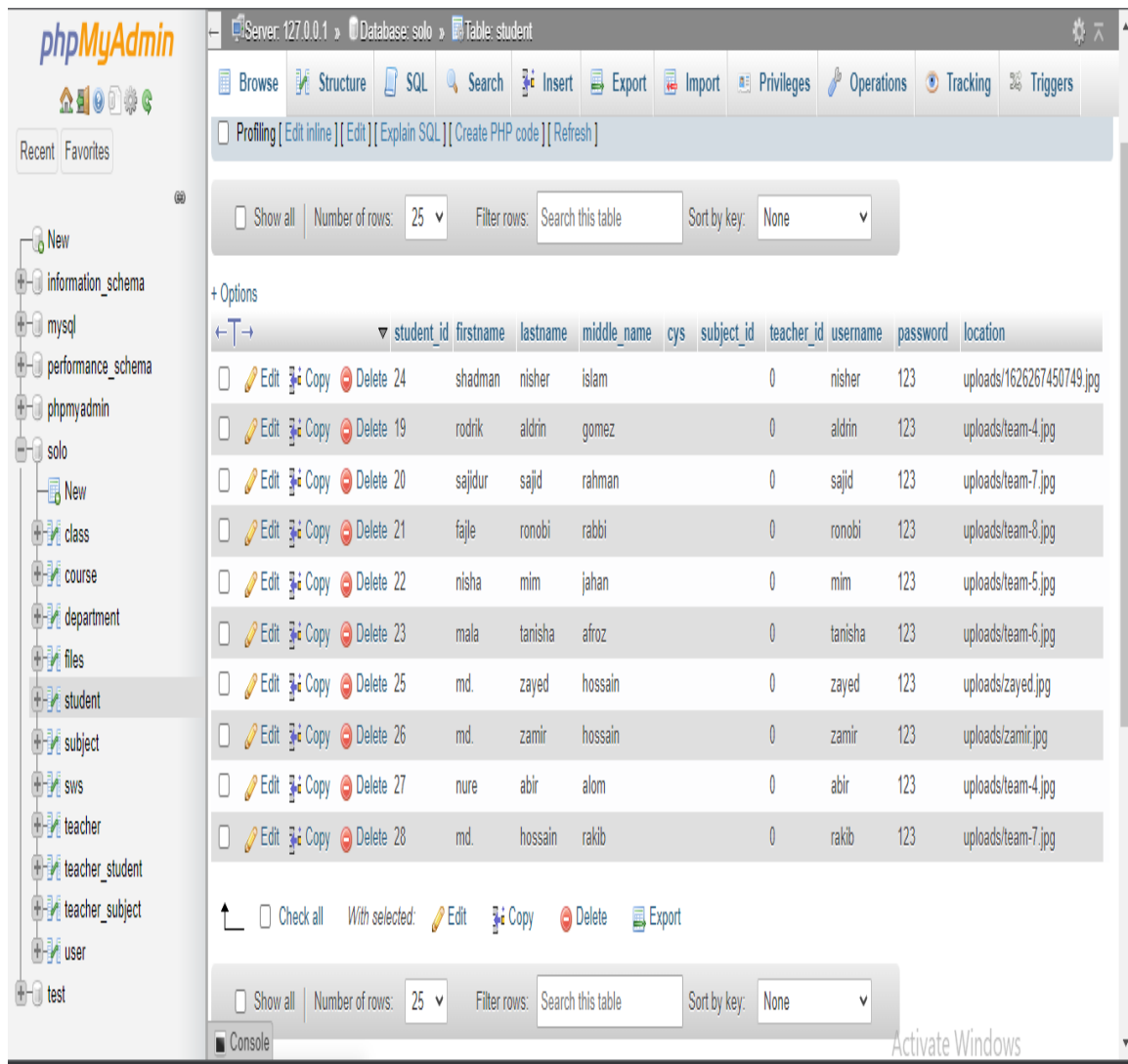


Figure 4.2.5 Student Database

## 4.2.6 Back-End for Subject

Figure 4.2.6 is the back –end for the subject database of our project.

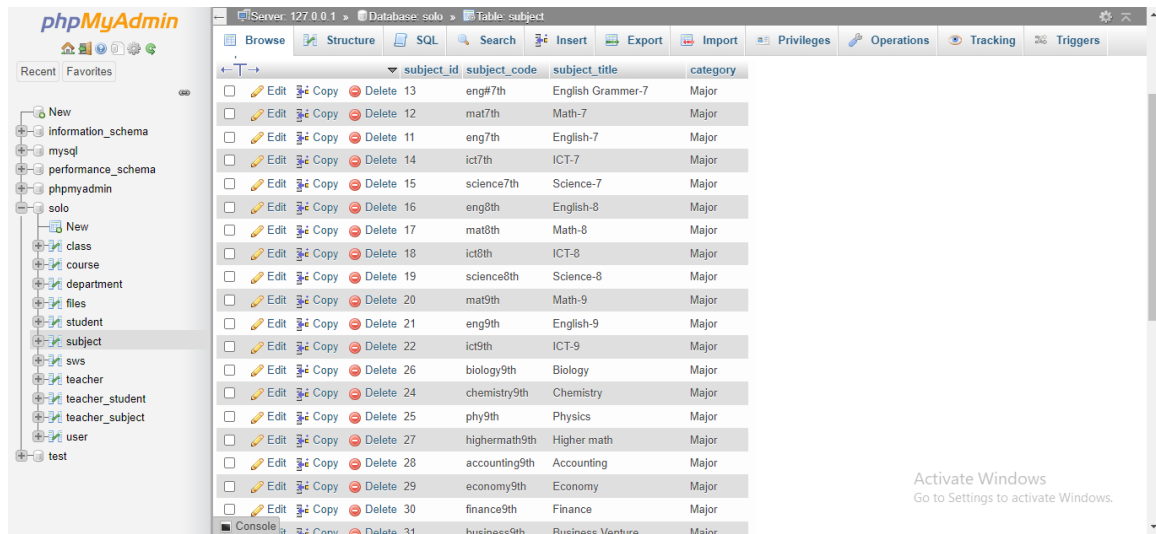


Figure 4.2.6 Back-End Subject

## 4.2.7 Back-End for Subject Wise Student

Figure 4.2.7 shows the back –end of the subject wise student database of the project.

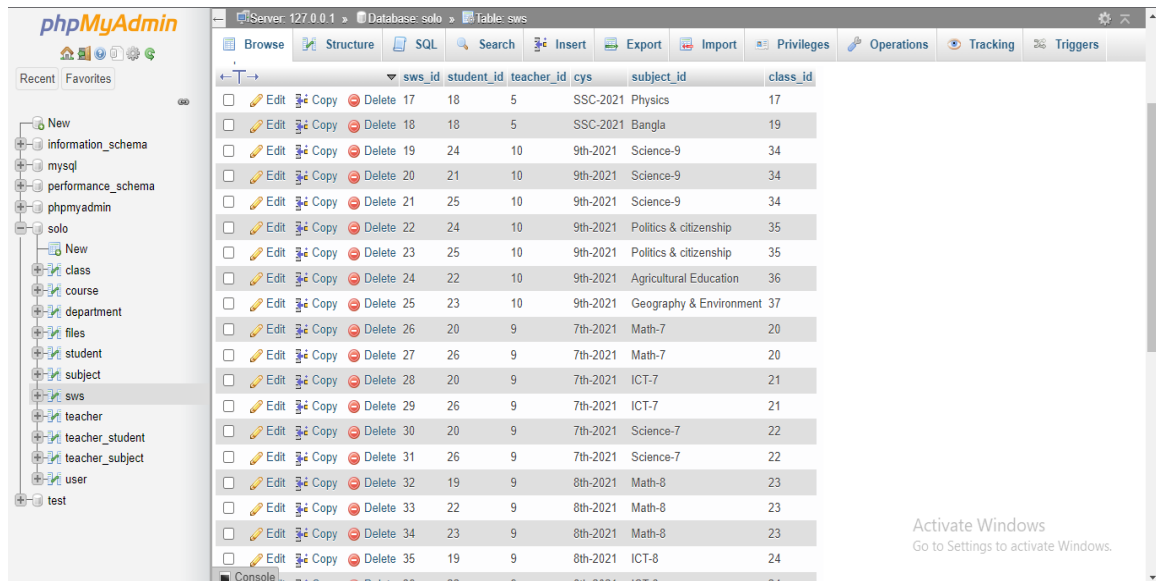
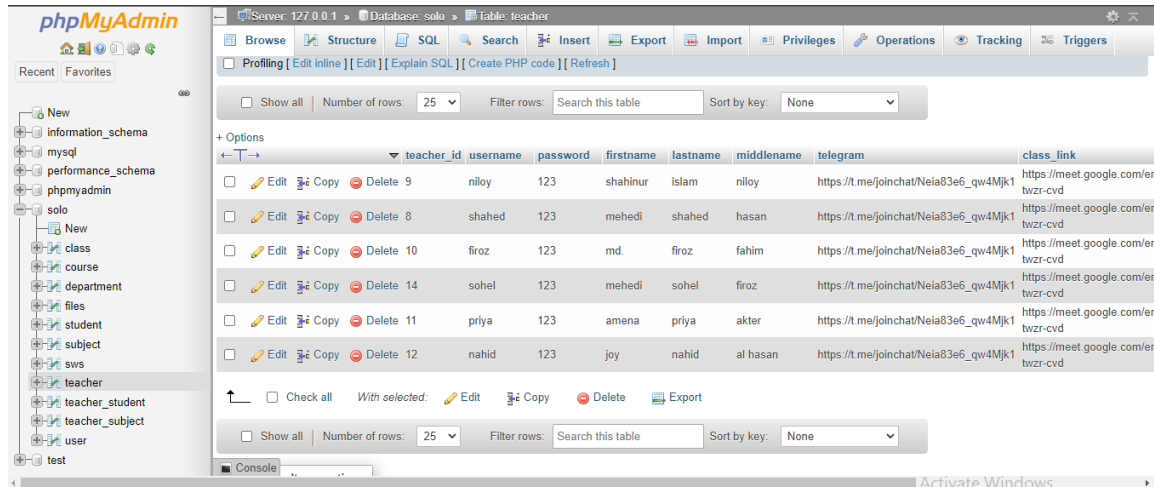


