** Padmakanya Multiple Campus**

Bagbazar, Kathmandu

**Tribhuvan University**

**DIGITAL ASSIGNMENT**

**A PROJECT REPORT**

**Submitted to**

**Department of Computer Science**

**Padmakanya Multiple campus**

***In partial fulfillment of the requirements for the Bachelors in Computer Science and Information Technology***

**Submitted By:**

Jeena Sherma(20292)

Binita Subedi(20289)

Rojina Dangi(20306)

Under the Supervision of

**Sudip Raj khadka**

** Padmakanya Multiple Campus**

Bagbazar, Kathmandu

**Tribhuvan University**

**Supervisor’s Recommendation**

I hereby recommend that the forthcoming project report prepared under my supervision by **Binita Subedi**, **Jeena Sherma** and **Rojina Dangi** entitled **"DIGITAL ASSIGNMENT MANAGEMENT SYSTEM"** in partial fulfillment of the requirements for the degree of Bsc CSIT is recommended for the final evaluation.

………………………………

**Mr Sudip Raj Khadka**

Project Supervisor

Bsc CSIT Department

Padmakanya Multiple Campus

Bagbajar, Kathmandu

** Padmakanya Multiple Campus**

Bagbazar, Kathmandu

**Tribhuvan University**

**LETTER OF APPROVAL**

This is to certify that this project prepared by **Jeena Sherma(20292) , Rojina Dangi(20306) and Binita Subedi(20289)** entitled **“DIGITAL ASSIGNMENT”** in partial fulfillment of the requirements for the degree of Bachelor in Computer Science and Information Technology has been evaluated. In our opinion it is satisfactory in the scope and quality as a project for the required degree.

|  |  |
| --- | --- |
| …………………………..  **Mr.** **Sudip Raj Khadka**  Supervisor /Lecturer  Department of Computer Science  Padmakanya Multiple Campus  Bagbazar, Kathmandu, Nepal | …………………………..  **Mr. Ramesh Singh Saud**  Coordinator  Department of Computer Science  Padmakanya Multiple Campus  Bagbazar, Kathmandu, Nepal |
| …………………………..  **Internal Examiner** | …………………………..  **External Examiner** |

**AKNOWLEDGEMENT**

First and foremost, we would like to express our special thanks of gratitude to our supervisor “**SUDIP RAJ KHADKA**” who guided us in doing this project. We would like to earnestly acknowledge the sincere efforts and valuable time given by him. He provided us with invaluable advice and helped us in difficult periods. His motivation and help contributed tremendously to the successful completion of the project. Any attempt at any level can‘t be satisfactorily completed without the support and guidance of our supervisor.

Besides, we would like to thank all the teachers who helped us by giving us advice and providing the equipment which we needed.

Then we would like to express special gratitude to our friend Mr. Rajneesh Bhandari and all the classmates who have helped us with their valuable suggestions and guidance at various stages of completion of this project.

We would also like to thank everyone who helped and motivated us to work on this project.

**ABSTRACT**

Digital Assignment is a software which is helpful for students as well as the school authorities. It streamlines the assignment creation, submission, grading, and feedback processes, providing an efficient and user-friendly platform for both students and instructors. This system makes it easy for us to do this task from anywhere, at any time, with the help of the internet. With intuitive features such as assignment creation, automated grading, and seamless integration with existing learning management systems, Digital Assignment offers a comprehensive solution for colleges seeking to optimize their assignment management workflow. In the current system all the activities are done manually and it is very time consuming and difficult to manage. We cannot keep the record of student assignments easily with the present system and is based on pen and paper. The existing system is very difficult and has a long process. With the new system, it is likely that students' knowledge will soar up as they become more efficient in submitting assignments on time because of the deadline. Additionally teachers can provide feedback regarding the submitted assignments by the students and students can view the feedback generated by the teachers. Also teachers can keep the record of the details of student’s assignment marks.

***Keywords: Assignment, Digital, Feedback.***

**TABLE OF CONTENTS**

[CHAPTER 1 1](#_Toc135771174)

[INTRODUCTION 1](#_Toc135771175)

[1.1 Overview 1](#_Toc135771176)

[1.2 Statement of Problem 2](#_Toc135771177)

[1.3 Objectives 2](#_Toc135771178)

[1.4 Scopes and Limitation 2](#_Toc135771179)

[1.5 Development Methodology 3](#_Toc135771180)

[1.6 Report Organization 4](#_Toc135771181)

[CHAPTER 2 5](#_Toc135771182)

[BACKGROUND STUDY AND LITERATURE REVIEW 5](#_Toc135771183)

[2.1 Background Study 5](#_Toc135771184)

[2.2 Literature Review 6](#_Toc135771185)

[CHAPTER 3 7](#_Toc135771186)

[SYSTEM ANALYSIS AND DESIGN 7](#_Toc135771187)

[3.1 System Analysis 7](#_Toc135771188)

[3.1.1 Requirement Analysis 7](#_Toc135771189)

[3.1.2 Feasibility Analysis 9](#_Toc135771190)

[3.1.3 Object Modelling 10](#_Toc135771191)

[CHAPTER 4 16](#_Toc135771192)

[SYSTEM DESIGN 16](#_Toc135771193)

[4.1 System Design 16](#_Toc135771194)

[4.1.1 Component Diagram 16](#_Toc135771195)

[4.1.2 Deployment Diagram 17](#_Toc135771196)

[4.2 Algorithm Details 18](#_Toc135771197)

[3.3.1 Tracking Algorithm 18](#_Toc135771198)

[3.3.2 Tracking Algorithm 18](#_Toc135771199)

[CHAPTER 5 20](#_Toc135771200)

[IMPLEMENTATION AND TESTING 20](#_Toc135771201)

[5.1 Implementation 20](#_Toc135771202)

[5.1.1 Tools Used 20](#_Toc135771203)

[5.1.2 Implementation Details of Modules 21](#_Toc135771204)

[5.2 Testing 21](#_Toc135771205)

[5.2.1 Unit Testing 21](#_Toc135771206)

[5.2.2 System Testing 28](#_Toc135771207)

[CHAPTER 6 31](#_Toc135771208)

[CONCLUSION AND FUTURE RECOMMENDATIONS 31](#_Toc135771209)

[6.1 Conclusion 31](#_Toc135771210)

[6.2 Future Recommendations 31](#_Toc135771211)

[REFERENCES 32](#_Toc135771212)

[APPENDICES 33](#_Toc135771213)

[Appendix A: Screen Shots 33](#_Toc135771214)

[Appendix B: Source Code 44](#_Toc135771215)

**LIST OF FIGURES**

[Figure 1 Waterfall Model 3](#_Toc120214427)

[Figure 2 Use Case Diagram 9](#_Toc120214428)

[Figure 3 Object Diagram 11](#_Toc120214429)

[Figure 4 Class Diagram 12](#_Toc120214430)

[Figure 5 Admin Sequence Diagram 13](#_Toc120214431)

[Figure 6 Teacher Sequence Diagram 14](#_Toc120214432)

[Figure 7 Student Sequence Diagram 15](#_Toc120214433)

[Figure 8 Activity Diagram 16](#_Toc120214434)

[Figure 9 Component Diagram 17](#_Toc120214435)

[Figure 10 Deployment Diagram 17](#_Toc120214436)

**LIST OF TABLES**

[Table 2 Testing Registration Form 20](#_Toc120214437)

[Table 3 Testing Login Form 21](#_Toc120214438)

[Table 4 Testing Admin Dashboard 22](#_Toc120214439)

[Table 5 Testing Teacher Dashboard 23](#_Toc120214440)

[Table 6 Testing Student Dashboard 25](#_Toc120214441)

[Table 7 System Testing 25](#_Toc120214442)

# CHAPTER 1

# INTRODUCTION

## 1.1 Overview

A digital assignment refers to an educational task or project that is completed and submitted electronically using digital tools and technologies. Unlike traditional pen-and-paper assignments, digital assignments leverage the power of technology to enhance the learning experience, foster creativity, and facilitate collaboration among students. Digital assignments can take various forms, including written essays, multimedia presentations, online quizzes, coding projects, video submissions, and more. These assignments often require students to utilize digital resources, software applications, and online platforms to complete their work. Digital Assignment is a software which is helpful for students as well as the school authorities. A Digital Assignment is a software that helps the teachers to schedule tasks and assign them to the respective students. Additionally, deadlines are provided to the students, and after the submission, they are evaluated by the teachers. In the current system all the activities are done manually and it is very time consuming and difficult to manage. Our System deals with the various assignment activities related to the students. We cannot keep the record of student assignments easily with the present system and is based on pen and paper. The existing system is very difficult and has a long process. Our system deals with such major problems and provides an online platform to the students to submit their assignment and view the assignment status details virtually. As the whole process does not require many manual efforts, the system can be termed “Time Saver.” As the projects and assignments can be given to the students online, no extra efforts need to be taken [1].

There are mainly 3 modules in this software

• Admin Module

• Teacher Module

• Student Module

In the software, Admin can login to the system through proper authentication and can gain access to all the system. Admin adds the Subjects, Teachers and Students and also assign them to their particular semesters. Admin can also update the students and teachers details. He assigns the teacher for different subjects accordingly.

In the Software we can register as a user and user has of two types, Student and Teacher. A teacher can register as user, but the email must be the same as that provided by an administrator. Authorized teacher can get the benefits of creating assignments with ease. Some additional resources like docs, Pdfs, images, etc. can be attached in a few seconds. Ultimately, this allows teachers to spend less time on these activities and utilize much in teaching their students. Teachers can quickly review the projects as well as the assignments uploaded by students. The teacher can review and grade accordingly. Also the teachers can keep the record of the student’s assignments marks which they receives by submitting the assignment.

Similarly as teacher, a student can register as user, with the email same as that provided by an administrator. Any subject data can be easily accessed by the students. Students can make use of resources like PDFs, Docs, images which are uploaded by the teachers. Student can also know whether his assignment is checked or not. He can manage both his checked and unchecked assignments. Also teacher can provide the necessary feedbacks to the students and can provide marking to the students. The feedbacks generated by the teacher can be viewed by the students.

## 1.2 Statement of Problem

The traditional way of submitting assignments and notes has a lot of limitations. The existing system has increased a lot of paper works and also it has very high chances of paper loss. The existing system is time consuming and not easily accessible. We cannot keep the record of student assignments with the present system and is based on pen and paper. The existing system is very difficult and has a long process.

## 1.3 Objectives

The main objectives of this project are:

* To avoid an excessive paper work load.
* To ensure the secure and loss of submitted papers.
* To track student's assignments activities records
* To make assignments available at any time and by easiest means.

## 1.4 Scopes and Limitation

Since each educational institution are quickly moving into online framework, it appears to be very successful and useful to embrace Digital Assignment in each educational institution. Numerous schools/organizations are willing to induce such framework which will offer assistance to perform errand through online system. Overall this may provide the environment of saving time and focusing and concerning students completely virtually. As this is planning to be made in lower cost and basic to utilize we are hoping that even small institute or organizations can bear to use our system.

## 1.5 Development Methodology

**Waterfall Model**

In the Waterfall approach, the whole process of software development is divided into separate phases. In this Waterfall model, typically, the outcome of one phase acts as the input for the next phase sequentially. Our system is short and has no ambiguous requirements. Our system requirements are very well documented, clear and fixed. The technology is understood and is not dynamic. The project costs are low and there is minimal risk. So we used the waterfall model to build the system.



Figure 1 Waterfall Model

## 1.6 Report Organization

Our document for this project is divided into following chapters:

**Chapter 1: Introduction**

This part of this report is occupied by introduction of our project, statement of the problem and scope and objectives are discussed.

**Chapter 2: Background Study and Literature Review**

This section of our report covers the research we made for this project like Background Study and Literature Review.

**Chapter 3: System analysis**

This section of our report covers the research we made for this project like feasibility analysis and system requirements.

**Chapter 4: System Design**

This section of our report covers the designing and development phase of our project.

**Chapter 5: Implementation and Testing**

This section of our report covers the implementations of various methods and tools while developing this system.

**Chapter 6: Conclusion and Enhancement**

This section of our report covers the final outcome of entire project according to the problem statement and objectives.

# CHAPTER 2

# BACKGROUND STUDY AND LITERATURE REVIEW

## 2.1 Background Study

"Digital Assignment" is one of the essential tasks in education. The need for teachers and students to access the E-learning system from any location in the globe has arisen as a result of the new normal, which is built on the foundation of online mode. In fact, the assignment management system makes it much easier for us to manage the assignments or projects.

An advantage of an efficient Digital Assignment is that it provides teachers and students with high-quality educational impressions. It enables both sides to receive feedback, exchange ideas, and utilize the wealth of web resources available. In order to manage all projects and assignments effectively. For instructional reasons, this online assessment tool is just too useful.

Research has highlighted the time-consuming nature of traditional assignment management methods, such as manual grading and paper-based submissions, which can hinder efficiency and limit timely feedback. Additionally, studies have emphasized the need for enhanced collaboration and communication between instructors and students, especially in the context of online or blended learning environments.

An assignment management system is a software that helps the teachers to schedule tasks and assign them to the respective students. Additionally, deadlines are provided to the students, and after the submission, they are evaluated by the teachers.

All the assignments, as well as the projects, come up with start and end dates. It facilitates the teachers to view the status of the assigned task and generate reports accordingly.

As the whole process does not require many manual efforts, the system can be termed “Time Saver.” As the projects and assignments can be given to the students online, no extra efforts need to be taken.

The teacher can avail of the benefits of creating assignments with ease. Some additional resources like docs, Pdfs, images, etc. can be attached in a few seconds.

Ultimately, this allows teachers to spend less time on these activities and utilize much in teaching their students. Teachers can quickly review the projects as well as the assignments uploaded by students. The teacher can review and grade accordingly. Students are even graded for their learning and Improvement [2].

## 2.2 Literature Review

**2.2.1** **Digital Tools and Platforms**

Numerous studies have explored the effectiveness of various digital tools and platforms in supporting assignment creation, submission, and assessment. These tools include learning management systems (LMS), online collaboration platforms, and multimedia authoring tools. Researchers have highlighted the importance of user-friendly interfaces, ease of integration with existing systems, and adaptability to different assignment types.

**2.2.2** **Automated Grading and Assessment**

Automation in grading and assessment has been a focal point of research. Scholars have investigated the use of machine learning algorithms, natural language processing, and rubric-based assessments to automate the grading process. The potential benefits include increased efficiency, reduced bias, and improved consistency in grading.

**2.2.2** **Feedback and Communication**

Literature emphasizes the significance of timely and constructive feedback in the assignment management process. Researchers have explored the use of digital tools to provide personalized feedback, audio or video comments, and peer feedback mechanisms. Effective communication channels between instructors and students, as well as opportunities for peer collaboration and discussion, have been identified as crucial elements for student engagement and learning outcomes.

**2.2.2** **Accessibility and Inclusivity**

The literature highlights the importance of ensuring accessibility and inclusivity in digital assignment management. Researchers have examined issues related to assistive technologies, accommodating diverse learning needs, and addressing potential barriers faced by students with disabilities.

# CHAPTER 3

# SYSTEM ANALYSIS AND DESIGN

## 3.1 System Analysis

### Requirement Analysis

#### Functional Requirements

Functional requirements identify the system deployment and system response to the specific problem and how the system is expected to behave in real life. This system mainly focuses on managing of student assignments digitally.

