

E-PRACTICAL SUBMISSIONS

(A solution for handwritten experiments)

UCS503 Software Engineering Project Report

End-Semester Evaluation

Submitted by:

Krishna Mundra (101803056)

Rhythm Singla (101803073)

Karan Madan (101853003)

BE Third Year, C0E

Submitted to:

Ms. Sawinder Kaur



Computer Science and Engineering Department

TIET, Patiala

November 2020

TABLE OF CONTENTS

Contents	Page No
1. Project overview (1 page)	3
2. Software Requirement Specification (SRS)	4
3. Structured Analysis	15
3.1 Data Flow Diagrams	15
3.1.1 DFD Level 0	15
3.1.2 DFD Level 1	16
3.1.3 DFD Level 2	17
3.2 ER Diagram	17
4. Object Oriented Analysis	18
4.1 Use Case Diagram	18
4.2 Use Case Templates	19
4.3 Activity Diagram	20
4.4 Collaboration Diagram	21
4.5 Sequence Diagram	22
4.6 State Chart Diagram	24
4.7 Component Diagram	24
4.8 Deployment Diagram	25
5. Testing	26
5.1 Test Plan	26
5.2 Test Case Report	28
5.3 Screenshots	34

1. Project overview

Project E- Practical Submission reduces the writing work of practical files and focuses more on performing practicals. Students can work in a stress-free environment. And the teachers don't have to maintain long records of various students.

This project aims to reduce large handwritten work in the "Lab Experiment Files", thus increasing the much-needed scope for practical knowledge.

E-Practical submission helps students pay more attention to practicals in their physical classes, thus reducing their burden of writing long files and physical submissions.

This also helps the teacher - the teacher doesn't have to maintain long records of numerous classes with many students. He/she shall get all the information on the portal where and when they can check the files and mark it.

So, there will be a thorough record of student's performance in a single place, which will remain for the whole semester without bearing any loss issues.

This product targets:

- Students (appearing in respective courses)
- Teachers (setting Course intended data)
- Documentation writers (involved in updating of course content)
- Developers (involved in updating of product features)

This software system will be an online practical management system for any university or school wishing to manage the submission and evaluation of practical experiments online. This software aims to cover students, teachers, and administrators within its scope. The primary roles of the benefactors/actors of this online practical submission system are:

Student - He/she will experiment in the respective labs and enter the experiment's readings in the portal after selecting the appropriate subject code, teacher, and experiment. The experiment file shall generate automatically with intended graphical and tabular data, which finally is sent to the teacher's end.

Teacher - He/she shall evaluate the experiment by selecting the appropriate batch details with the experiment name and submit the marked file back at the portal for student verifications.

Administrator - He/she shall update any information regarding the experiment, be it adding or removing the experiment as asked by the teacher.

2. Software Requirement Specification (SRS)

Software Requirements Specification

for

E-Practical Submissions

Version 1.0 approved

Prepared by: Karan Madan

Krishna Mundra

Rhythm Singla

Group 3

Submitted to: Ms. Sawinder Kaur

Thapar Institute of Engineering and Technology, Patiala

28 September 2020

Table of Contents

Table of Contents	ii
Revision History	ii
1. Introduction	1
1.1 Purpose	1
1.2 Document Conventions	1
1.3 Intended Audience and Reading Suggestions	1
1.4 Project Scope	1
1.5 References	2
2. Overall Description	2
2.1 Product Perspective	3
2.2 Product Features	3
2.3 User Classes and Characteristics	3
2.4 Operating Environment	3
2.5 Design and Implementation Constraints	3
2.6 User Documentation	3
2.7 Assumptions and Dependencies	4
3. System Features	4
3.1 System Feature 1	5
3.2 System Feature 2	6
3.3 System Feature 3	7
3.4 System Feature 4	8
3.5 System Feature 5	8
3.6 System Feature 6	9
4. External Interface Requirements	10
4.1 User Interfaces	10
4.2 Hardware Interfaces	11
4.3 Software Interfaces	11
4.4 Communications Interfaces	11
5. Other Nonfunctional Requirements	12
5.1 Performance Requirements	12
5.2 Safety Requirements	13
5.3 Security Requirements	13
5.4 Software Quality Attributes	14
6. Other Requirements	14
Appendix A: Glossary	14
Appendix B: Analysis Models	14
Appendix C: Issues List	14

Revision History

Name	Date	Reason for Changes	Version

1. Introduction

Project E- Practical Submission reduces the writing work of practical files and focuses more on performing practicals. Students can work in a stress-free environment. And the teachers don't have to maintain long records of various students.

1.1 Purpose

This project aims to reduce large handwritten work in the "Lab Experiment Files", thus increasing the much-needed scope for practical knowledge.

E-Practical submission helps students pay more attention to practicals in their physical classes, thus reducing their burden of writing long files and physical submissions.

This also helps the teacher - the teacher doesn't have to maintain long records of numerous classes with many students. He/she shall get all the information on the portal where and when they can check the files and mark it. So, there will be a thorough record of student's performance in a single place, which will remain for the whole semester without bearing any loss issues.

1.2 Document Conventions

When writing this SRS for E-Practical Submissions, the following terminologies are used:

Conventionally the complete document is to be dedicated only for those subjects involving lab experiments. We have standardized the priority levels for maintaining different sets of databases as follows:

Login data of students > Login data of Teachers and administrator > database of files corresponding to every student.

1.3 Intended Audience and Reading Suggestions

This product targets:

- Students (appearing in respective courses)
- Teachers (setting Course intended data)
- Documentation writers (involved in updating of course content)
- Developers (involved in updating of product features)

This SRS mainly breaks down E-Practical Submission into five essential sections rather questions:

1. How does the product work?
2. How will it be helpful?
3. Who all can use it?
4. What is the purpose behind it?
5. How does it focus more on practical skills while reducing the writing work to zero?