Figure 4.2.7 Back-End SWS

## 4.2.8 Back End for Teacher

Figure 4.2.8 shows that it is a database of teacher adding where every teacher has a unique id.



	teacher_id	username	password	firstname	lastname	middlename	telegram	class_link
<input type="checkbox"/>	9	niloy	123	shahinur	islam	niloy	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>
<input type="checkbox"/>	8	shahed	123	mehedi	shahed	hasan	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>
<input type="checkbox"/>	10	firoz	123	md.	firoz	fahim	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>
<input type="checkbox"/>	14	sohel	123	mehedi	sohel	firoz	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>
<input type="checkbox"/>	11	priya	123	amena	priya	akter	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>
<input type="checkbox"/>	12	nahid	123	joy	nahid	al hasan	<a href="https://t.me/joinchat/Neia83e6_qw4Mjk1">https://t.me/joinchat/Neia83e6_qw4Mjk1</a>	<a href="https://meet.google.com/er-twzr-cvd">https://meet.google.com/er-twzr-cvd</a>

Figure 4.2.7 Teacher Database

## 4.2.9 Back-End for Teacher wise Student

Figure 4.2.8 shows that it is a database of teacher wise student.

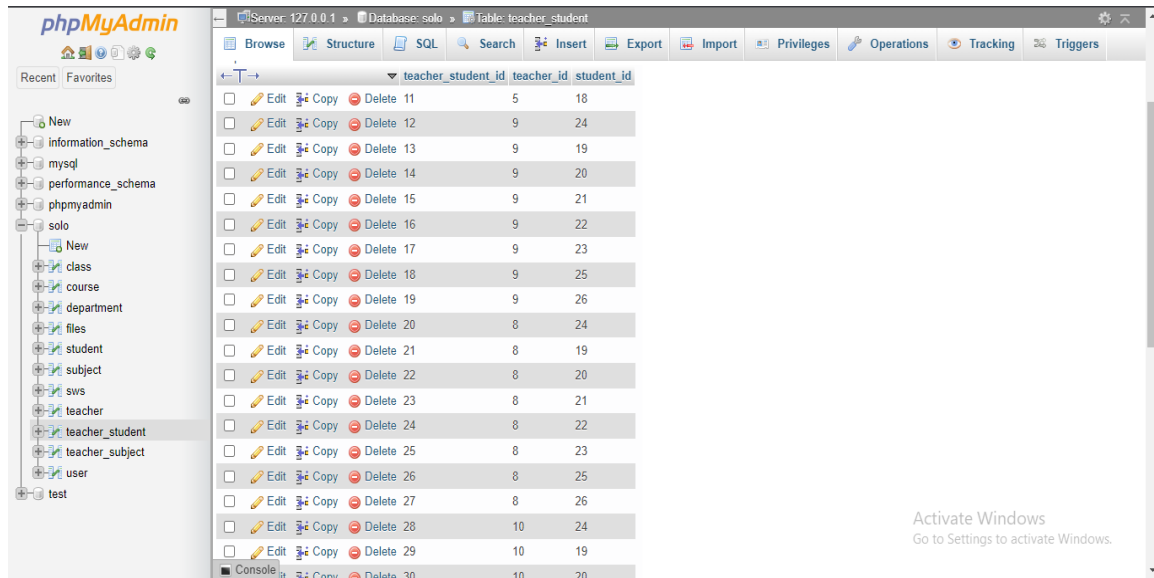


Figure 4.2.9 Teacher Wise Student Database

## 4.2.10 Back-End for Teacher Subject

Figure 4.2.10 is the back-end of a database where a teacher can add a subject for the student to the chosen class & department of the project.