The functional requirements of “Digital Assignment” includes the following tasks: -

1. The system should allow the admin, teachers and students to login with the help of valid username and password.
2. The system should allow access to all the modules after the authentication.
3. The system should allow admin to add the subjects, students and teachers.
4. The system should allow Admin to assign subject teacher and semester.
5. The system should allow Teacher to add, view and delete the assignments.
6. The system should allow Teacher to provide feedback regarding the assignments submitted by students.
7. The system should allow Teacher to record the marks of the assignments submitted by students.
8. The system should allow Student to view and submit their assignment.
9. The system should allow Student to view the assignment status whether it has been checked or not.
10. The system should allow Student to view the feedback generated by the teacher.
11. The system should allow users to view their profile.

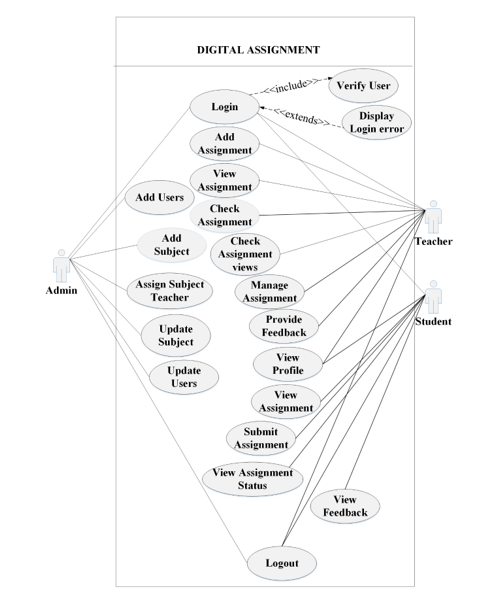


Figure 2 Use Case Diagram

#### Non-Functional Requirements

Non-Functional Requirements are the constraints or the requirements imposed on the system. They specify the quality attribute of the software.

* **Usability:**

Every user must be able to use the system. Users can easily navigate its interface in our system. Also users can easily determine what a feature is and what it can do.

* **Maintenance:**

The system must be able to be maintained. Our system can be easily maintained or recovered if there occurs any type of failures in the system.

* **Extendable and Scalability:**

For future enhancements, our system is expandable and adaptable.

* **Availability and Accessibility:**

At any time, our system is available and accessible. Our system can adjust to new conditions and has ability to access from anywhere with the help of internet.

* **Security:**

The system must be protected against unauthorized use and access. Our system allows access to only the authorized users after the proper authentication.

### Feasibility Analysis

A feasibility study is an analysis that considers all of a project's relevant factors including economic, technical, operational, and scheduling considerations to ascertain the likelihood of completing the project successfully.

#### Economic Feasibility

The development of this application is very economically feasible. The only thing to do is to create an environment with effective supervision. It is convenient in the sense that paperwork is completely eliminated. The system also saves time by automating the calculations that are made at the end of the month or according to the user's needs.

#### Technical Feasibility

The system operates on any device on which the browser is supported. The system is therefore technically feasible.

#### Operational Feasibility

The system is easy to use. The user does not require any special training to use the system. The system grants access to the user account after authentication.

#### iv. Schedule Feasibility

Schedule feasibility is the degree to which a deadline for a strategy, plan, project or process is realistic and achievable. The following table shows the project plan of our system.

### Object Modelling

#### Object Diagram



Figure 3 Object Diagram

#### Class Diagram



Figure 4 Class Diagram

The above class diagram shows that an admin can add 0 or many subjects and users. (Student and Teacher). Similarly one or many teacher can assign 0 or many assignments for the students and one or many students within that class submits 0 or 1 assignment. Also 1 teacher can teach 1 or many subjects.

**Dynamic Modeling:**

**Sequence Diagram**

##### **Admin Sequence Diagram**



Figure 5 Admin Sequence Diagram

The above Admin sequence diagram shows the sequence flow of admin in the system. The admin gets the access to add, update and delete subject, teacher and student whenever needed after the authentication with valid username and password while login. Also the admin can assign the subject teachers for different subjects.

##### **Teacher Sequence Diagram**



Figure 6 Teacher Sequence Diagram

The above Teacher sequence diagram shows the sequence flow of teacher in the system. The teacher gets the access to create, view and delete the assignments after the authentication with valid username and password while login. Also the teacher can provide the feedback to the students regarding the student’s assignment and can keep the records of the student’s assignments marks.

##### **Student Sequence Diagram**



Figure 7 Student Sequence Diagram

The above Student sequence diagram shows the sequence flow of student in the system. The students gets the access to view and submit the assignments created by their subject teacher after the authentication with valid username and password while login. Also the student can view the feedback generated by their teacher regarding the assignment.

**Process Modeling**

#### Activity Diagram

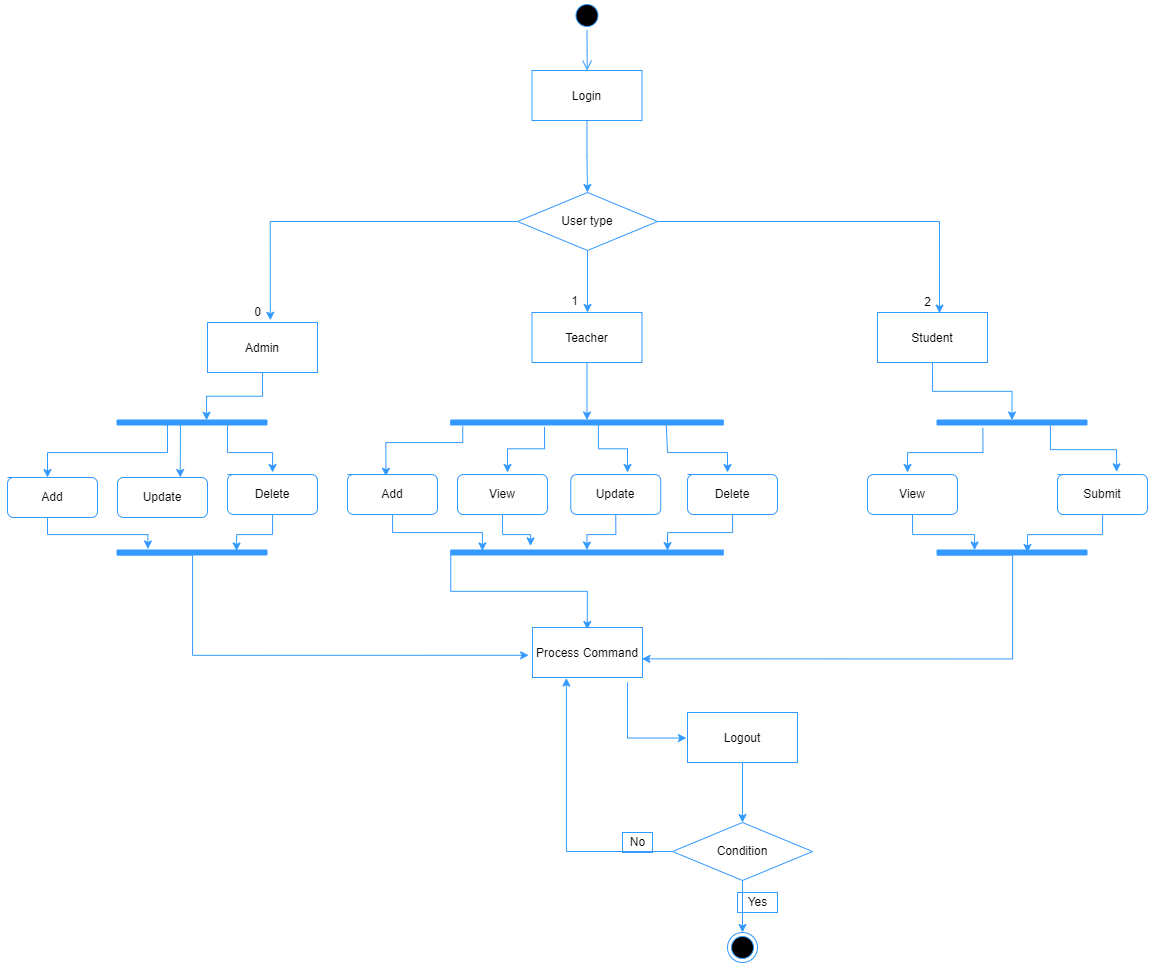


Figure 8 Activity Diagram

# CHAPTER 4

# SYSTEM DESIGN

## System Design

### 4.1.1 Component Diagram



Figure 9 Component Diagram

### 4.1.2 Deployment Diagram

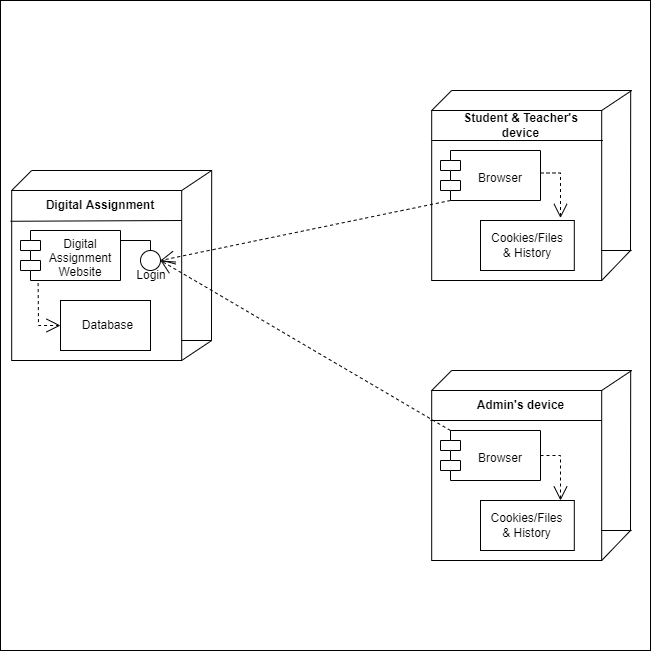


Figure 10 Deployment Diagram

## 4.2 Algorithm Details

### 3.3.1 Tracking Algorithm

A tracking algorithm is a type of algorithm that is used to monitor and analyze the movement or behavior of objects or entities over time. Tracking algorithms can be used in a variety of applications, such as surveillance, robotics, motion capture, and user behavior tracking. In the context of our website, the tracking algorithm we discussed is used to monitor which students have viewed a particular assignment. This information is useful for teachers to understand student engagement and to identify students who may need extra support or encouragement.

### 3.3.2 Tracking Algorithm

The Assignment View Tracking Algorithm is a tracking algorithm implemented in the online platform that allows teachers to monitor which students have viewed a particular assignment. The algorithm works by tracking the timestamps of when each student viewed the assignment, and then generating reports that show which students have viewed the assignment and when they did so. This information can help teachers to gauge student engagement and identify students who may need extra support or encouragement. However, it is important to ensure that user privacy is respected, and that users are informed about the data that is being tracked and how it is being used.

The main steps involved in the Assignment View Tracking Algorithm are:

Data Collection: The algorithm collects data on which students have viewed the assignment, as well as when they viewed it. This data is typically collected automatically by the platform, using cookies or other tracking technologies.

Data Processing: The algorithm processes the collected data to generate reports that show which students have viewed the assignment and when they did so. This may involve analyzing the timestamps of when the assignment was viewed by each student, and identifying patterns or trends in the data.

Data Visualization: The algorithm presents the results of the data processing in a user-friendly way, such as a table, chart, or other visualization. This allows teachers to easily see which students have viewed the assignment and when they did so, and to identify any students who may need extra support or encouragement.

Overall, the Assignment View Tracking Algorithm is a useful tool for teachers to monitor student engagement and to identify students who may need extra support. However, it is important to ensure that user privacy is respected, and that users are informed about the data that is being tracked and how it is being used.

# CHAPTER 5

# IMPLEMENTATION AND TESTING

## 5.1 Implementation

### 5.1.1 Tools Used

**Diagram Tools**

These tools are used to graphically represent system components, data and control flows between different software components and system tree. We used Visio and draw.io as our diagramming tool.

**Documentation Tools**

The documentation of a software project starts before the software process, goes through all the phases of the SDLC and after the project is finished. Here we used MS. Office 16 for our project documentation as Programming Tools

These tools consist of IDE programming environments. We used Visual Studio Code as a programming tool Web Development Tools. The web tools provide a live preview of what is being developed and what it will look like when finished. We used Google Chrome as a web development tool.

**Visual Studio Code**

We used Visual studio code as a code editor because it has support for debugging, syntax highlighting, smart code completion, snippets, and code refactoring. Also, we can change the theme, keyboard shortcuts, and settings and install extensions that add additional functionality.

**Hyper Text Mark-up Language (HTML)**

In order to execute and display the system in a web browser, we used HTML which is the standard marking language. In order to make the web system beautiful, friendly, attractive and fast, we have supported the CSS and the bootstrap.

**Java Script**

JavaScript is used for form processing and also for the general behavior of web pages.

**MySQL**

To store and manage the information of all system users, we used MySQL, an open source relational database management system. It runs on a server and we used the server version: 10.3.16-MariaDB.

**PHP**

We used PHP for web development (backend), because it is a server-side scripting language. It's a fast, flexible, and pragmatic tool for the world's most powerful websites. PHP version we use: 7.3.6.

**XAMPP**

We used XAMPP as a software platform that helps a local host or server test the system we have built. It is a platform that provides a suitable environment for testing and verifying operation of Apache, Perl, MySQL and PHP based projects through the host system.