1.4 Project Scope

This software system will be an online practical management system for any university or school wishing to manage the submission and evaluation of practical experiments online. This software aims to cover students, teachers, and administrators within its scope. The primary roles of the benefactors/actors of this online practical submission system are:

Student - He/she will experiment in the respective labs and enter the experiment's readings in the portal after selecting the appropriate subject code, teacher, and experiment. The experiment file shall generate automatically with intended graphical and tabular data, which finally is sent to the teacher's end.

Teacher - He/she shall evaluate the experiment by selecting the appropriate batch details with the experiment name and submit the marked file back at the portal for student verifications.

Administrator - He/she shall update any information regarding the experiment, be it adding or removing the experiment as asked by the teacher.

- More precisely, to design and develop a simple and intuitive website that shall provide the above features to the educational institute's users.

1.5 References

- 1) IEEE SRS Template.pdf (Uploaded on Thapar Learning Management System - LMS)
https://ada-lms.thapar.edu/moodle/pluginfile.php/64897/mod_folder/content/0/IEEE%20SRS%20Template.pdf?forcedownload=1
- 2) IEEE SRS Case Study_Attendance Monitoring System.pdf (Uploaded on Thapar Learning Management System - LMS)
https://ada-lms.thapar.edu/moodle/pluginfile.php/64897/mod_folder/content/0/IEEE%20SRS%20Case%20Study_Attendance%20Monitoring%20System.pdf?forcedownload=1
- 3) SRS.ppt (Powerpoint presentation of Software Requirement Specifications)
<https://ada-lms.thapar.edu/moodle/mod/folder/view.php?id=3088>

2. Overall Description

The students, teachers, and admin can log in via email id provided by Thapar University, Patiala. The student can select the branch, year, and subject, and accordingly, a list of experiments will come. Students can select the experiment and all the information about that experiment will come. Then the student can enter the readings and generate graphs. The pdf can be generated, downloaded, and submitted to the teacher.

The teacher can select the branch, year and batch and a list of students will come. The teacher can see the files and evaluate the files.

The admin can update, add, and delete the experiments and the experiment details.

2.1 Product Perspectives

E-Practical Submission is a web-based system. The system uses the student's serial number to access the account. The system provides a secure environment for all file submissions and markings.

2.2 Product Features

The central feature of E Practical Submission is that it reduces the handwritten work of practicals, and hence the students can focus more on practical knowledge.

Three categories of people can log in it:

1. **Student** - The student can select the year, subject, and teacher. Then they can choose the experiment, which will display all the details like aim, apparatus, theory, procedure, precautions, diagrams. The students have to enter the readings of the experiment and can generate the graph. After that, the student can submit the file to the teacher.
2. **Teacher** - The teacher can select the year and batch, and a list of all students who have submitted the files will be displayed. They can check and mark the file there.
3. **Administrator** - The administrator will add, delete, and update the experiments in a subject.

2.3 User Classes and Characteristics

There are three types of users involved: students, teachers, and administrators. A student makes the e-practical file and submits it to the teacher. The teacher checks the file, marks it, and returns it to the student. Administrators can update, add, and delete the experiments. An administrator keeps a check on the experiments to be performed.

The only skill needed by all three user categories is to know how to use the website efficiently.

2.4 Operating Environment

The portal doesn't need a specific environment to run. It can work in all environments. The user interface is well designed so that Students, Teachers, and Administrators require no training to use it.

2.5 Design and Implementation Constraints

The system provides access to all users - students, teachers, and administrators without any training. All the information is in a secure environment. And the submission, checking, marking, and returning of files is made in a safe environment.

The student's data for file submission and marks will be maintained.

2.6 User Documentation

The following documentation will be provided along with the website:

1. User manual for students
2. User manual for teachers
3. User manuals for administrators

2.7 Assumptions and Dependencies

The following assumptions are used in the software:

1. Users have an active internet connection or have access to view the website or software.
2. The user runs an operating system that supports Internet Browsing.

3. Users should fill proper details for login. No false information is accepted.
4. Our website will not be violating any internet ethnic or cultural rules.
5. All the experiments will be timely updated by the administrator.
6. Privacy is maintained in the submission and checking of practical files.
7. No students can misuse the portal.

3. System Features

This section tells about the major features of E-Practical Submission software. Some of the system features include Register User, Login User, Experiment Selection, Graph Generation, File Generation, and Submission and evaluation of files by the teacher. This section contains a Use Case Diagram, Class Diagram, and Activity Diagram.