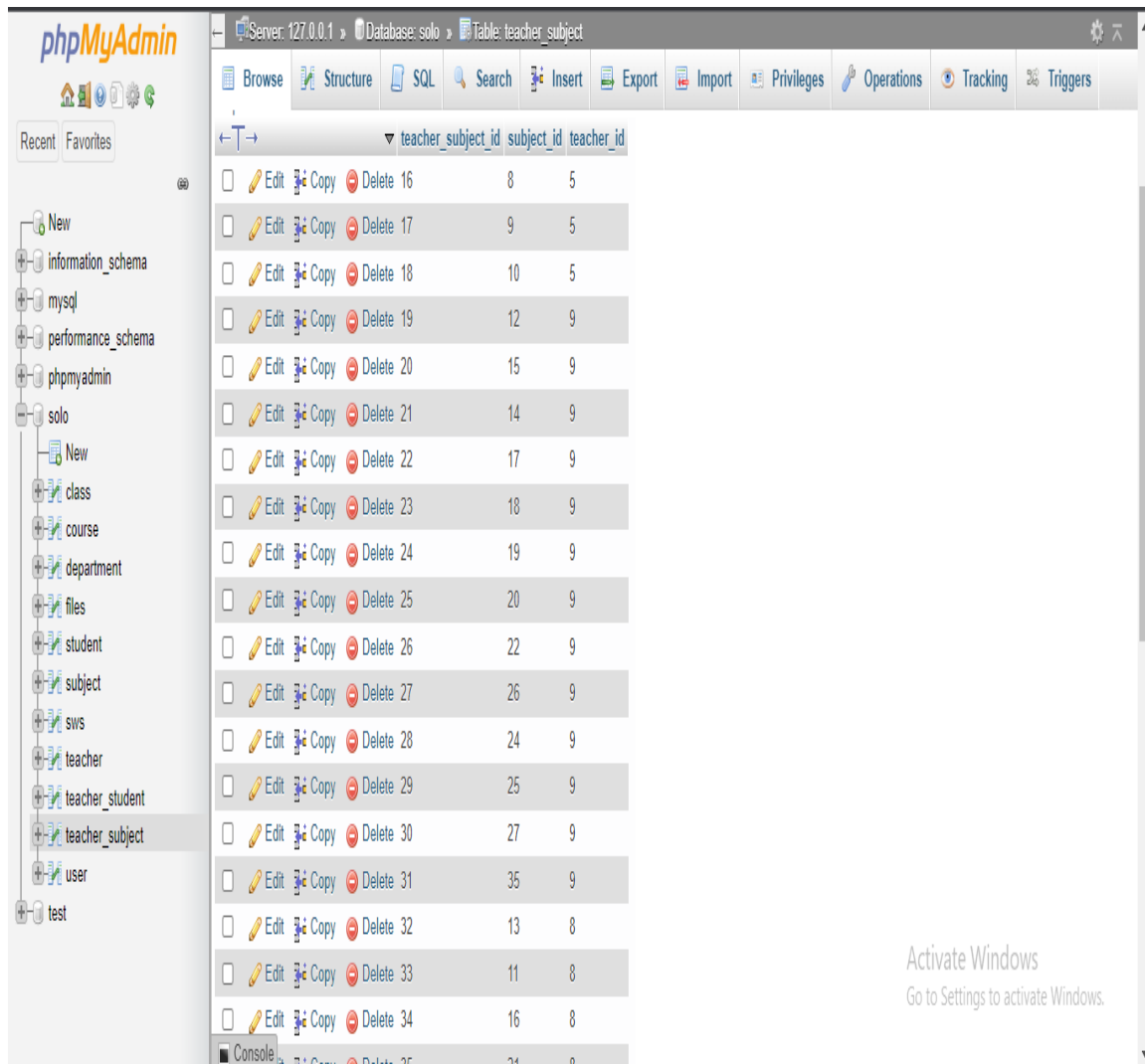


Figure 4.2.10 Back-End Teacher Subject

## 4.2.11 Back-End for Class

Figure 4.2.11 is the back-end of the class database of our project.

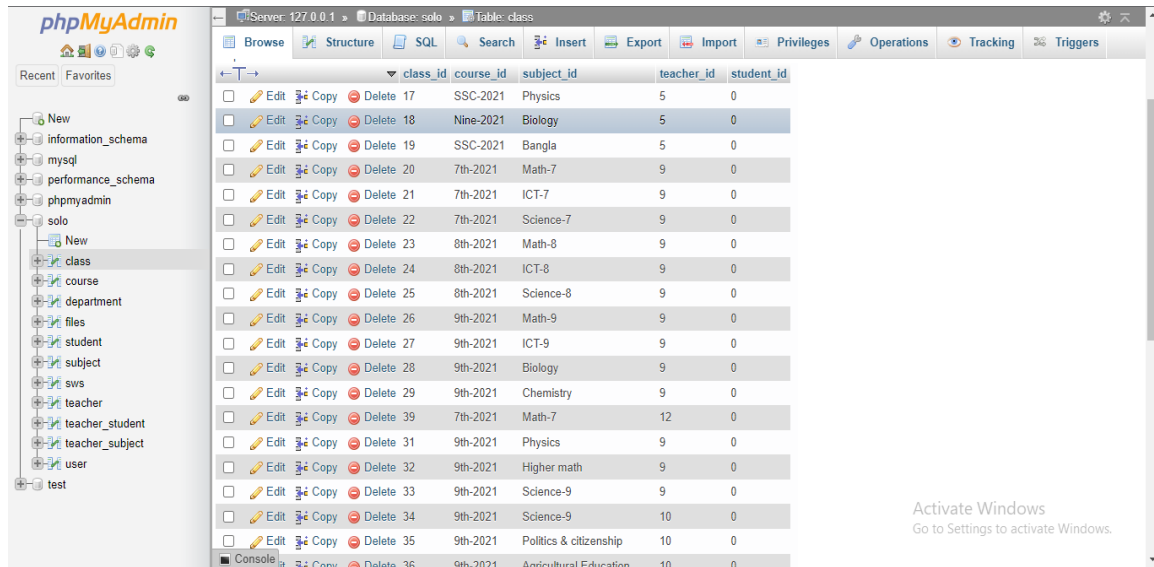


Figure 4.2.11 Back-End Class

### 4.3 Code Segment

Coding is the main logical back-end of any application. This portion of any system makes the software interactive and responsive. If we want to show all of the codes that won't be suitable for this report and will be very lengthy. Also, we think, it is not necessary at all. So, we add some of the important code segments of the project.

#### 4.3.1 Class Code

In figure 4.3.1 we have added one of the code segments of the class portion. This portion works with the student, teacher, and course.

```

1 <?php
2 $get_id = $_GET['id'];
3 include('header2.php');
4 include ('session.php');
5 $user_query = mysqli_query($conn,"select * from teacher where teacher_id='$
   session_id'" or die(mysqli_error());
6 $user_row = mysqli_fetch_array($user_query);
7
8 $course_query = mysqli_query($conn,"select * from class where teacher_id='$
   session_id' and class_id='$get_id'" or die(mysqli_error());
9 $course_row = mysqli_fetch_array($course_query);
10 $course_id = $course_row['course_id'];
11 ?>
12 <?php
13 $query_class = mysqli_query($conn,"select * from class where teacher_id='$session_id
   ' and class_id='$get_id'" or die(mysqli_error());
14 $row_class = mysqli_fetch_array($query_class);
15 $id_class = $row_class['class_id'];
16 ?>
17 <body>
18
19 <?php include('navhead_user.php'); ?>

```

Figure 4.3.1 Student Class Code

### 4.3.2 Class of Student Code

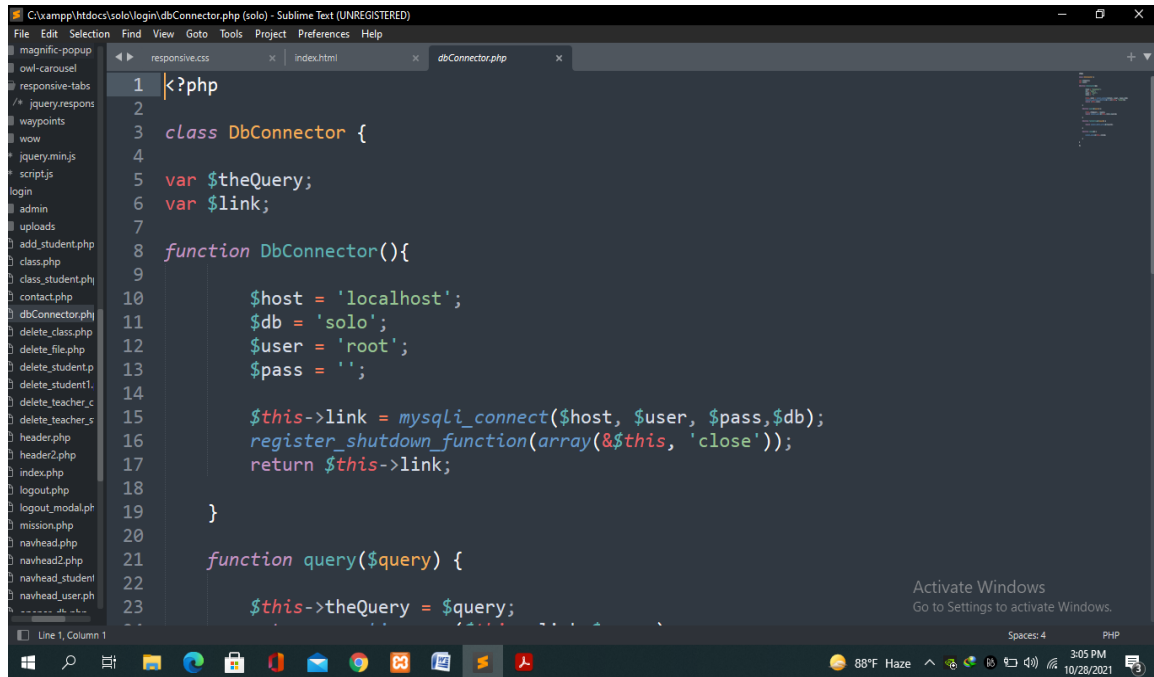
```

1 <?php
2 $get_id = $_GET['id'];
3
4 include('header2.php');
5 include ('session.php');
6 $user_query = mysqli_query($conn,"select * from student where student_id='$session_id
   '" or die(mysqli_error());
7 $user_row = mysqli_fetch_array($user_query);
8 ?>
9
10 <?php
11 $query_class = mysqli_query($conn,"select * from class where class_id='$get_id'" or
   die(mysqli_error());
12 $row_class = mysqli_fetch_array($query_class);
13 $teacher_id = $row_class['teacher_id'];
14
15 $teacher_query=mysqli_query($conn,"select *from teacher where teacher_id='$teacher_id
   '" or die(mysqli_error());
16 $teacher_row= mysqli_fetch_array($teacher_query);
17 ?>
18 <body>
19
20 <?php include('navhead_student.php'); ?>