### 5.1.2 Implementation Details of Modules

## 5.2 Testing

### 5.2.1 Unit Testing

Table 1 Testing Registration Form

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC  ID | Test Case Name | Test Case Description | Step | Expected Result | Actual Result | Status |
| TC  01 | Account sign up  form validation | Enter email provided by admin and fill other field to create an account. | Enter email provided by system admin and enter password | Display message “Successfully signup ” | Display message “Account created successfully” | Pass |
| TC  02 | Password Validation | Unmatched password | When no matched password | Display message “Password does not match.” | Display message “Password doesn’t match. Try entering similar password.” | Pass |

Table 2 Testing Login Form

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| TC ID | Test Case Name | Test Case User | Test Case  Description | Step | Expected Result | Actual  Result | Status |
| TC  03 | Login Validation | Admin | Enter valid username and password | Click Login as admin and  Enter valid username and password | Successful login or an error message “Invalid username or Password” have to be displayed. | Successful  Login  Directed to Admin dashboard. | Pass |
| TC  04 | Login Validation | Admin | Enter invalid username and password | Click login as admin and  Enter invalid username and invalid password | An error message “Invalid username or password...” must be displayed. | An error message “Invalid username or password.” was displayed. | Pass |
| TC  05 | Login Validation | Teacher | Enter valid username and password | Choose login as Teacher and  Enter username and password | Successful login or an error message “Invalid username or password” must be displayed. | Successful  Login  Directed to Teacher page. | Pass |
| TC  06 | Login Validation | Teacher | Enter invalid username and password | Choose login as Teacher and  Enter invalid username and password | An error message “Invalid User or Password” must be displayed. | An error message “Invalid username or password.” was displayed. | Pass |
| TC  07 | Login Validation | Student | Enter valid username and password | Click login as student and  Enter username and invalid password | Login Successful or an error message “Invalid username or password” must be displayed. | Successful  Login  Directed to Student page. | Pass |
| TC  08 | Login Validation | Student | Enter invalid username and password | Click login as student and  Enter invalid username and invalid password | An error message “Invalid username or password” must be displayed. | An error message “Invalid username or password.” was displayed. | Pass |

Table 3 Testing Admin Dashboard

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC ID | Test Case Name | Test Case Description | Step | Expected Result | Actual Result | Status |
| TC 09 | Add New Subject | Add New Subject | Add subject name and code | Successfully Add new Subject or display an error message “Subject is already added”. | Display message “Subject added successfully” | Pass |
| TC  10 | Insert Subject | Insert Subject that already exist in table. | Insert New Subject | Display an error message “Subject is already added”. | Display message “Subject added successfully” | Fail |
| TC  11 | Insert Subject | Insert Subject that already exist in table. | Insert New Subject | Display an error message “Subject is already added”. | Display an error message “Subject is already added”. | Pass |
| TC 12 | Add Teacher | Adding a new teacher | Add new teacher | Successfully add new Teacher or display an error message. | Teacher added successfully | Pass |
| TC  13 | Add an existing teacher | Try to add an existing teacher | Add teacher by filling forms | Teacher already exists. | Display message "Teacher already exists" | Pass |
| TC  14 | View Profile | View teacher profile | Click on a view details button | Display Profile | Profile displayed | Pass |
| TC  15 | Edit detail | Edit teacher detail | Click on an edit button and update new detail | Detail updated | Profile updated Successfully | Pass |
| TC  16 | Add Student | Adding a new Student | Add new student | Successfully add new Student or display an error message. | Student added successfully | Pass |
| TC  17 | Add an existing Student | Try to add an existing Student | Add Student by filling forms | Student already exists. | Display message " Student already exists" | Pass |
| TC  18 | View Profile | View Student profile | Click on a view details button | Display Profile | Profile displayed | Pass |
| TC  19 | Logout | End session | Logout to end session | Session end logout | Session end logout | Pass |

Table 4 Testing Teacher Dashboard

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Tc ID | Test Case Name | Test Case Description | Step | Expected Result | Actual Result | Status |
| TC  20 | Create assignment | Creating an assignment | Filling form | Assignment added successfully. | Assignment added successfully. | Pass |
| TC  21 | View assignment | View created assignment | Enter a Link | Display created assignment | Display created assignment | Pass |
| TC  22 | View Student's Assignment | View assignment submitted by students | Click on button to view student's assignment | Display submitted assignment by students | Assignment displayed | Pass |
| TC  23 | Check Assignment | Check assignment submitted by students | Click on button to check and give remarks | Checked | Checked | Pass |
| TC  24 | Provide Feedback | Provide feedback to student assignment. | Click on text field to provide feedback to the students. | Feedback is provided to students. | Feedback is provided to students. | Pass |
| TC  25 | Record Marks | Record marks of assignment submitted by students | Click on button to record the marks | Checked | Checked | Pass |
| TC  26 | Generate Average | Provide average marks of each student’s assignment. | Click on text field to provide feedback to the students. | Display average marks of each student’s assignment. | Average marks of each student’s assignment is displayed. | Pass |
| TC  27 | Check student's activity | Display the students who have submitted an assignment and vice versa | Click on view button | Students displayed | Student displayed | Pass |
| TC  28 | Logout | Logout to end session | Click logout button | Session end log out | Session end log out | Pass |

Table 5 Testing Student Dashboard

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC ID | Test Case Name | Test Case Description | Step | Expected Result | Actual Result | Status |
| TC  29 | View assignment | View assignment | Click on view button | Display assignment | Display assignment | Pass |
| TC  30 | Submit Assignment | Submit assignment | Upload a file by filling form | Successfully submitted | Assignment submitted successfully | Pass |
| TC  31 | Edit assignment | Edit submitted assignment | Click edit button to edit an assignment | Display an alert message Successfully edited | Displayed "Assignment edited successfully " | pass |
| TC  32 | Check assignment status and feedback. | Check assignment status and feedback. | Click on button to view | Display either checked or not checked and get feedback | Displayed 'checked' or 'not checked' status and got feedback | pass |
| TC  33 | logout | Logout to end session | Click logout button | Session end log out | Session end log out | pass |

### 5.2.2 System Testing

Table 6 System Testing

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| TC ID | Test Case Name | Test Case  Description | Step | Expected Result | Actual  Result | Status |
| TC  34 | Security  Testing | Checking Security to access the system | Login with registered username and password | Successful  Login  Directed to User dashboard. | Successful  Login  Directed to User dashboard. | Pass |
| TC  35 | Security  Testing | Checking Security to access the system | Try Login with unauthorized username and password | An error message “Invalid username or password” must be displayed. | An error message “Invalid username or password” displayed. | Pass |
| TC  36 | Security  Testing | Wrong or mistyped url | Try typing url that doesn't exist | Display 404 error message | Display 404 error message | Pass |
| TC  37 | Usability Testing | Duplicate assignment submission | Try submitting assignment of same assignment that is already submitted. | Display message “This assignment has already been submitted." | Display message “This assignment has already been submitted " | Pass |
| TC  38 | Usability Testing | Submit assignment after deadline | Try to submit assignment after deadline | You can't submit after deadline | Display message "You must submit before deadline." | Pass |
| TC  39 | Usability Testing | Calculate the no of students who've submitted an assignment. | Click on view button to check the result | Display the total no of students including number of students who've submitted and who've not submitted an assignment | Data displayed. | Pass |
| TC  40 | Usability Testing | View Assignment Status | Student Viewing their Attendance Status | Student Viewing their Attendance Status | Student Viewing their Attendance Status | Pass |
| TC  41 | Usability Testing | Teacher and Student changing their user detail. | Teacher and Student changing their user detail. | Teacher and Student changing their user detail. | Teacher and Student changing their user detail. | Pass |
| TC  42 | Load Testing | Testing Load of system | For demo test we have added 11 student and 5 teacher | System performs well | System performs well | Pass |
| TC  43 | Regression Testing | Testing new bugs during the development and changes. | Development and changes on code. | Bugs found and solved | Bugs found and solved | Pass |
| TC  44 | Migration testing | Migrating Project to another PC | Migrating Project to another PC | System run successfully on other device as well | System run successfully on other device as well | Pass |
| TC  45 | Functional Testing | Functional Testing | Make sure that functionality of the product is working as per the requirements defined. | All requirement fulfilled | All requirement fulfilled | Pass |

# CHAPTER 6

# CONCLUSION AND FUTURE RECOMMENDATIONS

## 6.1 Conclusion

The problems identified with the existing systems were solved well in line with our objective. Our system had reduced paperwork and was also safe from file loss and unauthorized access. By using this system both teacher and student can do their respective activities efficiently. Assignments were available at any time and could be easily accessed with a verified user login. Also teacher can generate the feedback regarding the assignments submitted by student and student can view the feedback provided by teacher. Now one can easily use this system as it is user friendly and responsive in web browser.

Using this system, the educational organizations will be able to save their time and effort to handle the student's assignments. Also the organizations can keep the record of the assignments of the students easily and can be easily accessible with proper authentication. In addition, teacher have the authority to assign marks for the assignments and keep the record of all the assignments.

## 6.2 Future Recommendations

This work can be further extended to global purposed system. The project has a very vast scope in future. In order to improve the effectiveness of the application to its greater height and full potential, it’s recommended that the following features should be added for future expansion of this project.

1. System available for global use
2. File and class note sharing.
3. User video chat/ conference system

# REFERENCES

|  |  |
| --- | --- |
| [1] | G. Meridian, "Vidyalaya," 12 01 2020. [Online]. Available: https://www.vidyalayaschoolsoftware.com/features/student-assignment-management-system. [Accessed 26 05 2022]. |
| [2] | A. M. S. |. H. I. I. B. F. SCHOOLS?, "EDU," 25 07 2018. [Online]. Available: https://www.myeducomm.com/blog/assignment-management-system-how-is-it-beneficial-for-schools/. [Accessed 02 06 2022]. |
| [3] | PHPGurukul, "Online College Assignment System Using PHP and MySQL," 09 04 2020. [Online]. Available: https://phpgurukul.com/online-college-assignment-system-using-php-and-mysql/. [Accessed 31 05 2022]. |
| [4] | BlackBoard, "Education Technology Solution," 22 08 2019. [Online]. Available: https://www.blackboard.com/en-eu. [Accessed 26 05 2022]. |