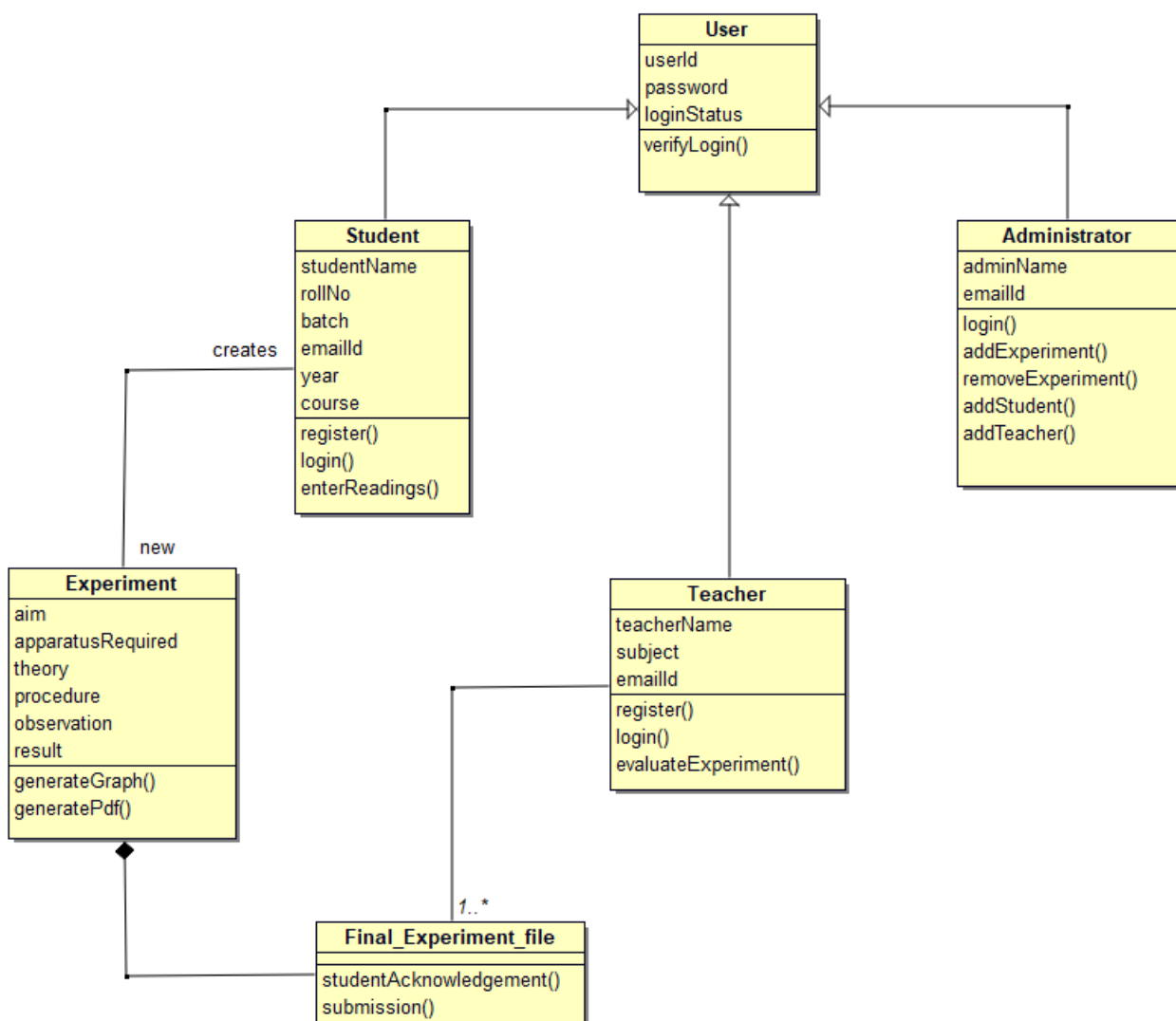


Figure 1: Class diagram

A class diagram in the Unified Modeling Language is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and the relationships among objects.

3.1 Register User (Student / Teacher)

Students and Teachers can register separately on the portal by providing necessary details.

3.1.1 Description and priority

The given feature will create an account of the user on our website using which their records will be stored on our database, and they can run all the intended features. For better security/privacy, a user can't access the website without an account. So, getting registered is a high priority, as all other functionalities depend on it.

3.1.2 Stimulus/Response Sequences

1. The user will click on the register/signup button.
2. A form will appear, which has to be filled by the user accordingly.
3. The system will check the validity of the credentials by comparing it with the current database.
4. The user account will be created if everything is valid.

3. Functional Requirements

1. Navigation to the registration page

1. **Input** - The user needs to click on the register/signup button
2. **Outputs and errors** - A form having some fields is displayed to the user.

1. User details

- a. **Input** - The user needs to input details in all the required fields of the form.
- b. **Outputs and errors** - The user will get a prompt of getting successfully registered if all the details entered by the user are valid/correct as checked by the system according to the present database entries. If any required field is missed, then the form will not get submitted, and the user will be prompted to do so. If there is an issue in a password or email is already registered, then an error message will be displayed depending on the condition, and the user will correct it.

3.2 User Login (Student / Teacher / Admin)

The students, teachers and admins will have a separate option to log in. They have to provide a valid email id and password to log in.

3.2.1 Description and priority

The given feature will help the user to login to their account. It enables the users to access the functionality of the website based on their roles in the system. For example, after logging in, the student can create a new experiment file and submit it. Similarly, after logging in, the teacher can access the submitted file and evaluate it. The admin, after logging in, can update any information or experiment. This feature is mandatory for other functionalities to work, so this is a high priority.

3.2.2 Stimulus/Response Sequences

1. The user(student / teacher / admin) will click on the login button.
2. A form will appear, and the user has to fill his/her credentials, i.e., email-id and password.
3. The system will check the validity of the credentials by comparing it with the current database.
4. The user will get logged in if everything is valid.

3.2.3 Functional Requirements

1. **Navigation to login page**
 - a. **Input** - The user needs to click on the login button.
 - b. **Outputs and errors** - The user has displayed a form having required fields asking email-id and password.
2. **User details**
 - a. **Input** - The user needs to input details in all the required fields of the form.
 - b. **Outputs and errors** - The user will get a prompt of getting successfully logged in if the credentials entered by the user are valid/correct as checked by the system according to the present database entries. If any required field is missed, then the form will not get submitted, and the user will be prompted to do so. If the password or email is incorrect, then an error message will be displayed, and the user will correct it, depending on the condition.

3.3 New experiment (User: Student)

The student can select an experiment which will be displayed according to the subject and year they have chosen.