```

Figure 4.3.2 Student php Code

### 4.3.3 Data base Connect with Php



```
1 <?php
2
3 class DbConnector {
4
5     var $theQuery;
6     var $link;
7
8     function DbConnector(){
9
10         $host = 'localhost';
11         $db = 'solo';
12         $user = 'root';
13         $pass = '';
14
15         $this->link = mysqli_connect($host, $user, $pass,$db);
16         register_shutdown_function(array(&$this, 'close'));
17         return $this->link;
18     }
19
20
21     function query($query) {
22
23         $this->theQuery = $query;
```

Figure 4.3.3 Database connect with php Code

### 4.3.4 Delete class

```
1 <?php
2 include('admin/connect.php');
3 $get_id=$_GET['id'];
4
5 mysqli_query($conn,"delete from class where class_id='$get_id'")or die(mysqli_error()
6 );
7 header('location:teacher_class.php');
8 ?>
```

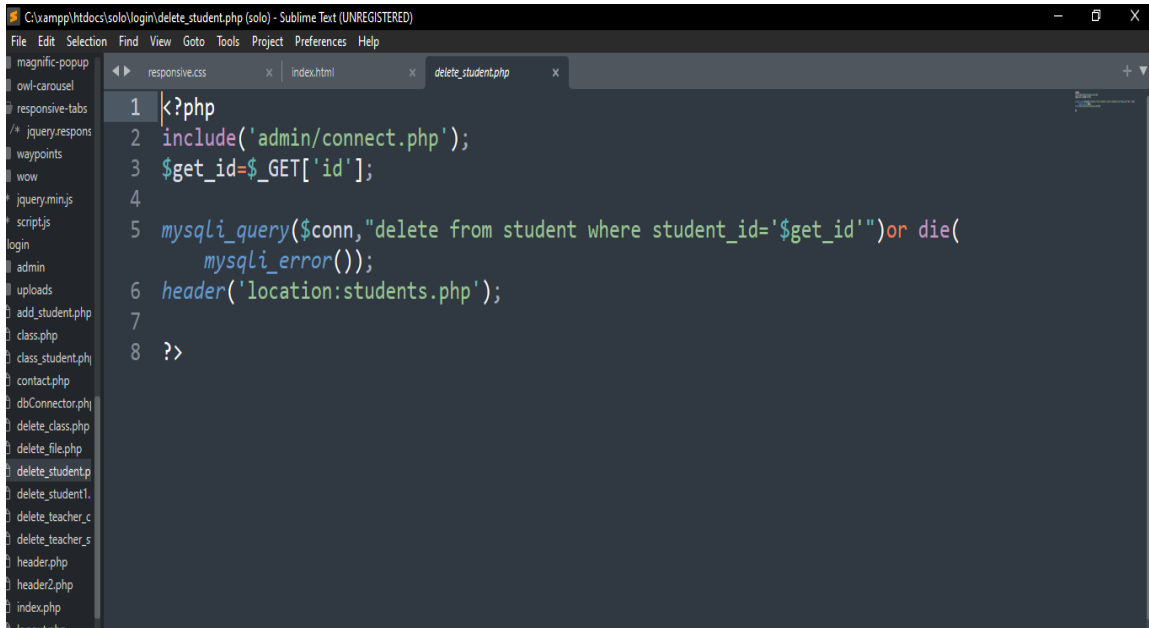
Figure 4.3.4 Delete Code

### 4.3.5 Delete File code

```
1 <?php
2 include('admin/connect.php');
3 $get_id = $_POST['file_id'];
4 $class_id=$_POST['class_id'];
5
6 mysqli_query($conn,"delete from files where file_id='$get_id'") or die(mysqli_error()
7 );
8 ?>
9
10 <script type="text/javascript">
11 window.location="class.php?php echo '?id=' . $class_id; ?>";
12 </script>
13
14
15
```

Figure 4.3.5 Delete file Code

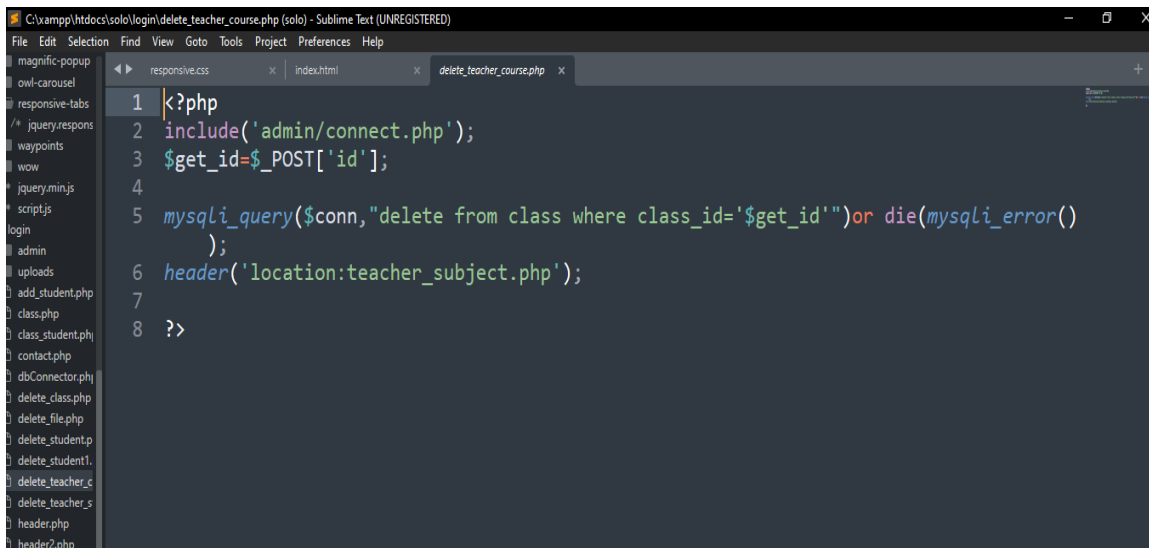
### 4.3.6 Delete Student



```
1 <?php
2 include('admin/connect.php');
3 $get_id=$_GET['id'];
4
5 mysqli_query($conn,"delete from student where student_id='$get_id'")or die(
6     mysqli_error());
7 header('location:students.php');
8 ?>
```

Figure 4.3.6 Delete student Code

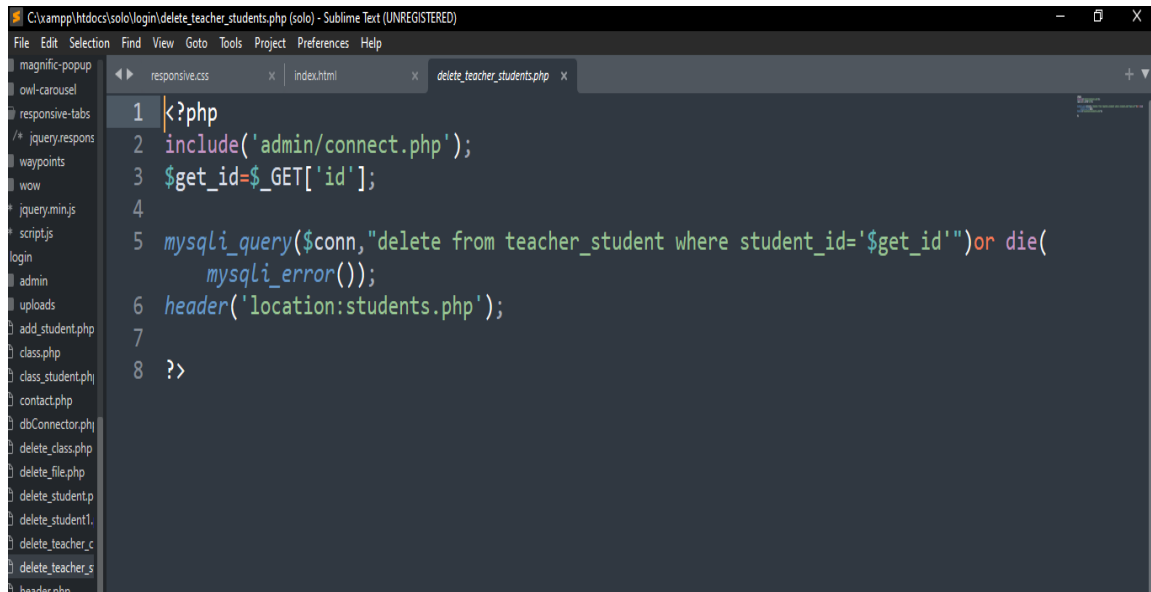
### 4.3.7 Delete Teacher



```
1 <?php
2 include('admin/connect.php');
3 $get_id=$_POST['id'];
4
5 mysqli_query($conn,"delete from class where class_id='$get_id'")or die(mysqli_error()
6     );
7 header('location:teacher_subject.php');
8 ?>
```