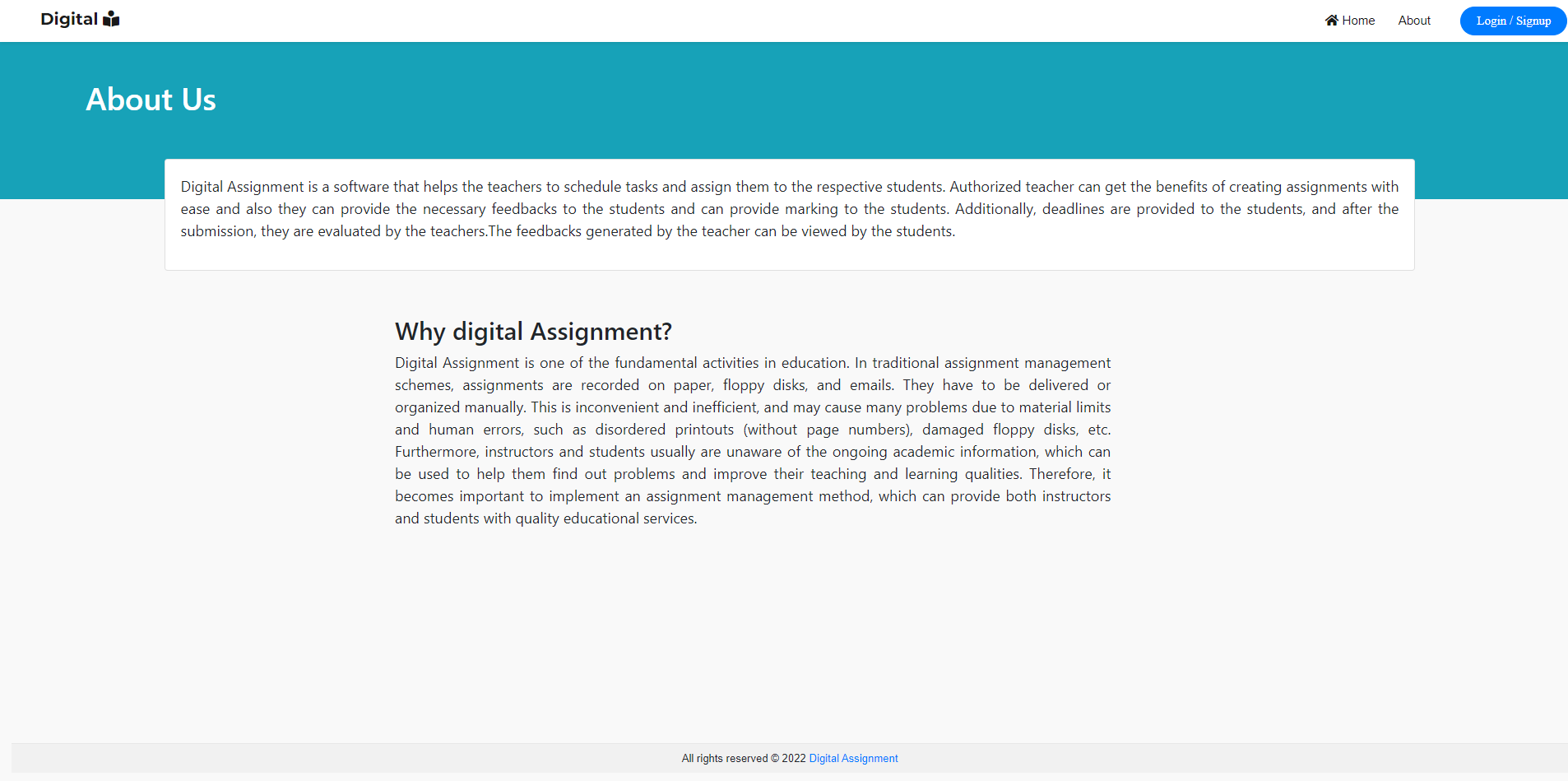
# APPENDICES

## Appendix A: Screen Shots

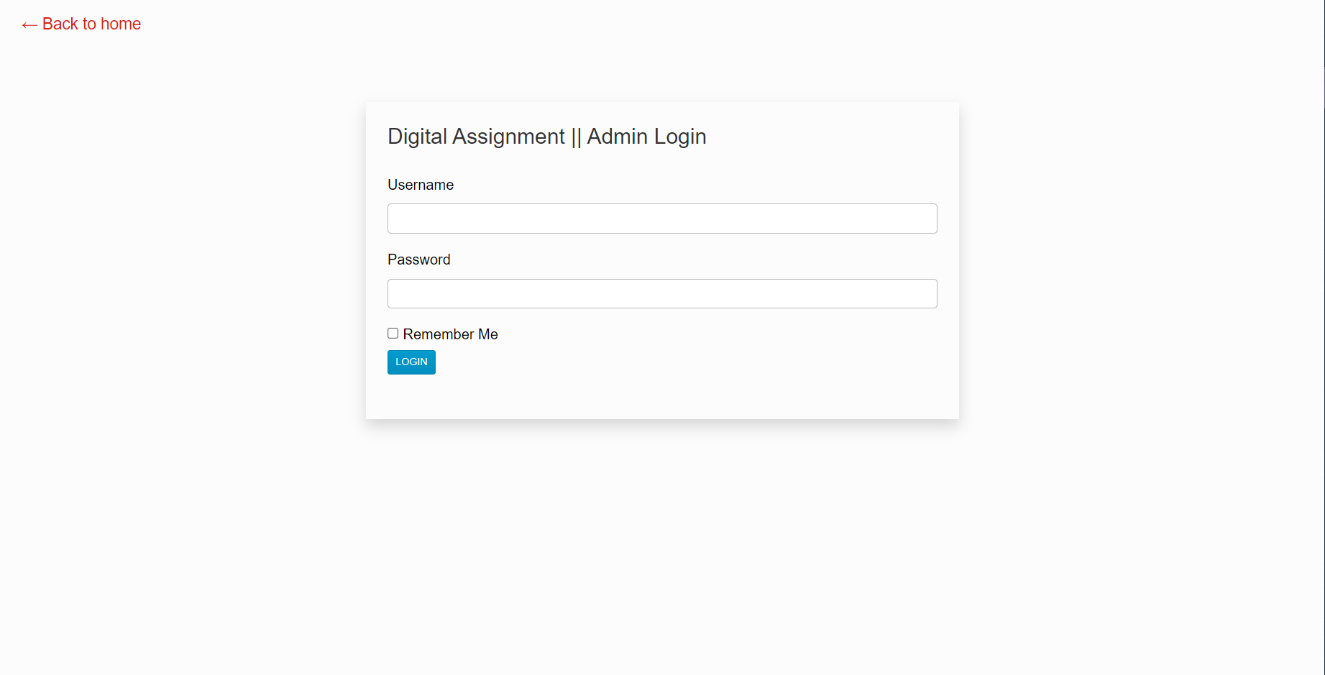
1. **Homepage**



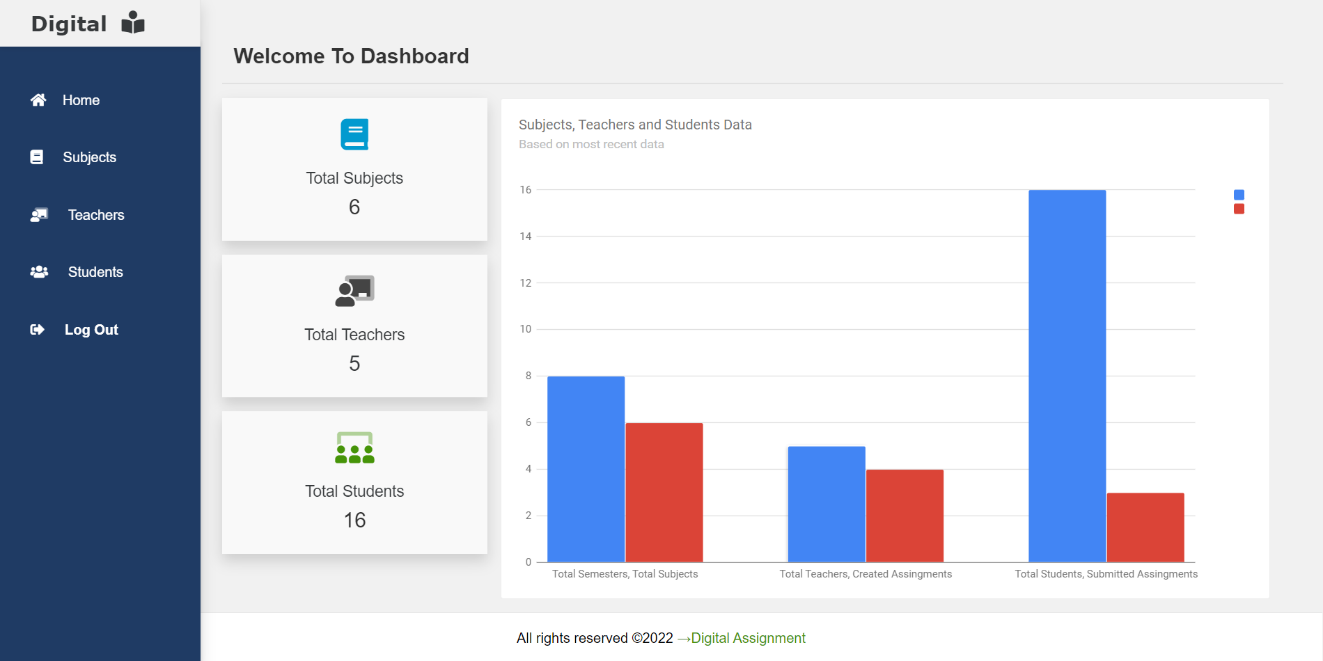
1. **About Us page**



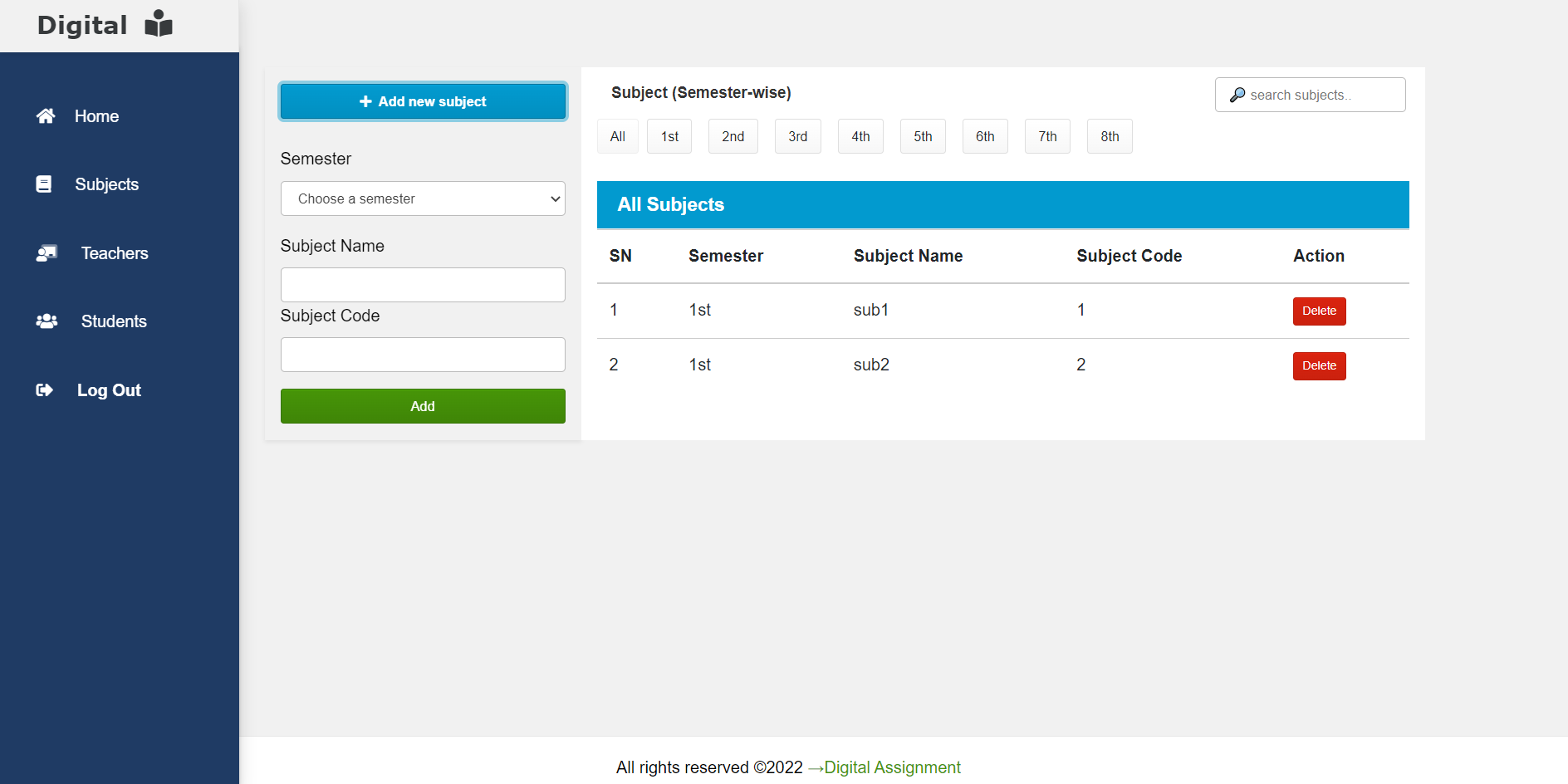
1. **Admin Login Page**



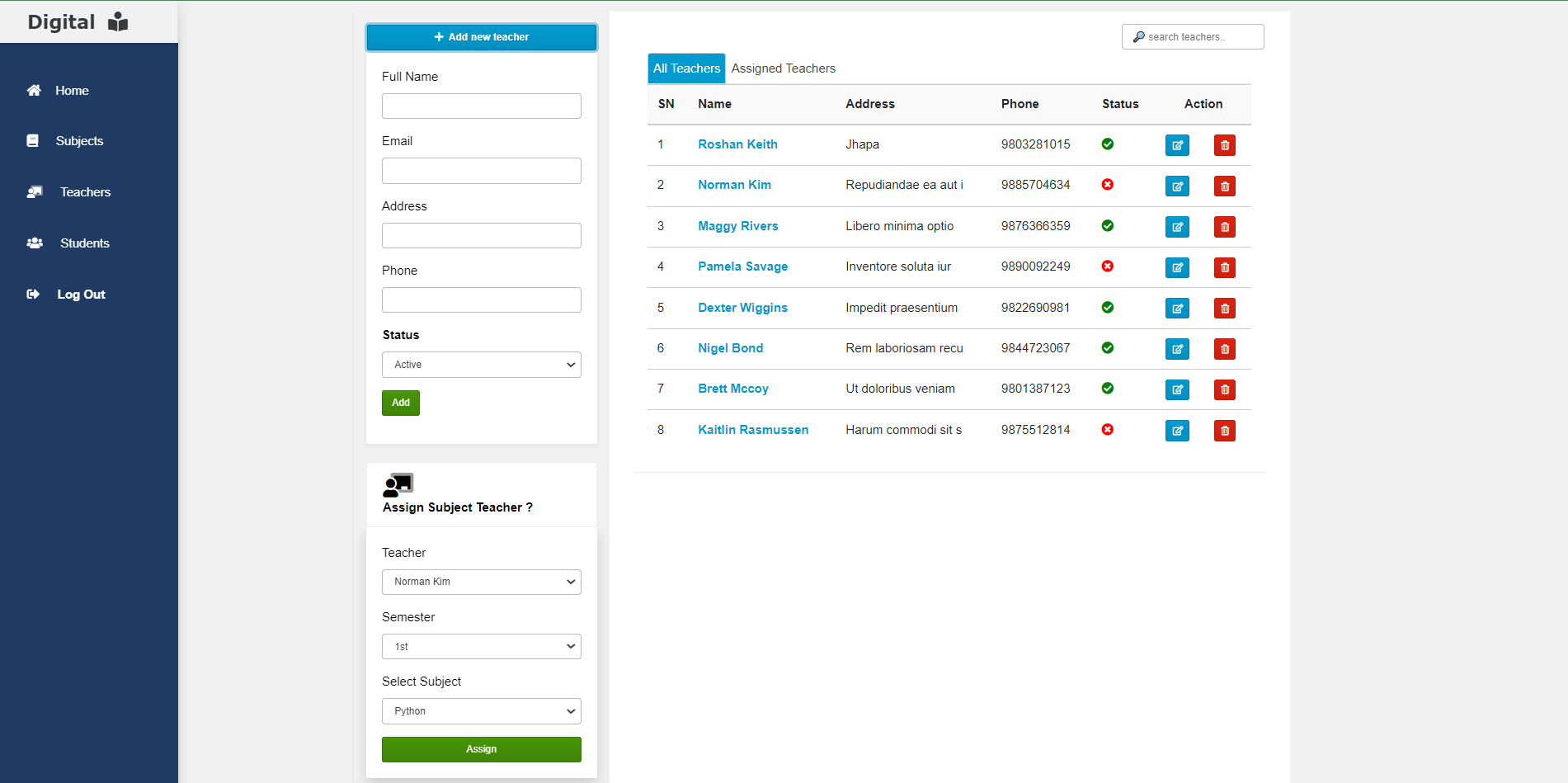
1. **Admin Dashboard**



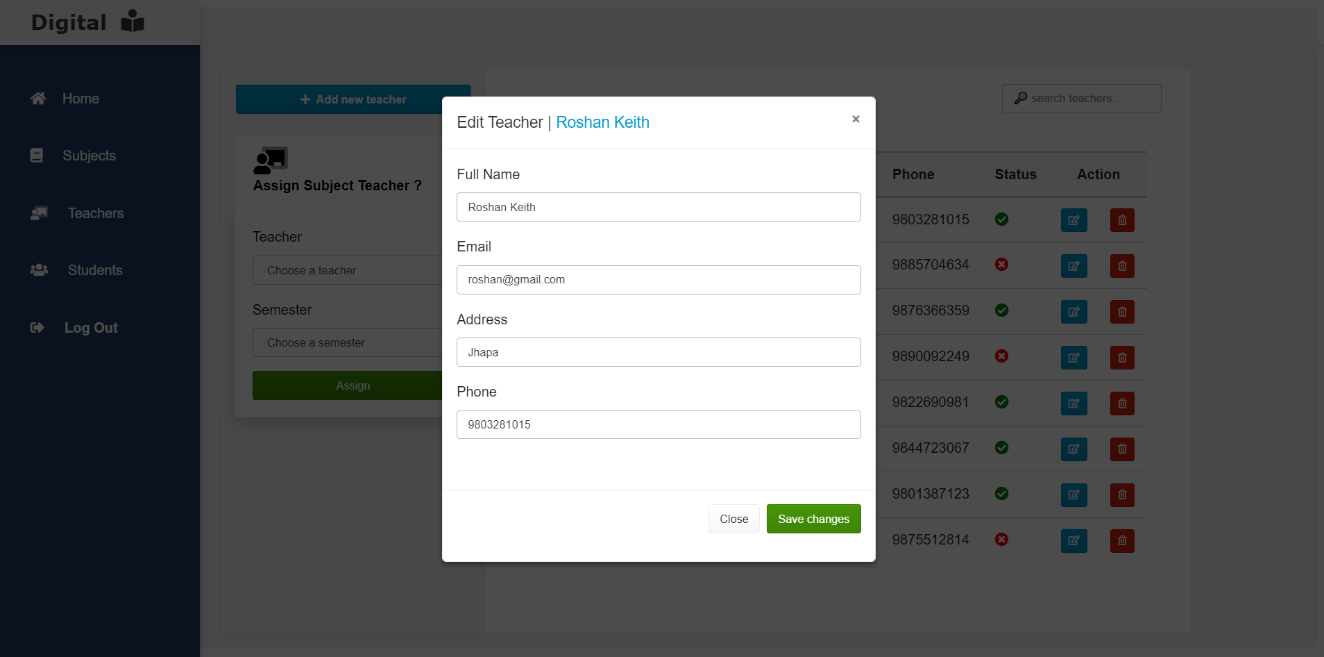
1. **Add Subject**



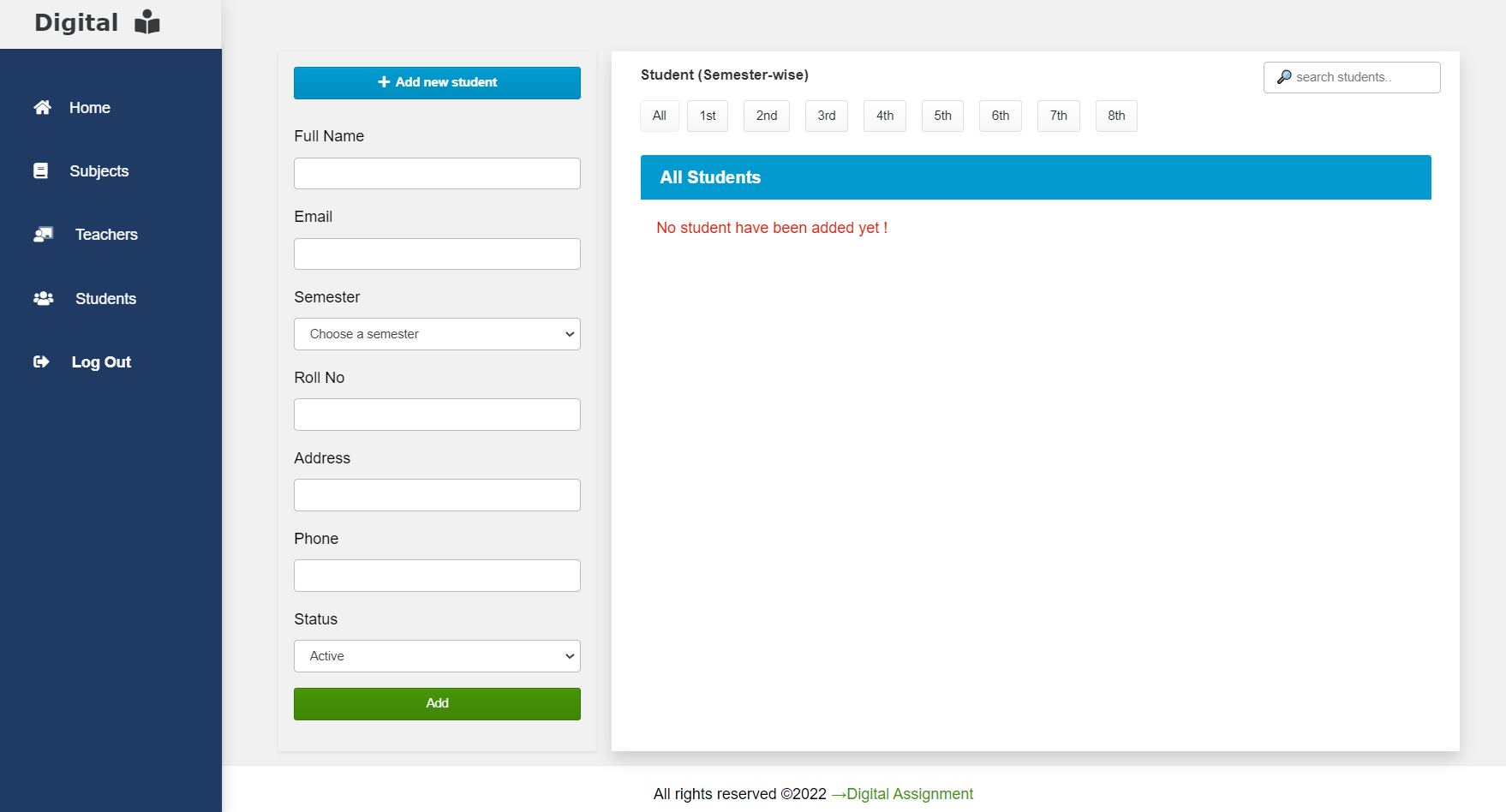
1. **Add and Assign Teacher**



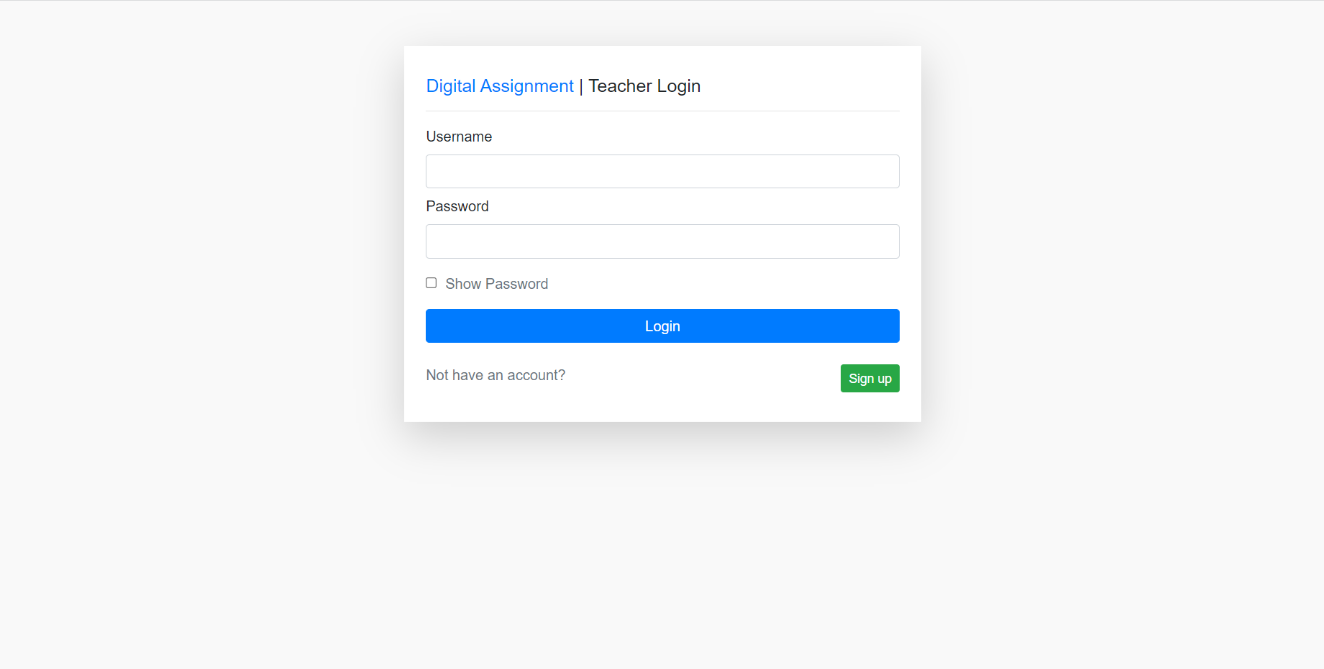
1. **Edit teacher**



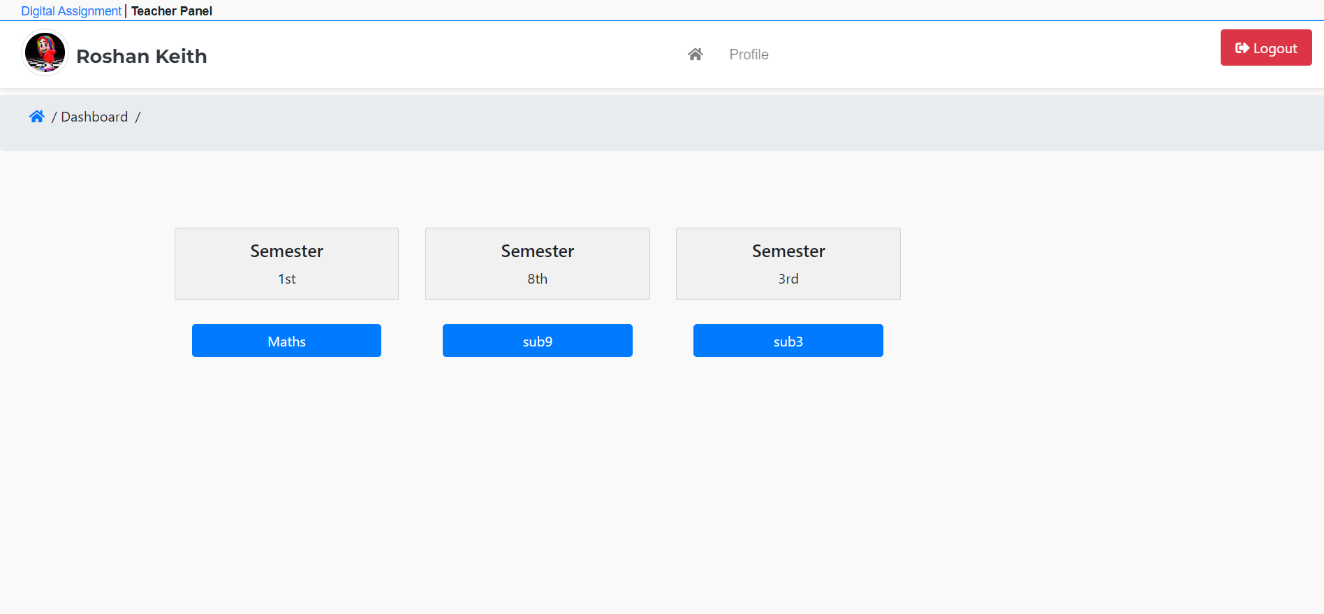
1. **Add Student**



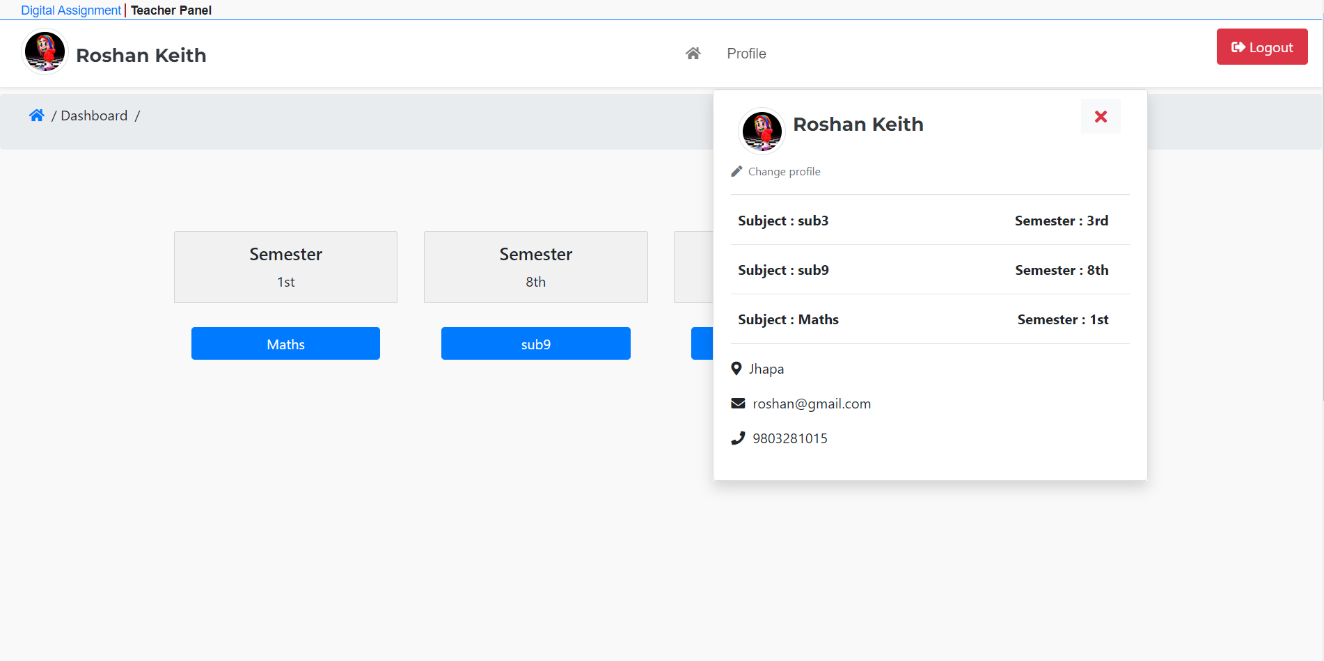
1. **Teacher Login Page**



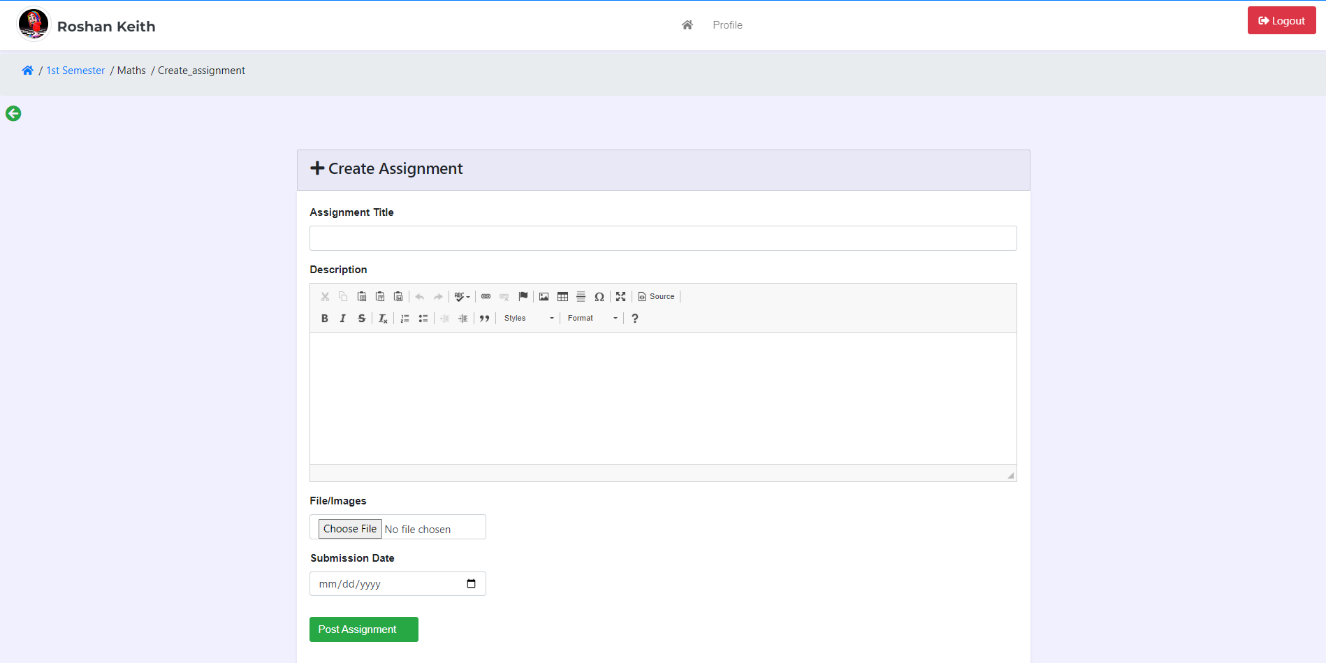
1. **Teacher Dashboard**



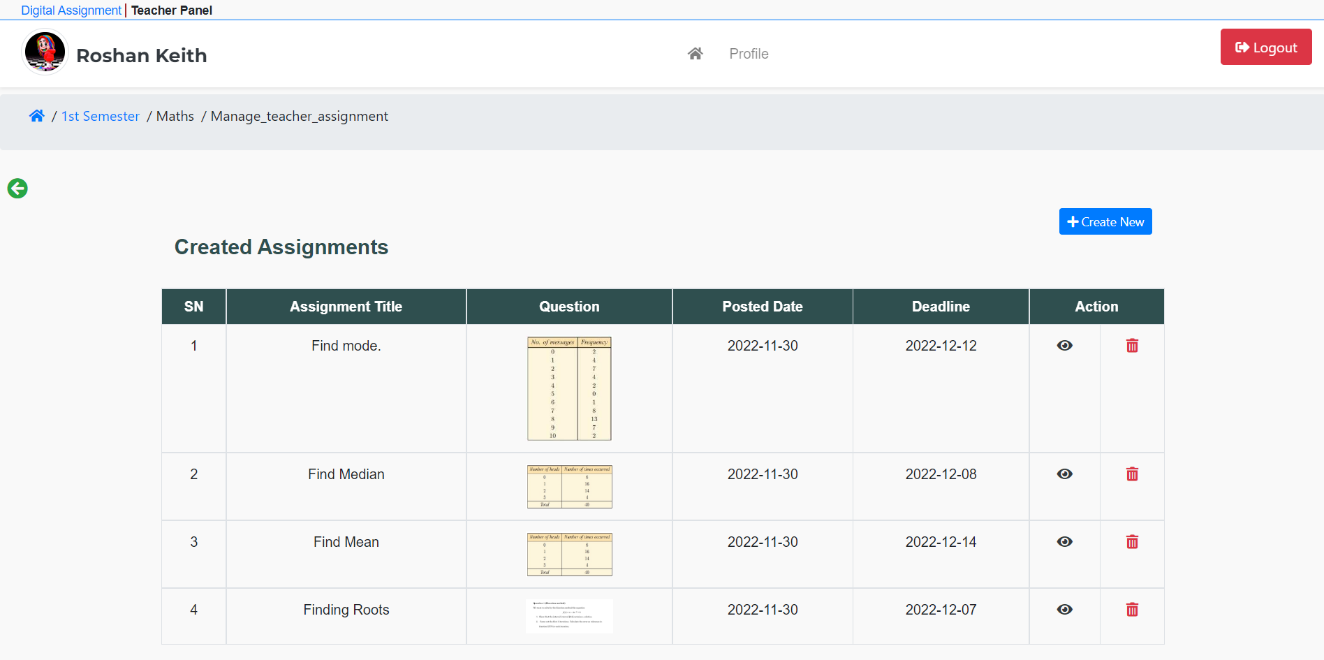
1. **Teacher Profile**



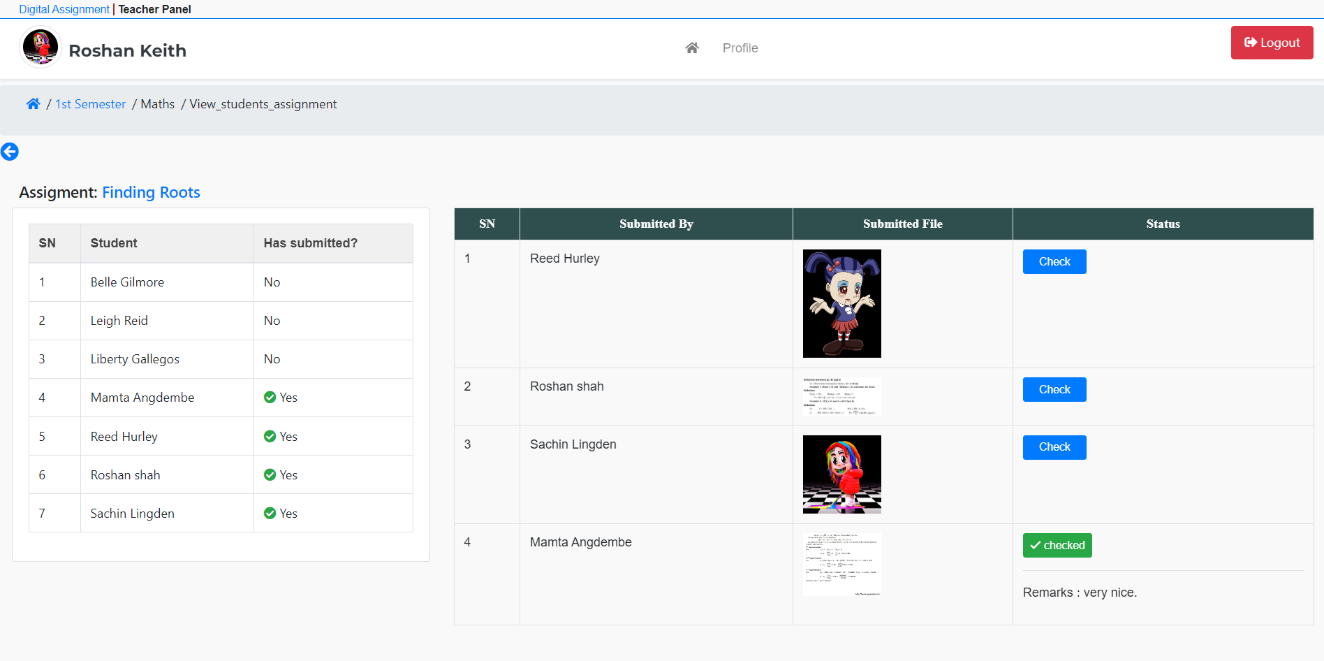
1. **Create Assignment**



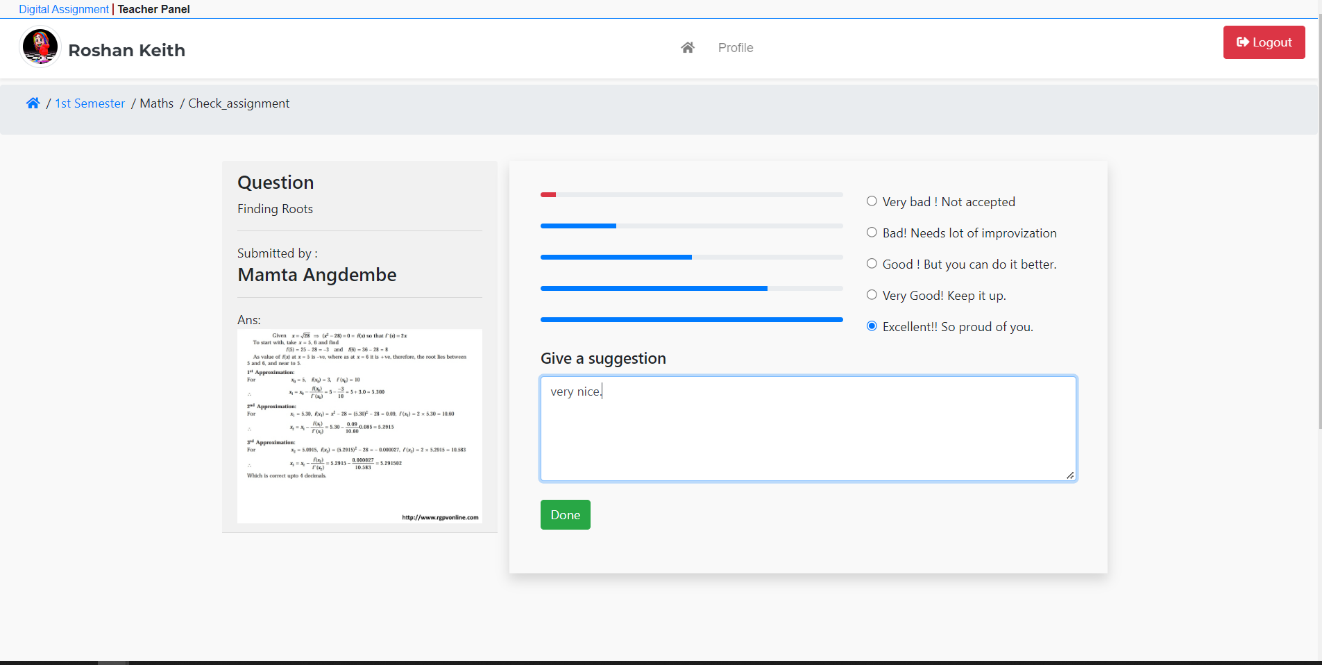
1. **Created Assignment**



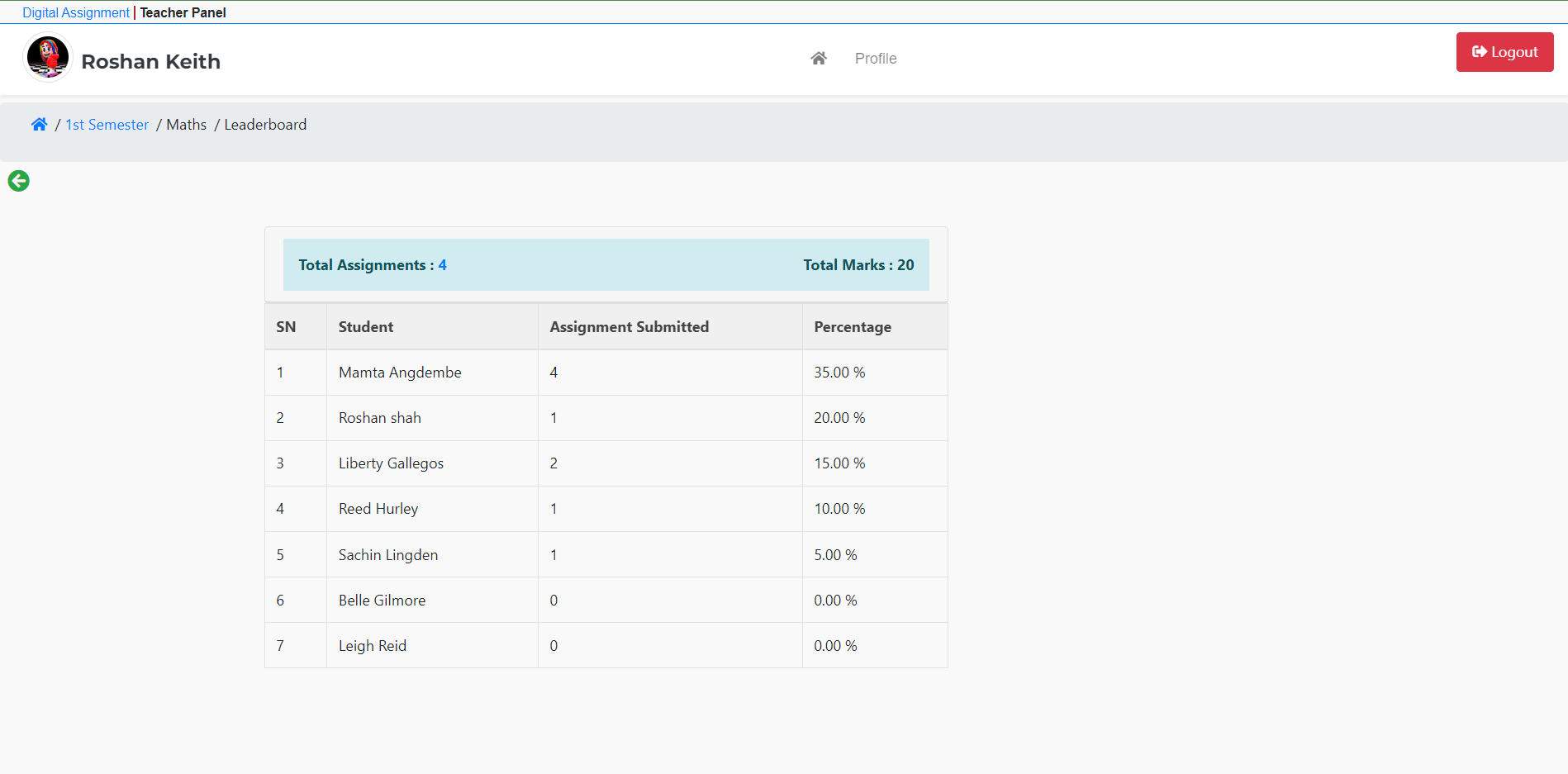
1. **View Student Assignment**



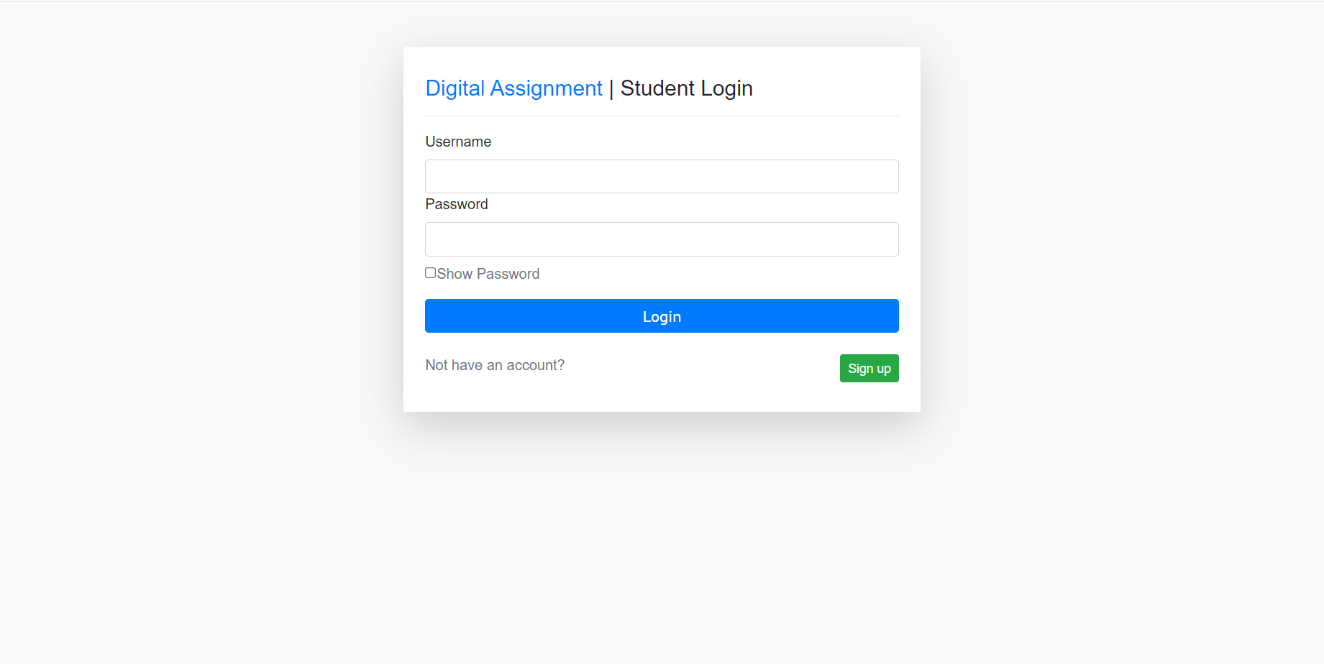
1. **Check Assignment**



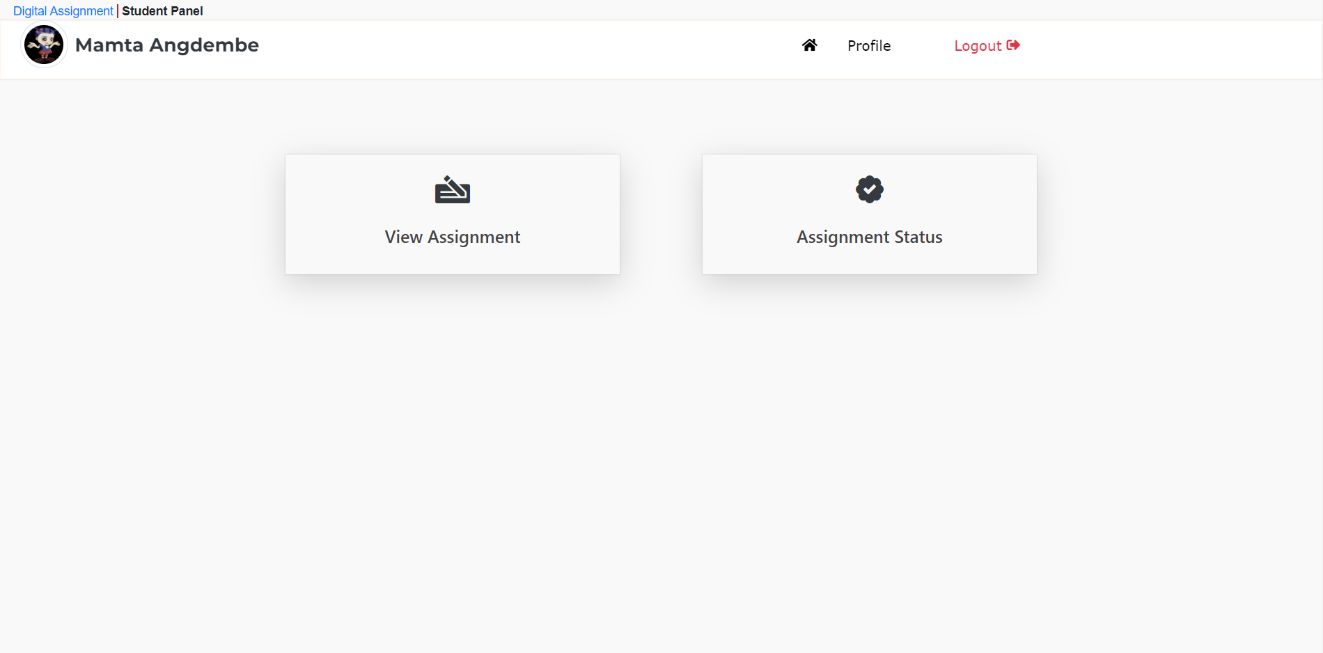
1. **View Assignment Record**



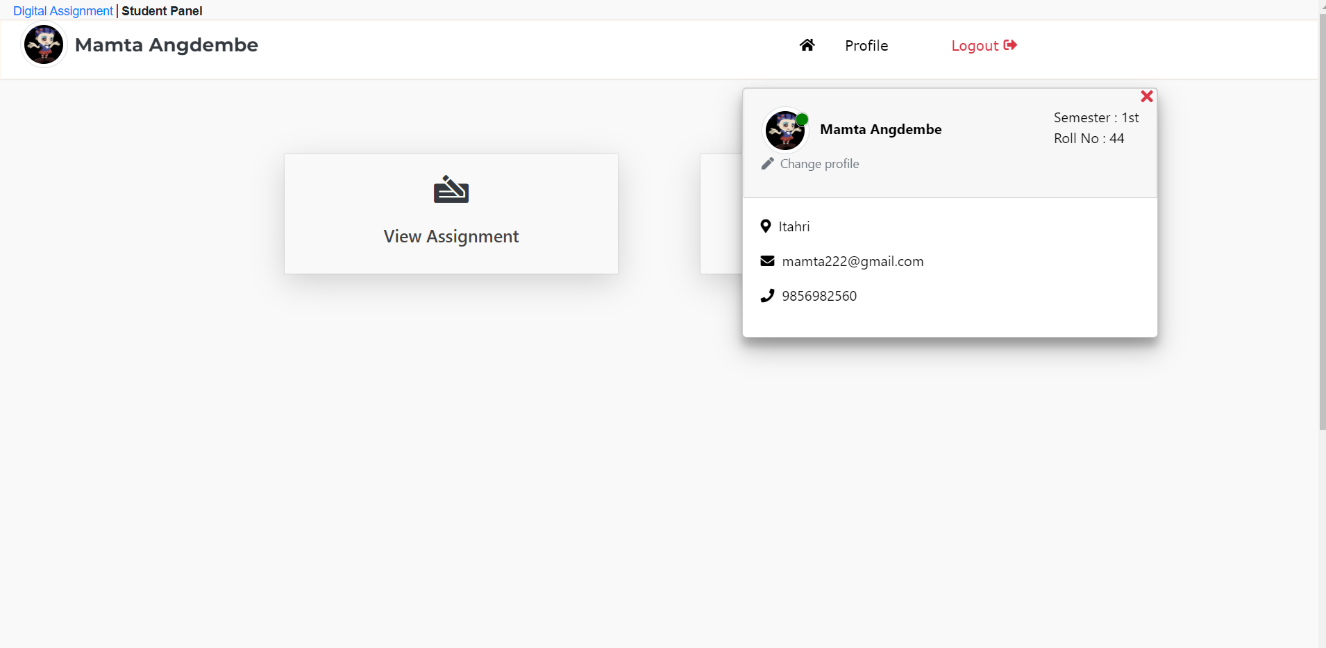
1. **Student Login Page**



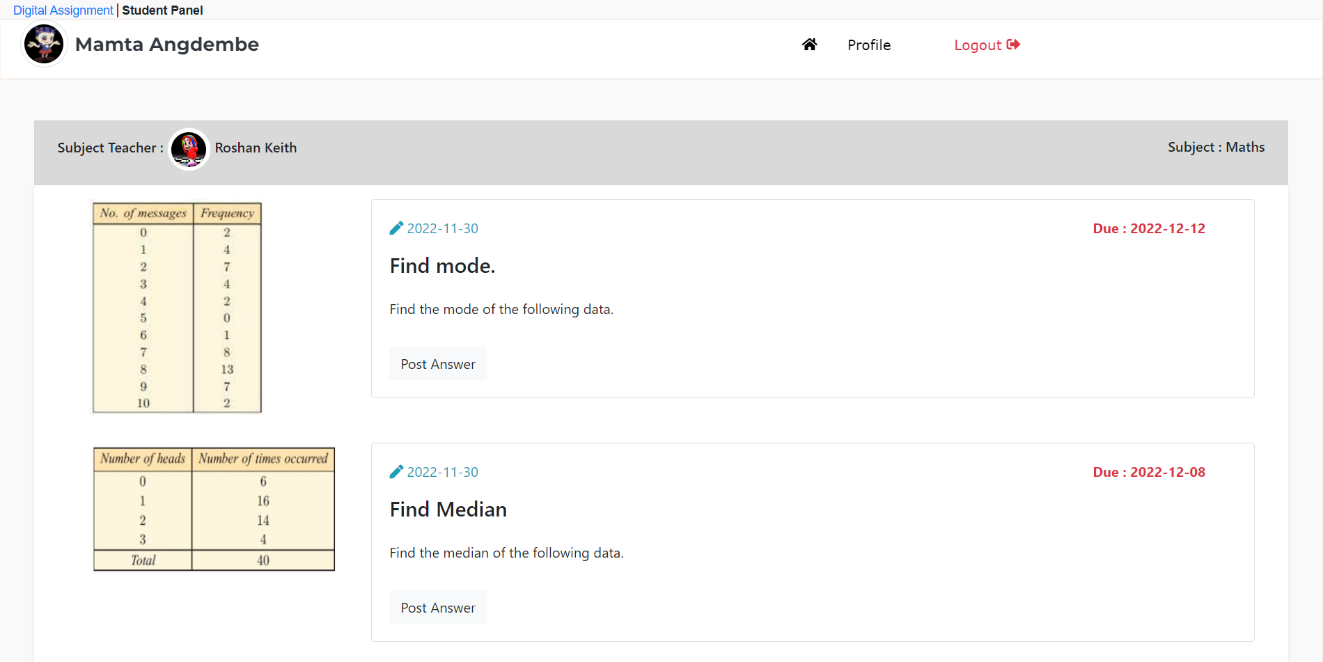
1. **Student Dashboard**



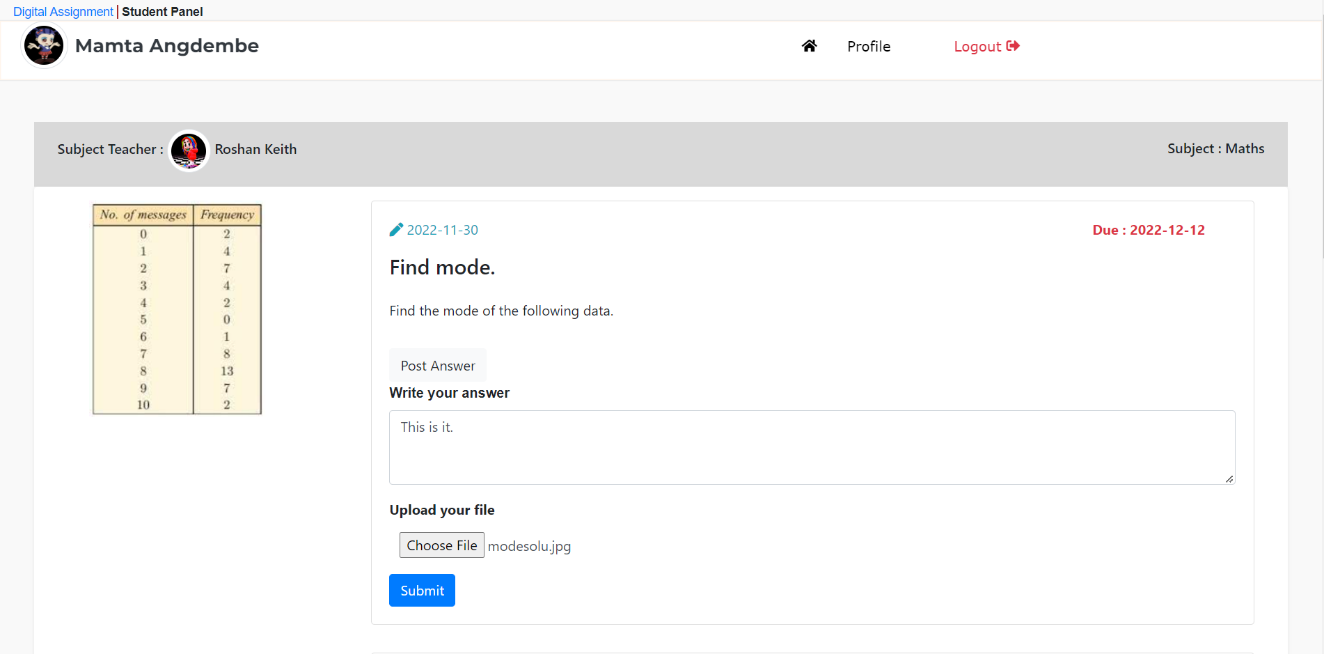