3.3.1 Description and priority

The given feature will help the student to select an experiment from our database. The student shall choose a new experiment under a subject and the teacher corresponding to that subject. As the practical file gets generated, it can be submitted and evaluated by the required teacher. So, this is a high priority.

3.3.2 Stimulus/Response Sequences

1. The user(student) will click on the new experiment button.
2. A page will appear, in which the user has to select the appropriate subject code, teacher's name, and experiment name from the provided dropdown fields.

3. The system will check the validity of the selected options in the fields by comparing it with the current database.
4. If everything is valid, then a new experiment file gets displayed to the student user.

3. Functional Requirements

1. Navigation to the new experiment page

- a. **Input** - The user needs to click on the new experiment button.
- b. **Outputs and errors** - The user has displayed a form having required fields.

2. Experiment details

- a. **Input** - The user needs to input details in all the required fields of the form.
- b. **Outputs and errors** - The user will be displayed the required experiment that matches our database according to the user. If any required field is missed, then the form will not get submitted, and the user will be prompted to complete the form.

3.4 Observations and Graph

When the students enter readings of the experiment performed they will have an option to generate a graph(if any).

3.4.1 Description and priority

The given feature will help the student to record readings and observations of the experiment selected previously. The experiment file will get generated, submitted, and evaluated even if they enter wrong values in the observation or readings fields. So, this is a low priority.

3.4.2 Stimulus/Response Sequences

1. The user(student) will click inside the observations field.
2. Readings/observations have to be entered here by the user, according to the experiment selected previously.
3. The readings get recorded.
4. A graph (if present) gets generated according to the observations of the student.

3. Functional Requirements

1. Navigation to observations field

- a. **Input** - The user needs to click in the observations field and enter the readings/observations.
- b. **Outputs and errors** - The observations entered by the user get recorded.

2. Graphical representation

- a. **Input** - The user needs to click on the generate-graph button.
- b. **Outputs and errors** - The user will be displayed the required graphical representation according to the readings entered by the student user. If any required graph is missed by the user, then an error message will be displayed, and the user will be prompted.

3.5 File generation and submission

After completing the experiment file, the student can generate it and submit it to the teacher.

3.5.1 Description and priority

1. Generate a file

This is the feature that holds the utmost importance. From putting in the necessary details into the document to generating automatic graphs and calculations, the "Generate File" button gives the actual view of the file to the student user. The student is displayed each and every detail of the experiment. The file submission and file evaluation feature depend on the file generation feature. So, it is a high priority feature.

2. Submit file

This feature ensures that the experiment file generated earlier in pdf format is sent to the concerned teacher for marking. This feature allows the teacher to maintain a steady record of the student's performance, and evaluate it. Submitting the file is mandatory. So, this is a high priority feature.

3.5.2 Stimulus/Response Sequences

This is a simple step that the student completes after clicking on the "Generate File" button. Once the student clicks here, they get a document prepared in either word or pdf format.

The above step is simply followed by a "Send for evaluation/Submit" button clicking on which the student can submit the generated file to the teacher for evaluation.

3.5.3 Functional Requirements

1. Navigation to the final created file

a) **Input** - The user needs to click on the "generate the file" button.

b) **Outputs and errors** - The user has displayed a full document of the desired experiment. In case of any error/discrepancy, they can revert to previous windows for further changes.

2. Submission

a) **Input** - The user needs to click on the "Send file for evaluation" button.

b) **Outputs and errors** - The student receives a message that the file has been submitted successfully to the teacher for evaluation.

3.6 Evaluation (User: Teacher)

The practical experiment submitted by the student after the acknowledgment of the respective teacher is evaluated. The results/ grades of the experiment are then updated on the portal.

3.6.1 Description and priority

This feature gives the teacher a view of the student's experiment file including the observations and graphs generated.

The job of the teacher is to evaluate each student from a specific year, branch and batch. The teacher goes through the document and evaluates it by giving the necessary marks to the student. This feature, if missed, doesn't stop the functioning of any other feature of the system, but it is a necessary feature. So this is a feature of medium priority.

3.6.2 Stimulus/Response Sequences

The teacher selects the appropriate year, branch, and batch. Then, a student is picked up for evaluation from a student table. The document is marked accordingly. The marked document with necessary feedback is sent back to the student.

The evaluated experiment then adds up in the database visible to both students and teachers for future references.

3.6.3 Functional Requirements

The core requirement for this process to carry out consists of write access to the teacher.

1. Annotation

The teacher can annotate on the student file in order to mark any mistake/correction or any necessary feedback.

2. Updation

When the teacher sends an evaluated file to the student, it ends up in a database that is accessible to both teacher and student, say, for example, a Google drive link.

The teacher and student can communicate for any discrepancy or rechecking in the remarks section accessible to both teacher and student.

4. External Interface Requirements

External interface requirements are types of functional requirements. They're important as they outline how our product will interface with other components of the system as well as with different users of the system.

4.1 User Interfaces

Using this system is relatively intuitive and straightforward. The website consists of simple images and easily understandable texts. The user encounters a Home button (to return back to home page) on every page of the website. The navigation bar at the home page is available at all succeeding web pages of the system. A user familiar with basic browser navigation skills should be able to understand all functionality provided by the system.

4.1.1 Home Page Interface

This is the first interface that every user encounters whenever he/she tries to

access the E-Practical Submission System. It consists of basic information of the E-Practical Submission System, how it works, and login and signup buttons in order to access other functionalities.

4.1.2 Login Interface

This is the interface that will allow the user to log in to the system if he/she has an account already. After pressing the Login button, a user will be able to login to the system. This interface was created using HTML and CSS codes.