Figure 4.3.7 Delete teacher Code

### 4.3.8 Delete by Teacher a Student



```
C:\xampp\htdocs\solo\login\delete_teacher_students.php (solo) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
responsive.css x index.html x delete_teacher_students.php x
1 <?php
2 include('admin/connect.php');
3 $get_id=$_GET['id'];
4
5 mysqli_query($conn,"delete from teacher_student where student_id='$get_id'")or die(
6     mysqli_error());
7 header('location:students.php');
8 ?>
```

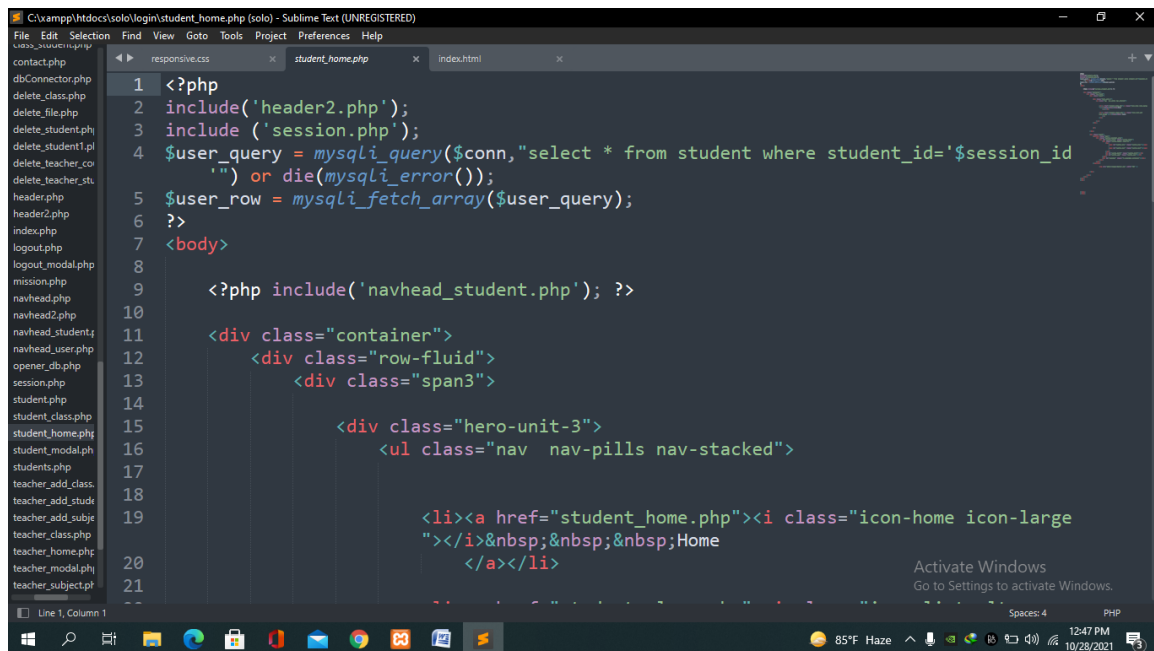
Figure 4.3.8: Delete teacher \_student Code

### 4.3.9 Student Home code





### 4.3.11 Student. Php code



```
1 <?php
2 include('header2.php');
3 include ('session.php');
4 $user_query = mysqli_query($conn,"select * from student where student_id='$session_id
   ''") or die(mysqli_error());
5 $user_row = mysqli_fetch_array($user_query);
6 ?>
7 <body>
8
9 <?php include('navhead_student.php'); ?>
10
11 <div class="container">
12     <div class="row-fluid">
13         <div class="span3">
14
15             <div class="hero-unit-3">
16                 <ul class="nav nav-pills nav-stacked">
17
18                     <li><a href="student_home.php"><i class="icon-home icon-large
19                         "></i>&nbsp;&nbsp;&nbsp;&nbsp;Home
20                     </a></li>
21
```

Figure 4.3.3: Student php Code

### 4.2.12 Student portal Code

```
19 <section class="main">
20 <div class="custom-calendar-wrap">
21 <div id="custom-inner" class="custom-inner">
22 <div class="custom-header clearfix">
23 <nav>
24 <span id="custom-prev" class="custom-prev"></span>
25 <span id="custom-next" class="custom-next"></span>
26 </nav>
27 <h2 id="custom-month" class="custom-month"></h2>
28 <h3 id="custom-year" class="custom-year"></h3>
29 </div>
30 <div id="calendar" class="fc-calendar-container"></div>
31 </div>
32 </div>
33 </section>
34
35 
36
37
38 </div>
39
```

Figure 4.3.12: Student portal Code

### 4.3.13 Student Modal

Figure 4.3.13 where student we create student login form.

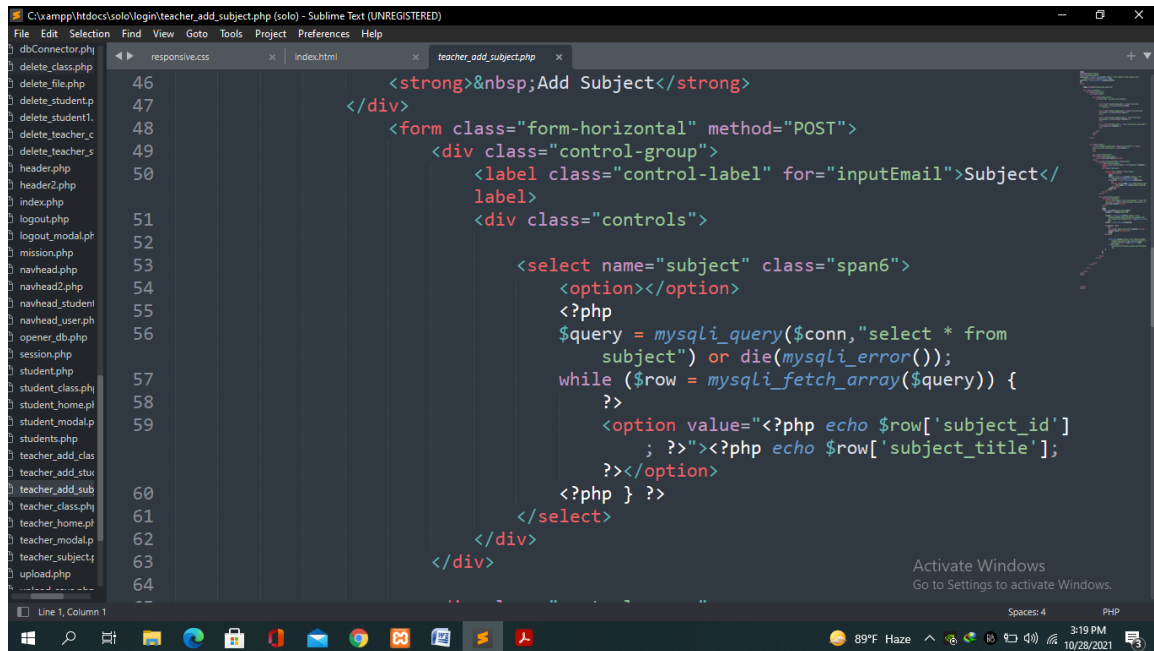
```
C:\xampp\htdocs\solo\login\student_modal.php (solo) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
responsive.css x student_modal.php x index.html x
delete_teacher_co
delete_teacher_stu
header.php
header2.php
index.php
logout.php
logout_modal.php
mission.php
navhead.php
navhead2.php
navhead_student_t
navhead_user.php
opener_db.php
session.php
student.php
student_class.php
student_home.php
student_modal.ph
students.php
teacher_add_class
teacher_add_stude
teacher_add_subje
teacher_class.php
teacher_home.php
teacher_modal.ph
teacher_subject.ph
upload.php
upload_save.php
vision.php
ideo
debug.log
index.html
12 <div class="control-group">
13 <label class="control-label" for="inputEmail">Username</label>
14 <div class="controls">
15 <input type="text" name="username" id="inputEmail" placeholder="
Username">
16 </div>
17 </div>
18 <div class="control-group">
19 <label class="control-label" for="inputPassword">Password</label>
20 <div class="controls">
21 <input type="password" name="password" id="inputPassword"
placeholder="Password">
22 </div>
23 </div>
24
25
26 <div class="control-group">
27 <div class="controls">
28 <button type="submit" name="login" class="btn btn-info"><i class=
"icon-signin icon-large"></i>&nbsp;Sign in</button>
29 </div>
30
31
```