1. **Student Profile**



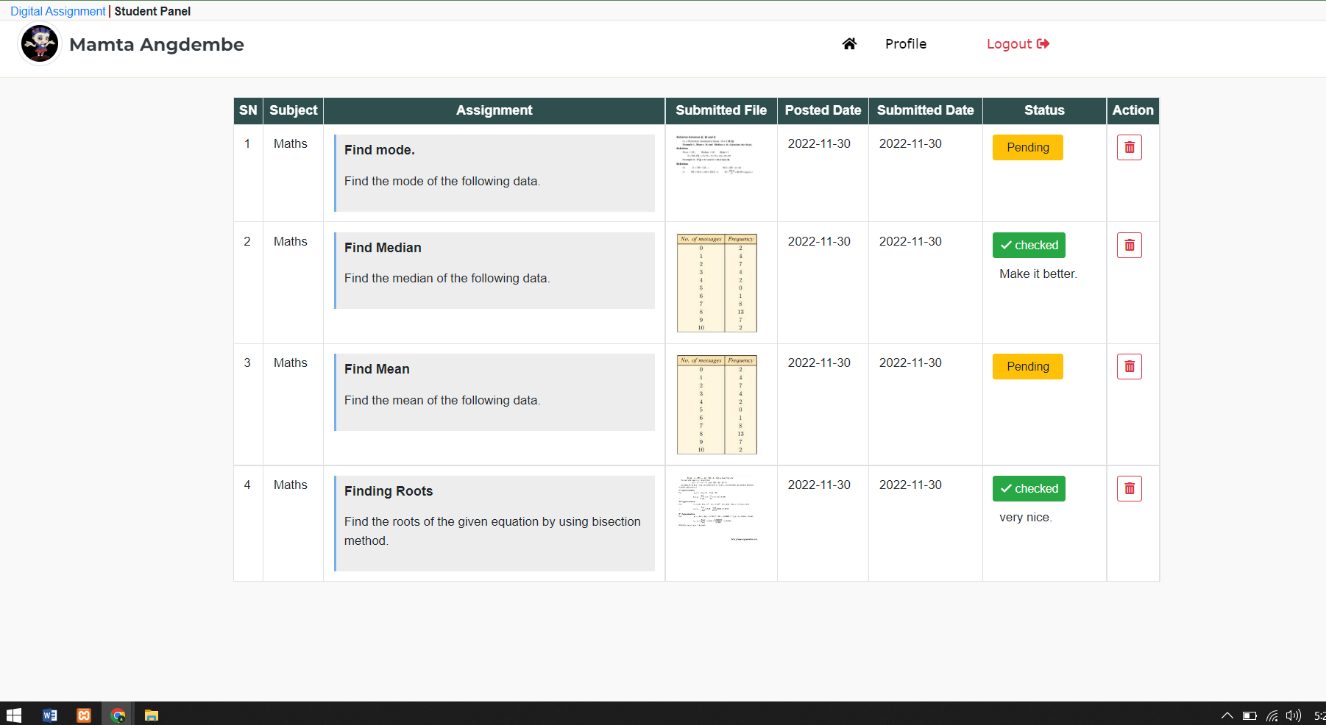
1. **View Assigned Assignment**



1. **Submit Assignment**



1. **Assignment Status**



## Appendix B: Source Code

**Admin Login Page**

<?php

require\_once ("config/config.php");

require\_once ("config/db.php");

require\_once root('layouts/header.php')

if(!empty($\_POST) && $\_POST['submit']=='submit'){

$username=$\_POST['username'];

// $password=md5($\_POST['password']);

$password=$\_POST['password'];

$user\_select=$obj->Query("SELECT \* FROM tbl\_admin WHERE email='$username' and password='$password'");

if($user\_select){

$user\_select= $user\_select[0];

session\_start();

$\_SESSION['admin-status']="loggedin";

$\_SESSION['mainuser']=$user\_select->username;

$\_SESSION['admin-login']='true';

echo "<script>window.location.href='".base\_url()."'</script>";

}else {

$\_SESSION['error'] = "Invalid username or password!";

}

}

$a = "Digital Assignment";

?>

<div style="height:10vh"></div>

<div class="container mt-2 bg-snow rounded" style="margin-top: 40px!important;font-family: roboto, sans-serif!important;">

<div class="row justify-content-center">

<div class="col-md-5 shadow-lg p-4 bg-white">

<h4 class="pt-2"> <a href="<?= exit\_url(); ?>" class="text-info"> <?= $a ?> </a> &#124; Admin Login </h4>

<hr>

<?php if (isset($\_SESSION['error'])) { ?>