4.1.3 Registration Interface

This is another interface in which an unregistered user can register himself/herself. A guest user can click on the “Create account / Signup” link and fill the short registration form where he/she can enter his/her information. The registration interface is constructed using HTML form for the web interface, Cascaded Stylesheet (CSS) that is used to provide the visual layout styles of different HTML elements and Javascript functions that are used in validating users’ password.

4.1.4 Dashboard Interface

This is another interface that the users(Teacher, Student, and Admin) encounter when they are successfully logged into the accounts, either by Login or Signup.

The user view and the further functionalities of this dashboard are dependent on the role of the user i.e. Student, Teacher, or Admin. The users shall interface basic forms with a number of fields to be filled or selected from dynamic dropdowns to proceed to the next page.

4.2 Hardware Interfaces

It should be any device that could link the user to the website of the E-Practical Submission System. The system should work on almost all mobile devices, home desktops, and laptop computers which support JavaScript and HTML5.

This Information system will work more efficiently on the following hardware specifications

- i) Hard disk – at least 64 GB
- ii) RAM – at least 500MB
- iii) Keyboard – Standard keyboard for interface
- iv) Mouse – Standard mouse with two buttons
- v) LAN / Wi-Fi connection for interacting with the website

4.3 Software Interfaces

The system will be intended to run on Firefox 4 and above, Google Chrome 10 and above, and Internet Explorer 8 and above.

4.4 Communications Interfaces

The communications functions required by this online practical management and submission system is an internet connection, so that the Admin, Student, and Teacher can interact with each other. It uses TCP/IP protocol.

5. Other Non-functional Requirements

A non-functional requirement defines the performance attribute of a software system. Types of Non-functional requirements are Scalability Capacity, Availability, Reliability, Recoverability, Data Integrity, etc.

5.1 Performance Requirements

Performance requirements define how well the system performs certain functions under specific conditions. Examples are speed of response, throughput, execution time and storage capacity. The service levels comprising performance requirements are often based on supporting end-user tasks.

• Response Time:

1. For the portal to show up for the first time and for uninterrupted usage of the features there will be a time span of 1 second with an error rate of 0.1 second.
2. For a sign-up confirmation email the user will be notified within 1min with an error rate of 30seconds.
3. Feedback exchange between different users will be of around less than 1 minutes with an error rate of 0.1 seconds.
4. Failed login mails shall be in a range of 1 minute with an error rate of 30 seconds.
5. Practical Submissions from student's end to teacher will be in a range of 5 seconds with an error rate of 0.5 seconds.
6. Practical Evaluations from teacher's end to student will be in a range of 1 min with an error rate of 1second.

• Workload:

The figure below shows the arrival rate for requests to a website:

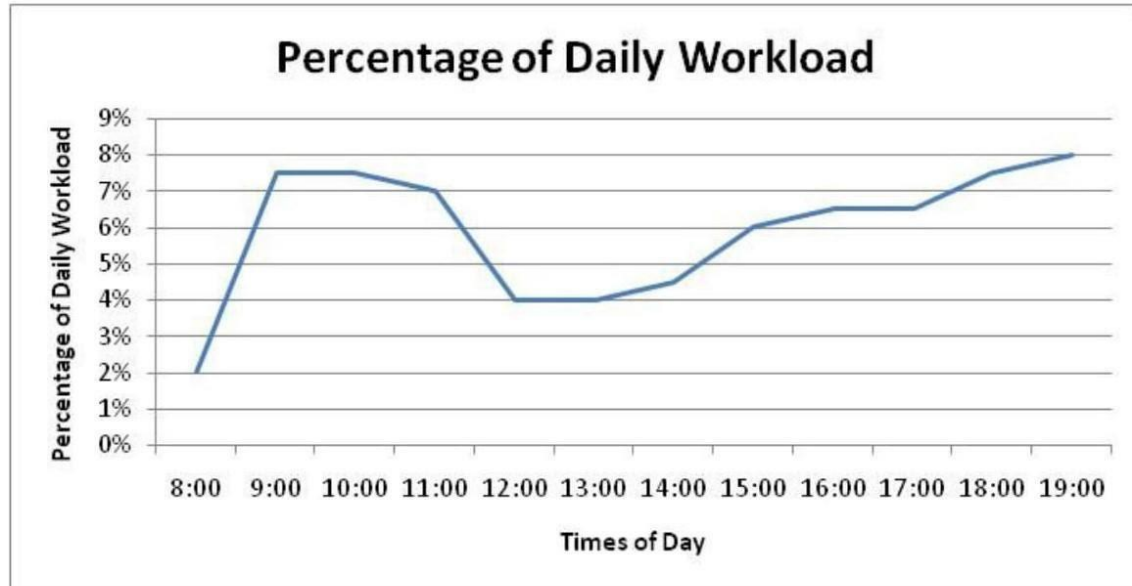


Figure No. 2: Percentage of Daily Workload

• Platform:

Hardware Utilities:

- i) Hard disk – at least 64 GB
- ii) RAM – at least 500MB
- iii) Keyboard – Standard keyboard for interface
- iv) Mouse – Standard mouse with two buttons
- v) LAN / Wi-Fi connection for interacting with the website

Operating System choices: Windows 7 and above/ MacOS /Linux

5.2 Safety Requirements

If there is damage to the database due to some failure, then we can't recover it. The students are advised to submit the file to the teacher and download the file at the same time once the file is generated to avoid any loss.

5.3 Security Requirements

Only students, teachers, and administrators of Thapar University, Patiala, can use the software. The user identity authentication requirement is email id given by the college.

A student, teacher, and administrator can log in only with the Thapar email id. This ensures security and prevents misuse of software.