Figure 4.3.13: Student Modal Code

### 4.3.14 Teacher Add Class Code



### 4.3.16 Teacher Add Subject code



```
46 <strong>&nbsp;Add Subject</strong>
47 </div>
48 <form class="form-horizontal" method="POST">
49 <div class="control-group">
50 <label class="control-label" for="inputEmail">Subject</label>
51 <div class="controls">
52 <select name="subject" class="span6">
53 <option></option>
54 <?php
55 $query = mysqli_query($conn,"select * from
56 subject") or die(mysqli_error());
57 while ($row = mysqli_fetch_array($query)) {
58 >
59 <option value="<?php echo $row['subject_id']
60 ; ?"><?php echo $row['subject_title'];
61 ?></option>
62 <?php } ?>
63 </select>
64 </div>
</div>
```

Figure 4.3.16: Teacher Add Subject Code

### 4.3.17 Teacher Home code





```

40 session_write_close();
41 echo("</script>location.href = 'teacher_class.php'</script>");
42 exit();
43 }
44
45 $rd2 = mt_rand(1000, 9999) . "_File";
46
47
48 if ((!empty($_FILES["uploaded_file"])) && ($_FILES["uploaded_file"]["error"] == 0))
49 {
50     $filename = basename($_FILES["uploaded_file"]["name"]);
51
52     $ext = substr($filename, strrpos($filename, '.') + 1);
53
54     if (($ext != "exe") && ($_FILES["uploaded_file"]["type"] != "application/
55         x-msdownload")) {
56
57         $newname = "uploads/" . $rd2 . "_" . $filename;
58
59         if (!file_exists($newname)) {
60

```

Figure 4.3.20: Upload File save Code

### 4.3.21 Upload File Code

```

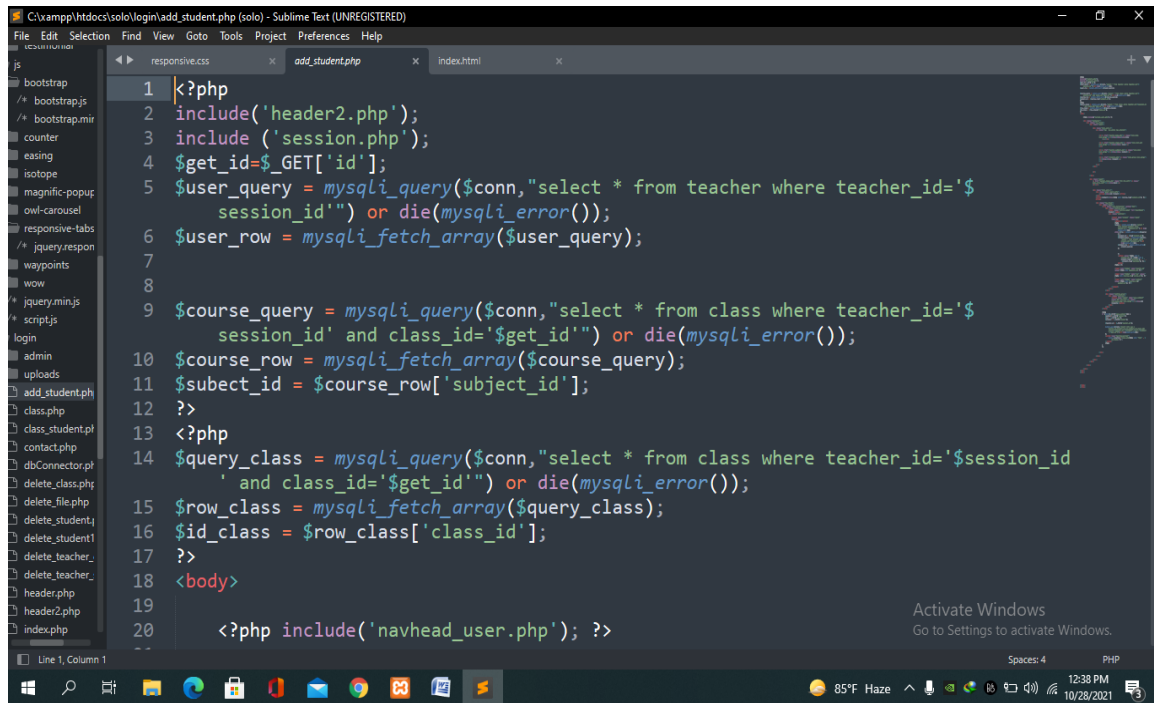
46
47
48 if ((!empty($_FILES["uploaded_file"])) && ($_FILES["uploaded_file"]["error"] == 0))
49 {
50     $filename = basename($_FILES["uploaded_file"]["name"]);
51
52     $ext = substr($filename, strrpos($filename, '.') + 1);
53
54     if (($ext != "exe") && ($_FILES["uploaded_file"]["type"] != "application/
55         x-msdownload")) {
56
57         $newname = "uploads/" . $rd2 . "_" . $filename;
58
59         if (!file_exists($newname)) {
60
61             if ((move_uploaded_file($_FILES["uploaded_file"]["tmp_name"], $newname))
62                 ) {
63
64                 $qry2 = "INSERT INTO files (
65                     fdesc,floc,fdatein,teacher_id,class_id,fname,ftest)VALUES a($

```

Figure 4.3.21: Teacher Upload File Code



### 4.3.22 Admin Add Student



The image shows a screenshot of a Sublime Text editor window titled 'C:\xampp\htdocs\solo\login\add\_student.php (solo) - Sublime Text (UNREGISTERED)'. The editor displays PHP code for an 'add\_student.php' file. The code includes headers, session management, and database queries to retrieve user and class information. The code is as follows:

```
1 <?php
2 include('header2.php');
3 include('session.php');
4 $get_id=$_GET['id'];
5 $user_query = mysqli_query($conn,"select * from teacher where teacher_id='$
  session_id'" or die(mysqli_error());
6 $user_row = mysqli_fetch_array($user_query);
7
8
9 $course_query = mysqli_query($conn,"select * from class where teacher_id='$
  session_id' and class_id='$get_id'" or die(mysqli_error());
10 $course_row = mysqli_fetch_array($course_query);
11 $subject_id = $course_row['subject_id'];
12 ?>
13 <?php
14 $query_class = mysqli_query($conn,"select * from class where teacher_id='$session_id
  ' and class_id='$get_id'" or die(mysqli_error());
15 $row_class = mysqli_fetch_array($query_class);
16 $id_class = $row_class['class_id'];
17 ?>
18 <body>
19
20 <?php include('navhead_user.php'); ?>
```

Figure 4.3.22: Admin Add Student Code

### 4.3.23 Logout Code

```

C:\xampp\htdocs\solo\login\logout_modal.php (solo) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
responsive.css x logout_modal.php x index.html x
1 <div id="logout" class="modal hide fade" tabindex="-1" role="dialog" aria-labelledby=
  "myModalLabel" aria-hidden="true">
2 <div class="modal-header">
3 </div>
4 <div class="modal-body">
5 <div class="alert alert-info">Are you sure want to <strong>Logout</strong>?</div>
6
7 </div>
8 <div class="modal-footer">
9 <a href="logout.php" class="btn btn-info"><i class="icon-check icon-large"></i>&
  nbsp;Yes</a>
10 <button class="btn" data-dismiss="modal" aria-hidden="true"><i class="
  icon-remove-sign icon-large"></i>&nbsp;Close</button>
11
12 </div>
13 </div>