<div class="alert alert-primary my-2">

<?php echo $\_SESSION['error'];unset($\_SESSION['error']); ?>

</div>

<?php } ?>

<form class="form-group" method="post">

<label>Username</label>

<input type="text" name="username" class="form-control" required>

<label>Password</label>

<input type="password" name="password" class="form-control" required id="Visible">

<button class="btn btn-info btn-block mt-4" type="submit" name="submit" value="submit">Login </button><br>

</form>

</div>

</div>

</div>

<div class="container d-none">

<div class="row justify-content-center">

<div class="col-md-6 shadow pb-4 pl-4 pr-4 pt-4">

<h3> Digital Assignment || Admin Login</h3><br>

<form method="post">

<div class="form-group">

<label>Username</label>

<input type="text" name="username" class="form-control"required>

</div>

<div class="form-group">

<label>Password</label>

<input type="password" name="password" class="form-control" required>

</div>

<button name="submit" value="submit" class="btn btn-sm btn-info mt-1">LOGIN

</button>

<!-- <span><a href="forget\_password.php">Forgot Password</span><br>

<br> -->

<br><br>

</form>

</div>

</div>

</div>

**Create Assignment**

<?php

$t\_sem = $\_GET['sem'];

$t\_sub = $\_GET['sub'];

$t\_sem\_str = $obj->select('semesters', '\*', 'name', array($t\_sem));

$t\_sem\_id = $t\_sem\_str[0]['id'];