5.4 Software Quality Attributes

1. Availability - All the experiments should be listed with all the details in advance so that whenever students perform an experiment, they can make the e-file.
2. Maintainability - The administrator should ensure all the details are correct.
3. Usability - The portal should satisfy the majority of teachers and students.
4. Correctness - Each experiment details like apparatus, procedure, diagrams, etc. should be correct.

6. Other Requirements

Database requirements for the software are- experiments of all the subjects, according to branch, year, and course of the student. The experiment should contain all the data like procedure, materials required, theory, precautions, diagrams, etc.

The project can be used at various colleges and this will help various students in focusing more on practicals and reduce manual writing work.

Appendix A: Glossary

- SRS: This is the document that gives an overview of the entire product to be delivered to the user. These are provided with information-driven diagrams that make it easy for the user, developer/designer, tester for getting a good catch over that product.
- DFD: These are the Data Flow Diagrams specifically designed to provide how each module in the product falls under the vicinity of others. Basically, it gives the flow of software information between all modules.
- Use-Case Diagram: It is a representation of a user's interaction with the system that shows the relationship between the user and the different 'use cases' in which the user is involved. Such diagrams can identify different types of users of our system and the different use cases.
- Class-Diagram: So basically, it is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations (or methods), and the relationships among objects.
- Activity-Diagram: portrays the control flow from a start point to a finish point showing the various decision paths that exist while the activity is being executed. In a nutshell, it depicts the behaviour of a system

Appendix B: Analysis Models

All the necessary diagrams have already been included in the early part of the SRS document.

Appendix C: Issues List

These are the issues that remain yet to be resolved and will be dealt with progressively:

1. Complete list of experiments for the concerned group of users.
2. Database of users remains pending which will be created on subsequent sign-ups.

3. A serious issue of information breach or uninvited users is a problem to be concerned about.
4. Security protocols over which the server is designed may not remain up-to-date.

3. Structured Analysis

In software engineering, structured analysis and structured design are methods for analyzing business requirements and developing specifications for converting practices into computer programs, hardware configurations, and related manual procedures.

3.1 Data Flow Diagrams

Data flow diagrams are used to graphically represent the flow of data in a business information system

3.1.1 DFD Level 0

DFD L-0 (context diagram), shows a data system as a whole and emphasizes the way it interacts with external entities. Here the external entities: 'Student', 'Teacher', 'Admin' are shown how they interact with the 'System Process' i.e. E-Practical Submission.

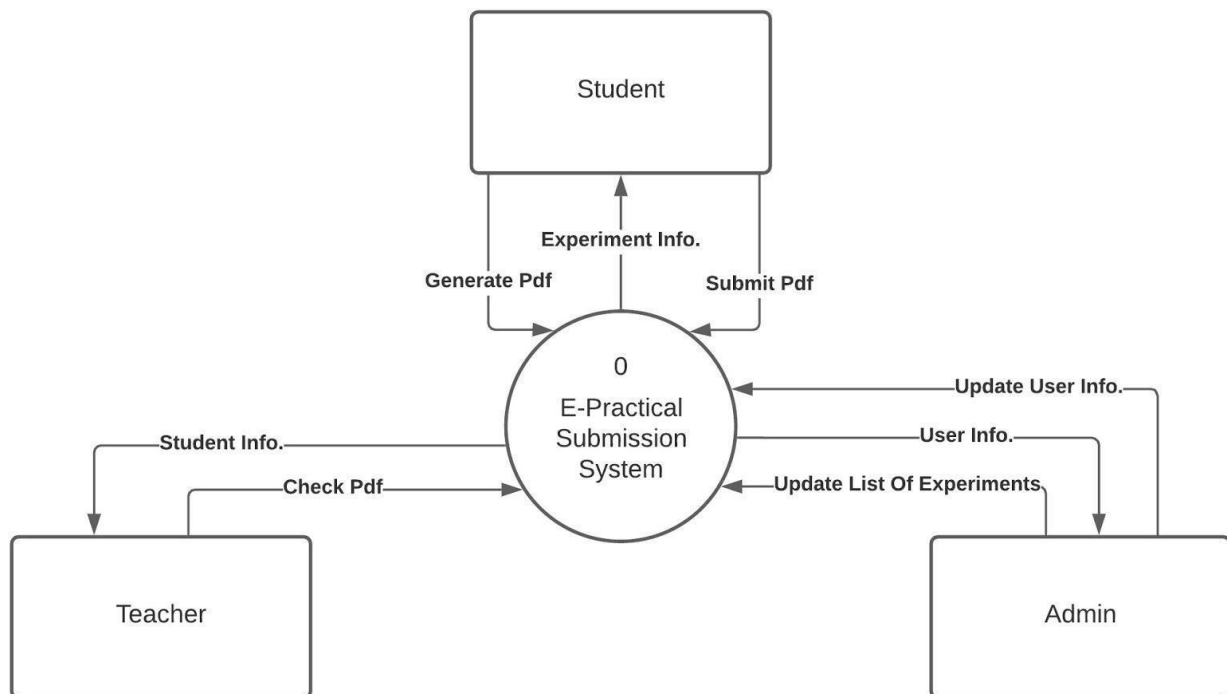


Figure 3.1: Data flow diagram Level-0

3.1.2 DFD Level 1

DFD L-1 breaks down the main processes into subprocesses that can then be analyzed and improved on a more intimate level. This breaks down the complete flow into 5 subprocesses including Pdf Generation, Practical Submission, Practical Checking, User info update, and last but not the least Update Experiments.