```

Figure 4.3.23: Logout Code

### 4.3.24 Contact Code

```

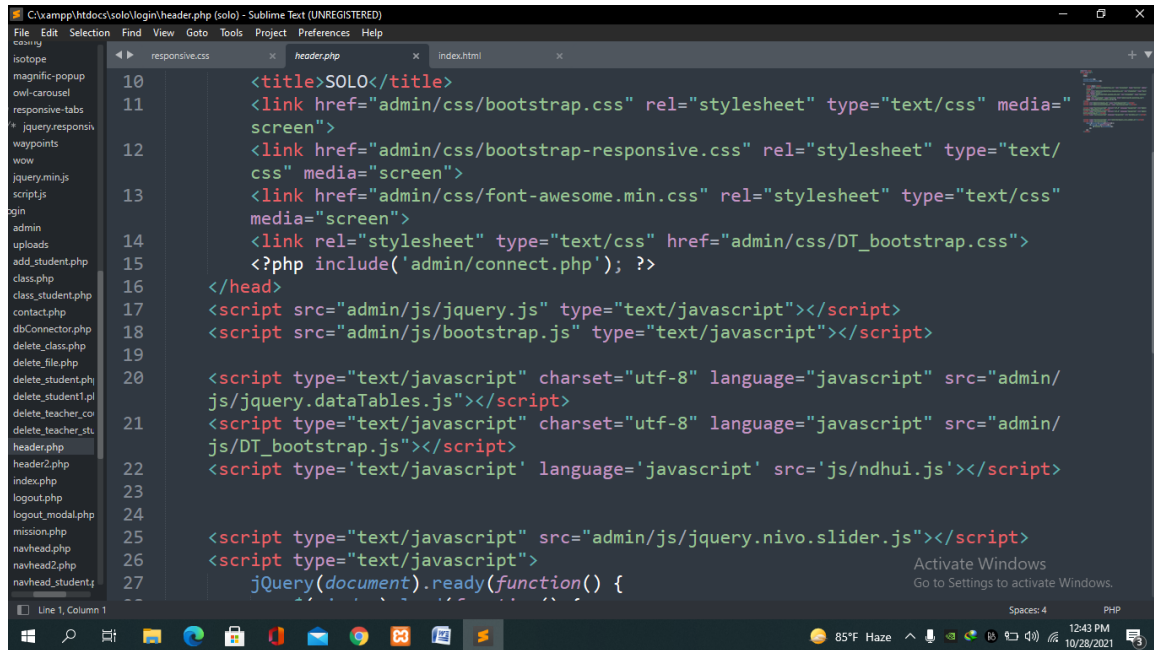
C:\xampp\htdocs\solo\login\contact.php (solo) - Sublime Text (UNREGISTERED)
File Edit Selection Find View Goto Tools Project Preferences Help
responsive.css x contact.php x index.html x
47
48 <div class="row-fluid">
49 <div class="span6">
50
51 <p><i class="icon-circle"></i>&nbsp;Niloy - +880
  1746-630938</p>
52 <p><i class="icon-circle"></i>&nbsp;Admin - +880
  1746-630938</p>
53 <p><i class="icon-circle"></i>&nbsp;Admin - +880
  1746-630938</p>
54 <p><i class="icon-circle"></i>&nbsp;Admin - +880
  1746-630938</p>
55
56 </div>
57 <div class="span6">
58
59 <p><i class="icon-circle"></i>&nbsp;Admin - +880
  1624-505891</p>
60 <p><i class="icon-circle"></i>&nbsp;Manager -
  +880 1624-505891</p>
61 <p><i class="icon-circle"></i>&nbsp;Manager -
  +880 1624-505891</p>
62 <p><i class="icon-circle"></i>&nbsp;Manager
  1624-505891</p>

```

Figure 4.3.24: Contact Code

Figure 4.3.24: Contact Code

### 4.3.25 Header Code

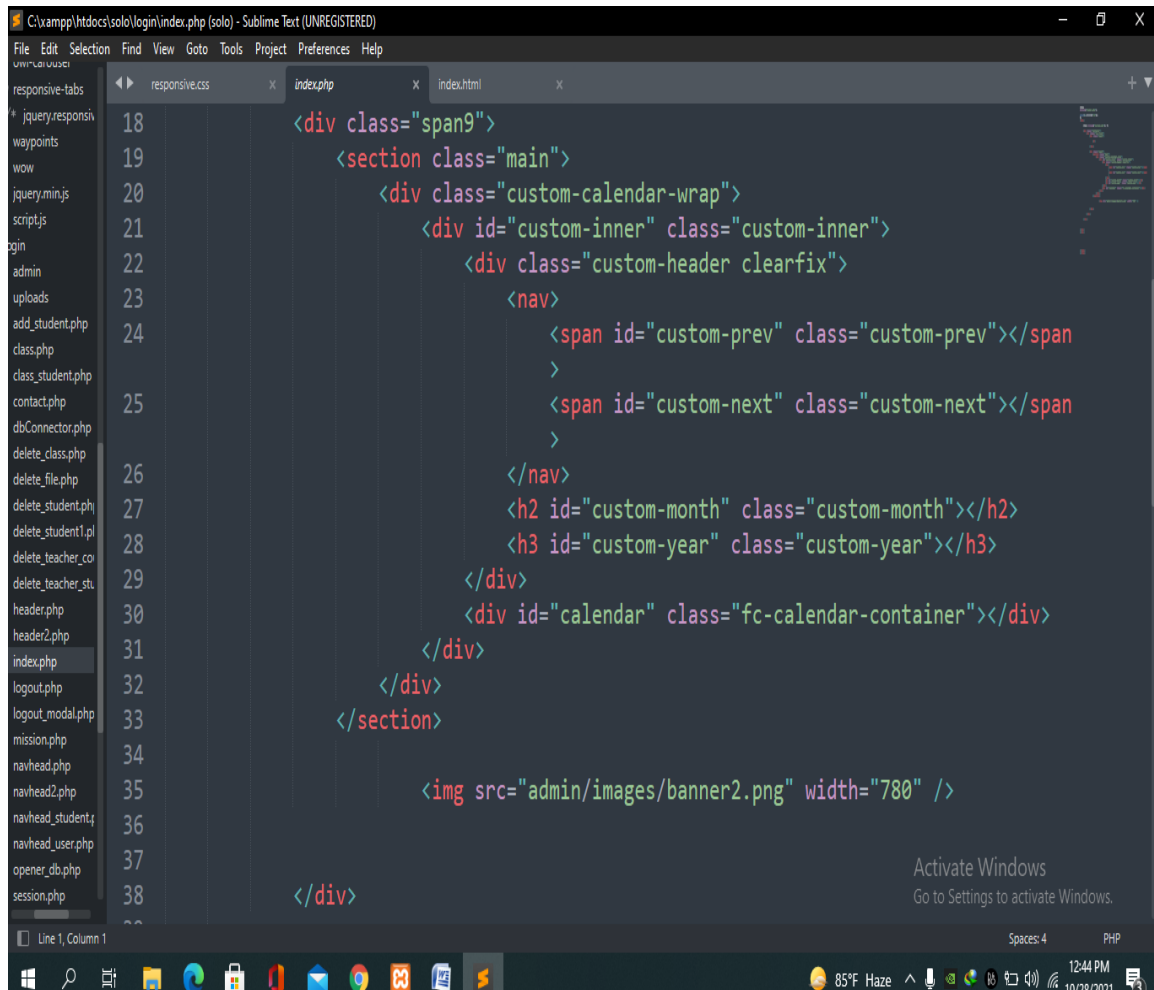


```
10 <title>SOLO</title>
11 <link href="admin/css/bootstrap.css" rel="stylesheet" type="text/css" media="
screen">
12 <link href="admin/css/bootstrap-responsive.css" rel="stylesheet" type="text/
css" media="screen">
13 <link href="admin/css/font-awesome.min.css" rel="stylesheet" type="text/css"
media="screen">
14 <link rel="stylesheet" type="text/css" href="admin/css/DT_bootstrap.css">
15 <?php include('admin/connect.php'); ?>
16 </head>
17 <script src="admin/js/jquery.js" type="text/javascript"></script>
18 <script src="admin/js/bootstrap.js" type="text/javascript"></script>
19
20 <script type="text/javascript" charset="utf-8" language="javascript" src="admin/
js/jquery.dataTables.js"></script>
21 <script type="text/javascript" charset="utf-8" language="javascript" src="admin/
js/DT_bootstrap.js"></script>
22 <script type='text/javascript' language='javascript' src='js/ndhui.js'></script>
23
24
25 <script type="text/javascript" src="admin/js/jquery.nivo.slider.js"></script>
26 <script type="text/javascript">
27     jQuery(document).ready(function() {
```