if (isset($\_POST['submit'])) {

if ($\_POST['submit'] == 'submit') {

$old = $\_POST;

$check = $obj->select('tbl\_create\_assignment', '\*', 'title', array($\_POST['title']));

if ($check) {

$\_SESSION['titleError'] = "This assignment already exists!";

} else {

$imgName = $\_FILES['image']['name'];

$tmp\_name = $\_FILES['image']['tmp\_name'];

$location = 'create\_assignment' . '/' . $imgName;

move\_uploaded\_file($tmp\_name, $location); //upload file

unset($\_POST['submit']);

$\_POST['file'] = $imgName;

$\_POST['semester'] = $t\_sem\_id;

$\_POST['subject'] = $t\_sub;

$target = "manage\_teacher\_assignment.php?sem=$t\_sem&sub=$t\_sub";

$obj->Insert("tbl\_create\_assignment", $\_POST); //insert query

echo '<script>alert("Assignment created successfully")</script>';

echo "<script> window.location.href='" . base\_url($target) . "'</script>";

}

}

}

if (isset($\_SESSION['teacher\_id'])) {

$teacher\_sub = $obj->select('tbl\_teacher', '\*', 'tid', array($\_SESSION['teacher\_id']));

if ($teacher\_sub) {

$subject = $teacher\_sub[0]['tsubject'];

// echo $subject;

// print\_r($teacher\_sub);

}

}

include('teacherheader.php');

?>

<style>

body {

background-color: #f0f0ff;

}

</style>

<a href="activity.php?sem=<?= $t\_sem ?>&sub=<?= $t\_sub; ?>"><i class="fas fa-arrow-circle-left p-2 text-success" style="font-size: 1.5em;"></i></a>