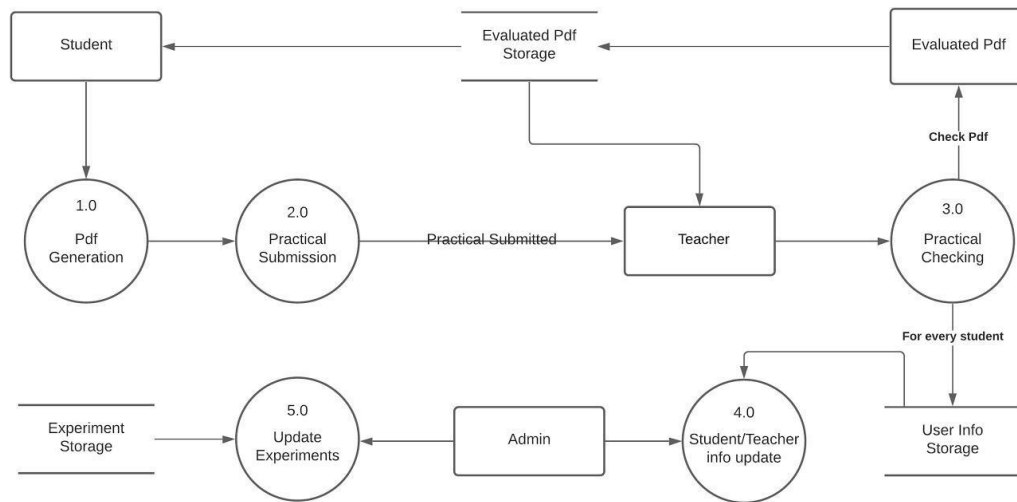


Figure 3.2: Data flow diagram Level-1

3.1.3 DFD Level 2

A level 2 data flow diagram (DFD) offers a more detailed look at the processes that make up an information system than a level 1 DFD does. It can be used to plan or record the specific makeup of a system. Here the sub-processes have been broken down into the final 9 processes.

4 Object Oriented Analysis

4.1 Use Case Diagram

A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved.

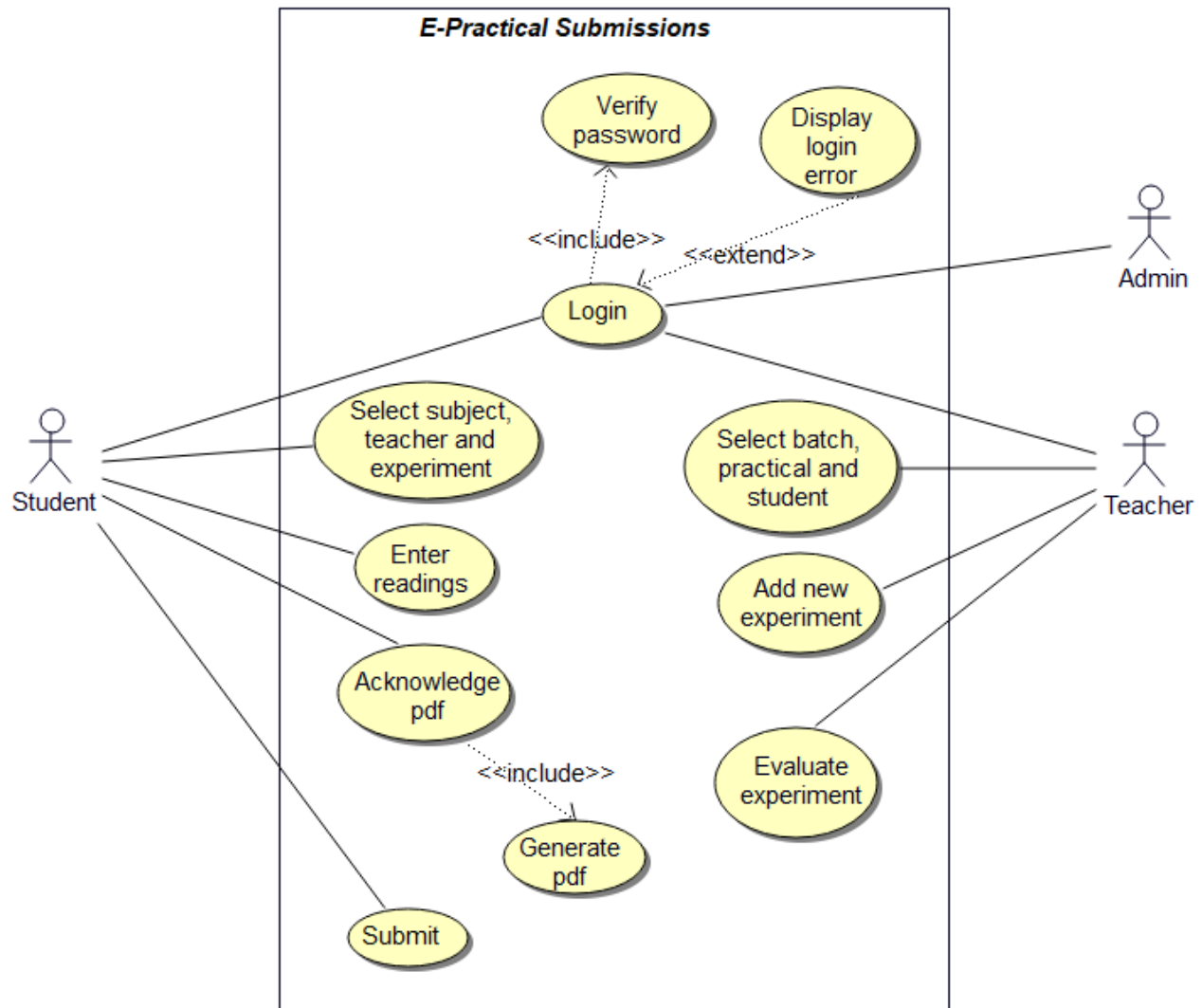


Figure 5: Use Case Diagram

4.2 Use Case Templates

Use Case Template is a business document which provides a story of how a system, and its actors, will be utilized to achieve a specific goal.