Figure 4.3.25: header file Code

### 4.3.26 Index

In figure 4.3.26 it's an index function code. All HTML front end, CSS design & link to bootstrap & other framework is made in this index term. All contents are written in this term. The navigation bar is also created in this function by connecting with javascript. Font awesome is used for type & framework used as code ignitor.



```
18 <div class="span9">
19 <section class="main">
20 <div class="custom-calendar-wrap">
21 <div id="custom-inner" class="custom-inner">
22 <div class="custom-header clearfix">
23 <nav>
24 <span id="custom-prev" class="custom-prev"></span
25 >
26 <span id="custom-next" class="custom-next"></span
27 >
28 </nav>
29 <h2 id="custom-month" class="custom-month"></h2>
30 <h3 id="custom-year" class="custom-year"></h3>
31 </div>
32 <div id="calendar" class="fc-calendar-container"></div>
33 </div>
34 </section>
35 
36
37
38 </div>
```

Figure 4.3.26: Index html Code

### 4.3.27 Responsive View

The screenshot shows a Sublime Text editor window with the following CSS code in the responsive.css file:

```
1  /* ===== Mobile Menu =====  
2  ===== */  
3  ===== */  
4  
5  .navbar-header {  
6      position: relative;  
7  }  
8  
9  #mobile-nav-open-btn {  
10     font-size: 30px;  
11     color: #f4c613;  
12     cursor: pointer;  
13     z-index: 2;  
14     position: absolute;  
15     right: 0;  
16     top: 6px;  
17     /* hide mobile nav open btn */  
18     display: none;  
19 }  
20  
21 #mobile-nav {  
22     /* by default, mobile nav will be hidden with height 0% */  
23     height: 0%;  
24 }
```

Figure 4.3.27.1: Responsive View Code

The screenshot shows a Sublime Text editor window with the following CSS code in the responsive.css file:

```
91  
92  
93  /* Large Devices (Desktops & Laptops) */  
94  
95  @media (min-width: 1200px) and (max-width: 1299px) {  
96     /* Navigation */  
97     .site-nav-wrapper {  
98         padding: 0 20px;  
99     }  
100 }  
101  
102  
103  /* Medium Devices (Landscape Tablets & Medium Desktops) */  
104  
105  @media (min-width: 992px) and (max-width: 1199px) {  
106     /* Navigation */  
107     .site-nav-wrapper {  
108         padding: 0;  
109     }  
110     /* Generic */  
111     .vertical-heading h2 {  
112         font-size: 29px;  
113     }  
114 }
```

Figure 4.3.27.2: Responsive View Code

## **CHAPTER 5**

### **TESTING AND IMPLEMENTATION**

#### **5.1 Testing and Implementation:**

The environment in which our users and developers use this application can vary. Therefore, testing in a multitude of environments is a practical idea to ensure that the application runs smoothly as expected no matter where it's being used and regardless of who is using this. [4]

Test implementation is the process of developing and prioritizing test procedures, creating test data and, optionally, preparing test harnesses and writing automated test scripts. This is when tests are organized and prioritized and when test designs are implemented as test cases, test procedures, and test data. [5]

#### **5.1.1 Unit Testing**

Unit testing is used for when the coding is implementing then the developer checked for his own primary mistakes. So, in our case we have checked when we were coding and tried to find out the initial errors. It happens when we were coding and then test instantly.

#### **5.1.2 Integration Testing:**

We have firstly done the front-end design then we have included all the back-end code. Then we have plaid that if any kind of layout is broken or not. And also checked that all interface works perfectly or not.

### **5.1.3 Functional Testing:**

Initially we have selected all the function that we are required for our application then we have finished all the initial requirement of this project which need to be. And all the functions have been tested individually. Because functions are the main important part for a system. If function does not work properly then the system will be insignificant.

### **5.1.4 System Testing:**

System testing measures if the application working on other environments. As our project is fully web-based so we have tested every browser and other operating systems. Also, we have checked using different devices.

### **5.1.5 Usability Testing:**

Performance testing measures user performance and how friendly the UI is. So, to do that we have a survey of a few users and get their information in the UI and how many can easily navigate that easily.

## **5.2 Test Result:**

We have done several types of tests only for a specific purpose and after performing all the tests there is an effect. Based on these results we may obtain confirmation regarding our system and it is both good and bad. Results are provided below:

### **5.2.1 Result of Unit Testing:**

The complete project is based on coding. There are thousands of lines of code we have made on our own project. In this large number of line code it is quite possible to make mistakes or errors. In order to detect these types of errors we performed a unit test. We find some errors too mistakes we got a second chance to fix them. We resolved them immediately.

### **5.2.2 Result of Integration Testing:**

The integration test was a bit of a shock to what the code we originally wrote again back-end design, are those code-setting ones that work well. Because both designs they are the most important factor in any type of project. Let's say we did the best we could looking front-end design but there are several errors in the back-end and front-end and back-end design is not the same and at the same time what the users will say. So, to avoid that kind of problem we did an integration test. And no error was found during that test.

### **5.2.3 Result of Functional Testing:**

Functional testing was very important. Because we did well in the front and in the back-end design but we have made a mistake in working code. So, that is the total code value icon useless. Functional testing was a way to know if everything was working properly or not. We inspected all the work. During this test we had a bad one the login code and the registration page are facing some issues. There was a logical mistake. Then we fixed them. Another front-end management area was a cache problem in browser. This problem was also overcome by removing old files and installing a new one. Therefore, we have completed our performance testing.



#### **5.2.4 Result of System Testing:**

It was very necessary to check because we know that most people use windows operating system. So, we did a system test. Then we can examine what we look at if our app worked successfully on auxiliary systems and locations. So, there no error found.

#### **5.2.5 Result of Usability Testing:**

So this test has worked for different people. So that most of them have access to this program easily. But some of them can't find the quiz option and there was a problem for the teachers to find out upload option. But after a few minutes of using it they can get it and they can use it without difficulties.

## **CHAPTER 6**

### **CONCLUSION AND FUTURE SCOPE**

#### **6.1 Conclusion & Limitations:**

Everything is going to be online nowadays and if any institute becomes only offline based there is a great chance that that will fail to maintain all its' functionalities. We have observed very clearly that requirement last year in the COVID-19 pandemic. If any offline institution has an online form that will be a great backup for any future obstacles. And if an online platform is there where the teacher and students can be connected 24/7, they can perform tests and have classes conducted we think the main requirement of studies is fulfilled. Our project has served this need. This is a complete solution for an unwanted situation. And also a great help to maintain online activities parallel with

offline which will save much time, expense and will be easy for both teacher and students.

There are some limitations we had to face during developing this project. For COVID-19, corona lockdown made us do the whole project physically disconnected from each other. All the works were done by meeting on Google meet online platform, that's why we could not give the best output of us. One of the main problems was we had to solve one's problem by another of us getting source code through mail and Google drive. If we had to update anything the whole thing was sent to one of us and others had to wait for his work done. When he could solve or update that then the other members could start from the next step. Although we tried to make it as perfect as it could be but some limitations still may be remained without our acknowledgment. Also after publishing this project we will try to remove those limitations if we are informed.

## **6.2 Future scope:**

This project has a large scope in the future. As education become a basic need, its' necessity will never end. Besides, the internet also becoming popular day by day in geometric increment. So, this kind of project is going to be a huge part of the education sector. Our project can also be a runner on that track. In the future, this project can be updated to be more popular and user-friendly. We will take steps which are included below:

- Develop an android system too
- Assignment submission option
- Students self-enrollment system
- Taking attendance
- More lively & digitally monitoring system

## References

- [1] Unique School Management System. Available at <https://www.uc-school.com> . [Last accessed at 10.10.2021 at 8.00 pm]
- [2] Eskooly. Available at <https://www.eskooly.com> . [Last accessed at 10.10.2021 at 8.30 pm]
- [3] Robi 10-minute School. Available at <https://10minuteschool.com/contact/>. [Last accessed at 02.11.2021 at 12.30 pm]
- [4] Testing and implementation. Available at <https://archerpoint.com/importance-software-testing-during-implementation/> [Last visited: 28.10.2021 at 10.00 pm]
- [5] Testing and implementation process. Available at <https://cania-consulting.com/2019/10/25/test-implementation-and-test-execution/> [Last visited: 28.10.2021 at 10.20 pm]