<div class="container body-container rounded"><br>

<div class="row justify-content-center">

<div class="col-sm-12 mb-5">

<div class="card card-header">

<h4> <i class="fas fa-plus"></i>

Create Assignment</h4>

</div>

<div class="card-body bg-white shadow-sm">

<form action="" method="post" id="form" class="form-group" enctype="multipart/form-data">

<div class="error">

<!--For showing alert message -------------------------->

<?php if (isset($\_SESSION['create'])) { ?>

<div class="alert alert-success">

<?php echo $\_SESSION['create'];

unset($\_SESSION['create']); ?>

</div>

<?php } ?>

<?php if (isset($\_SESSION['error'])) { ?>

<div class="alert alert-danger">

<?php echo $\_SESSION['error'];

unset($\_SESSION['error']); ?>

</div>

<?php } ?>

<!------------------------End----------------------------->

</div>

<?php

$connection = mysqli\_connect("localhost", "root", "", "digital\_assignment");

// Check connection

if (mysqli\_connect\_errno()) {

echo "Failed to connect to MySQL: " . mysqli\_connect\_error();

}

?>

<style>

.navbar-home {

display: none;

}

button {

font-family: poppins, sans-serif;

}

</style>

<div class="form-group" style="display:none;">

<label style="font-family: nunito, sans-serif;font-weight:600">

Subject

</label>

<?php

$result = mysqli\_query($connection, "SELECT subjectname FROM addsubject where sub\_id = $subject order by subjectname");

$result = mysqli\_fetch\_array($result);

// while($row = mysqli\_fetch\_array($result))

// echo "<option selected disabled value='" . $row['subjectname'] . "'>" . $row['subjectname'] . "</option>";

?>

<input type="text" name="subject" class="form-control" value="<?php if (isset($result['subjectname'])) echo $result['subjectname']; ?>" readonly>

</div>

<div class="" id="errorSub" style="color:red"></div>

<div class="form-group">

<label style="font-family: nunito, sans-serif;font-weight:600">Assignment Title</label>

<input type="text" name="title" class="form-control" id="assignment\_title" required value="<?php if (isset($old)) {

echo $old['title'];

} ?>">

<a style="color: red;">

<?php if (isset($\_SESSION['titleError'])) {

echo $\_SESSION['titleError'];

unset($\_SESSION['titleError']);

} ?></a>

</div>

<div class="form-group">

<label style="font-family: nunito, sans-serif;font-weight:600"> Description</label><br>

<textarea rows="3" cols="50" name="description" id="assignment\_desc" required="required"></textarea>

<a style="color:red" id="descError"></a>

</div>

<div class="form-group w-25">

<label style="font-family: nunito, sans-serif;font-weight:600">File/Images</label>

<input type="file" name="image" class="form-control" id="filename" style="background-color: #fff;">

<a style="color:red" id="fileError"></a>

</div>

<div class="form-group w-25" hidden>

<label style="font-family: nunito, sans-serif;font-weight:600">Issued Date</label>

<input type="date" name="created\_date" readonly value="<?php echo date('Y-m-d') ?>" id="issueDate" class="form-control" required="required">

<a style="color:red" id="issueDateError"></a>

</div>

<div class="form-group w-25">

<label style="font-family: nunito, sans-serif;font-weight:600">Submission Date</label>

<input type="date" name="deadline" class="form-control" id="submissionDate" required="required">

<a style="color:red" id="subDateError"></a>

</div>

<input type="hidden" name="posted\_by" value="<?= $\_SESSION['posted\_by']; ?>">

<button class="btn btn-success my-3" name="submit" value="submit" onclick="return validate()"> Post Assignment <i class="fas fa-send text-success "></i></button>

</form>

</div>

</div>

</div>

</div>

<script>

function validate() {

// issue data validation

var isD = document.getElementById('issueDate').value;

if (isD == '' | null) {

document.getElementById('issueDateError').innerHTML = 'Issue date is required!';

return false;

} else {

document.getElementById('issueDateError').innerHTML = '';

}

// submission data validation

var sD = document.getElementById('submissionDate').value;

if (sD == '' | null) {

document.getElementById('subDateError').innerHTML = 'Submission date is required!';

return false;

} else {

document.getElementById('subDateError').innerHTML = '';

}

if (isD >= sD) {

document.getElementById('subDateError').innerHTML = 'Submission date must be greater than issued date!';

return false;

} else {

document.getElementById('subDateError').innerHTML = '';

}

}

</script>

<script src="<?= base\_url('lib/ckeditor/ckeditor.js') ?>"></script>

<script>

CKEDITOR.replace('description');

</script>

**Submit Assignment**

<?php

// print\_r($\_SESSION);

if (!isset($\_SESSION['isStudent']) && $\_SESSION['isStudent'] != 'true') {

header('location:student\_login.php');

}

if (isset($\_POST['submit'])) {

if ($\_POST['submit'] == 'submit') {

$id = $\_POST['assignment'];

$deadline = $obj->select('tbl\_create\_assignment', 'deadline', 'id', array($id));

$deadline = $deadline[0]['deadline'];

if ($\_POST['submitted\_date'] > $deadline) {

$\_SESSION['error'] = "You've crossed deadline. Don't exceed the deadline from next time";

header('location:submit\_assignment.php');

exit();

}

$assi\_id = $\_POST['assignment'];

$assi\_id = $\_POST['assignment'];

$studentt = $\_SESSION['submitted\_by'];

$check = $obj->Query("SELECT \* FROM tbl\_submit\_assignment WHERE assignment = '$assi\_id' and submitted\_by = '$studentt'");

if ($check) {

$\_SESSION['assiError'] = "You've already submitted this assignment!";

} else {

$imgName = $\_FILES['image']['name'];

$tmp\_name = $\_FILES['image']['tmp\_name'];

$location = 'submit\_assignment' . '/' . $imgName;

move\_uploaded\_file($tmp\_name, $location); //upload file

array\_pop($\_POST); //popping submit form post

$\_POST['file'] = $imgName;

if ($obj->Insert("tbl\_submit\_assignment", $\_POST)) {

$\_SESSION['create'] = "Assignment submitted successfully!";

} else {

echo "fail";

echo "<script> window.location.href='" . base\_url('view\_assignment\_status.php') . "'</script>";

}

}

}

}

include('studentheader.php');

?>

<style>

label {

font-size: 17px;

font-family: roboto, sans-serif;

}

</style>

<div class="container" style="min-height:80vh;"><br>

<div class="row justify-content-start">

<div class="col-md-8 shadow p-0">

<div class="card shadow">

<?php if (isset($\_SESSION['create'])) { ?>

<div class="alert alert-success">

<?php echo $\_SESSION['create'];

unset($\_SESSION['create']); ?>

</div>

<?php } ?>

<?php if (isset($\_SESSION['error'])) { ?>

<div class="alert alert-danger">

<?php echo $\_SESSION['error'];

unset($\_SESSION['error']); ?>

</div>

<?php } ?>

<div class="card-header text-left pb-0 mb-0">

<h4 class=""> Submit Assignment</h4><br>

</div>

</div>

<div class="card-body">

<form action="" method="post" id="form" class="form-group" enctype="multipart/form-data">

<div class="form-group">

<?php

if (isset($\_GET['id'])) {

$sub = $obj->select('tbl\_create\_assignment', '\*', 'id', array($\_GET['id']));

}

// print\_r($sub);

?>

<label class="font-weight-bold"><span>Assignment Title : </span></label>

<span><?php if (isset($sub[0]['id'])) echo $sub[0]['title']; ?></span>

<input type="hidden" name="assignment" value="<?php if (isset($old)) {

echo $old['assignment'];

} ?>">

<?php if (isset($\_SESSION['assiError'])) { ?>

<a class="text-danger my-2 d-block">

<?php

echo $\_SESSION['assiError'];

unset($\_SESSION['assiError']); ?>

</a>

<?php } ?>

</div>

<div class="form-group">

<label class="font-weight-bold"><span>Description :</span></label>

<span><?php if (isset($sub[0]['id'])) echo $sub[0]['description']; ?></span>

</div>

<br><br>

<div class="form-group">

<?php

if (isset($\_GET['id'])) {

$sub = $obj->select('tbl\_create\_assignment', '\*', 'id', array($\_GET['id']));

}

// print\_r($sub);

?>

<input type="hidden" name="assignment" readonly class="form-control" value="<?php if (isset($sub[0]['id'])) echo $sub[0]['id']; ?>">

<div class="form-group">

<input type="hidden" name="submitted\_date" value="<?php echo date('Y-m-d') ?>" id="issueDate" class="form-control" readonly>

<a style="color:red" id="issueDateError"></a>

</div>

<div class="form-group" style="margin-top:-3rem">

<label class="font-weight-bold">Upload your file</label>

<input type="file" name="image" class="form-control" id="filename">

<a style="color:red" id="fileError"></a>

</div>

<!--For showing alert message -------------------------->

<?php if (isset($\_SESSION['create'])) { ?>

<div class="alert alert-success">

<?php echo $\_SESSION['create'];

unset($\_SESSION['create']); ?>

</div>

<?php } ?>

<?php if (isset($\_SESSION['error'])) { ?>

<div class="alert alert-danger">

<?php echo $\_SESSION['error'];

unset($\_SESSION['error']); ?>

</div>

<?php } ?>

<!------------------------End----------------------------->

<style>

button {

font-family: poppins, sans-serif;

}

</style>

<input type="hidden" name="submitted\_by" value="<?php if (isset($\_SESSION['submitted\_by'])) {

echo $\_SESSION['submitted\_by'];

}

?>">

<button class="btn btn-success" name="submit" value="submit" onclick="return validate()"> Submit </button>

</form>

</div>

</div>

</div>

</div>

<div style="height:14vh">

</div>

<script>

function validate() {

// file validation

var val = document.getElementById('filename').value;

if (val == '' | null) {

document.getElementById('fileError').innerHTML = 'File must be attached!';

return false;

} else {

document.getElementById('fileError').innerHTML = '';

}

// submission data validation

var sD = document.getElementById('submissionDate').value;

if (sD == '' | null) {

document.getElementById('subDateError').innerHTML = 'Submission date is required!';

return false;

} else {

document.getElementById('subDateError').innerHTML = '';

}

if (isD >= sD) {

document.getElementById('subDateError').innerHTML = 'Submission date must be greater than issue date!';

return false;

} else {

document.getElementById('subDateError').innerHTML = '';

}

}

$(document).ready(function() {

$('#subject').on('change', function() {

var aid = $(this).val();

var data = {

aid: aid

}

$.ajax({

type: "POST",

url: 'get\_deadline.php',

data: data,

success: function(e) {

$('#deadline').html(e);

}

})

})

})

$(document).ready(function() {

$('#subject').on('change', function() {

var aid = $(this).val();

var data = {

aid: aid

}

$.ajax({

type: "POST",

url: 'get\_issued\_date.php',

data: data,

success: function(e) {

$('#date\_of\_issued').html(e);

}

})

})

})

</script>

**Algorithm for Quick Sort**

<?php

$t\_sem = $\_GET['sem'];

$t\_sub = $\_GET['sub'];

if (!isset($\_SESSION['isTeacher']) && $\_SESSION['isTeacher'] != 'true') {

header('location:teacher\_login.php');

}

$stusem = $\_GET['sem'];

$listStu = $obj->Query("SELECT \* from tbl\_student where semester = '$stusem' order by sname asc ");

?>

<?php include('teacherheader.php'); ?>

<?php

$nameArray = array();

$marksArray = array();

$countAssignmentArray = array();

foreach ($listStu as $key => $value) :

array\_push($countAssignmentArray, $obj->Query("SELECT count(id) as howManyAssi from tbl\_submit\_assignment where submitted\_by = '$value->sname'")[0]->howManyAssi);

array\_push($marksArray, intval($obj->Query("SELECT sum(grade) as marks\_gain from tbl\_submit\_assignment where submitted\_by = '$value->sname'")[0]->marks\_gain));

array\_push($nameArray, $value->sname);

endforeach;

$leaderBoard = array($countAssignmentArray, $marksArray, $nameArray);

?>

<!-- <pre><?php print\_r($leaderBoard) ?></pre> -->

<?php

//Swap Function

function swap(&$array0, &$array1, &$array2, $left, $right)

{

$temp\_marksArray = $array0[$right];

$temp\_nameArray = $array1[$right];

$temp\_countAssignmentArray = $array2[$right];

$array0[$right] = $array0[$left];

$array1[$right] = $array1[$left];

$array2[$right] = $array2[$left];

$array0[$left] = $temp\_marksArray;

$array1[$left] = $temp\_nameArray;

$array2[$left] = $temp\_countAssignmentArray;

}

//Quick Sort Function

function quicksort(&$array0, &$array1, &$array2, $left, $right)

{

if ($left < $right) {

$boundary = $left;

for ($i = $left + 1; $i < $right; $i++) {

if ($array0[$i] > $array0[$left]) {

swap($array0, $array1, $array2, $i, ++$boundary);

}

}

swap($array0, $array1, $array2, $left, $boundary);

quicksort($array0, $array1, $array2, $left, $boundary);

quicksort($array0, $array1, $array2, $boundary + 1, $right);

}

}

$array0 = $marksArray;

$array1 = $nameArray;

$array2 = $countAssignmentArray;

quicksort($array0, $array1, $array2, 0, count($array0));

$leaderBoard = array("countAssignment" => $array2, "totalMarks" => $array0, "nameArray" => $array1);

?>

<!-- <pre><?php print\_r($leaderBoard) ?></pre> -->

<a href="activity.php?sem=<?= $t\_sem ?>&sub=<?=$t\_sub;?>"><i class="fas fa-arrow-circle-left mx-2 text-success" style="font-size: 1.5em;"></i></a>

<div class="container mt-4">

<div class="row">

<div class="col-8">

<div class="card">

<div class="card-header">

<div class="alert-info p-3">

<?php

$subjectt = $\_GET['sub'];

$tot\_marks\_subwise = $obj->Query("SELECT count(id) as count\_tot\_assi from tbl\_create\_assignment where subject = '$subjectt'");

$total\_assi = $tot\_marks\_subwise[0]->count\_tot\_assi;

$total\_marks = $total\_assi \* 5;

?>

<div class="row font-weight-bold">

<div class="col-6 ">Total Assignments : <a href="manage\_teacher\_assignment.php?sem=<?= $t\_sem ?>&sub=<?= $t\_sub; ?>&view\_assignment"><?= $total\_assi ?>

</div></a>

<div class="col-6 d-flex justify-content-end">

Total Marks : <?= $total\_marks ?>

</div>

</div>

</div>

</div>

</div>

<?php

if ($total\_marks == 0) { ?>

<h5 class="text-danger p-3">No results for this subject .</h5>

<?php } else { ?>

<table class="table stuTbl table-bordered table-responsive-lite table-hover">

<tr style="position: sticky; top:0;background:#f0f0f0;color:#444 !important;">

<th>SN</th>

<th>Student</th>

<th>Assignment Submitted</th>

<th>Percentage</th>

</tr>

<?php

for ($index = 0; $index < count($leaderBoard['totalMarks']); $index++) { ?>

<tr>

<td><?= $index + 1 ?></td>

<td><?= $leaderBoard['nameArray'][$index] ?></td>

<td>

<?= $leaderBoard['countAssignment'][$index] ?>

</td>

<td>

<?= number\_format((float)round(($leaderBoard['totalMarks'][$index] / $total\_marks) \* 100, 2), 2, '.', ',') ?> %

</td>

</tr>

<?php } ?>

</table>

<?php } ?>

</div>

</div>

<div style="height:40vh">

</div>