This Use Case template provides you with everything you need to develop your Use Case Document.

Use Case:	E-Practical Submission
Use Case ID:	1
Actors:	Student, Teacher, Admin
Goal in Context:	To reduce the manual work of writing and submitting practical files
Preconditions:	The user should be logged in the system.
Task Sequence:	<ol style="list-style-type: none"> 1. The student logs onto the E-Practical Submission Website. 2. He/ She selects the subject and experiment. 3. Student enters the observations and can submit it to teacher. 4. Student can also generate pdf of the file. 5. The teacher logs in the system. 6. They can choose the batch and evaluate students accordingly. 7. Students can see the marks given by teacher. 8. The admin can log in to the system. 9. They can update, add and delete experiments.
Post Conditions:	<ol style="list-style-type: none"> 1. Student can have pdf of files and observation tables. 2. Student can check their marks. 3. Teacher has information of full batch.

Table No. 1: Use Case Template for E-Practical Submissions

4.3 Activity Diagram

Activity diagrams are graphical representations of workflows of stepwise activities and actions with support for choice, iteration and concurrency.

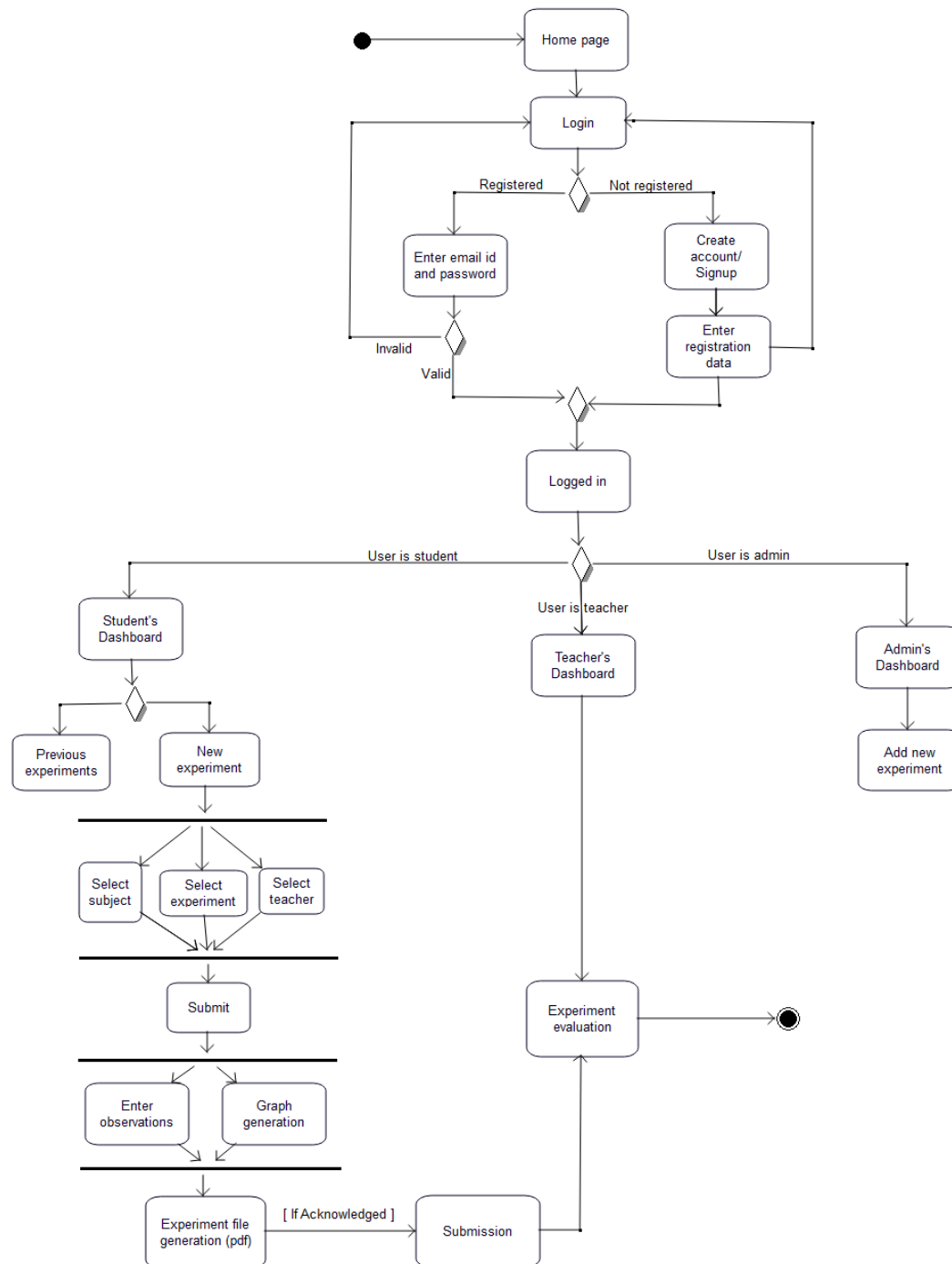


Figure 6: Activity Diagram

4.4 Collaboration Diagram

A collaboration diagram, is an illustration of the relationships and interactions among software objects in the Unified Modelling Language (UML). These diagrams can be used to portray the dynamic behavior of a particular use case and define the role of each object.

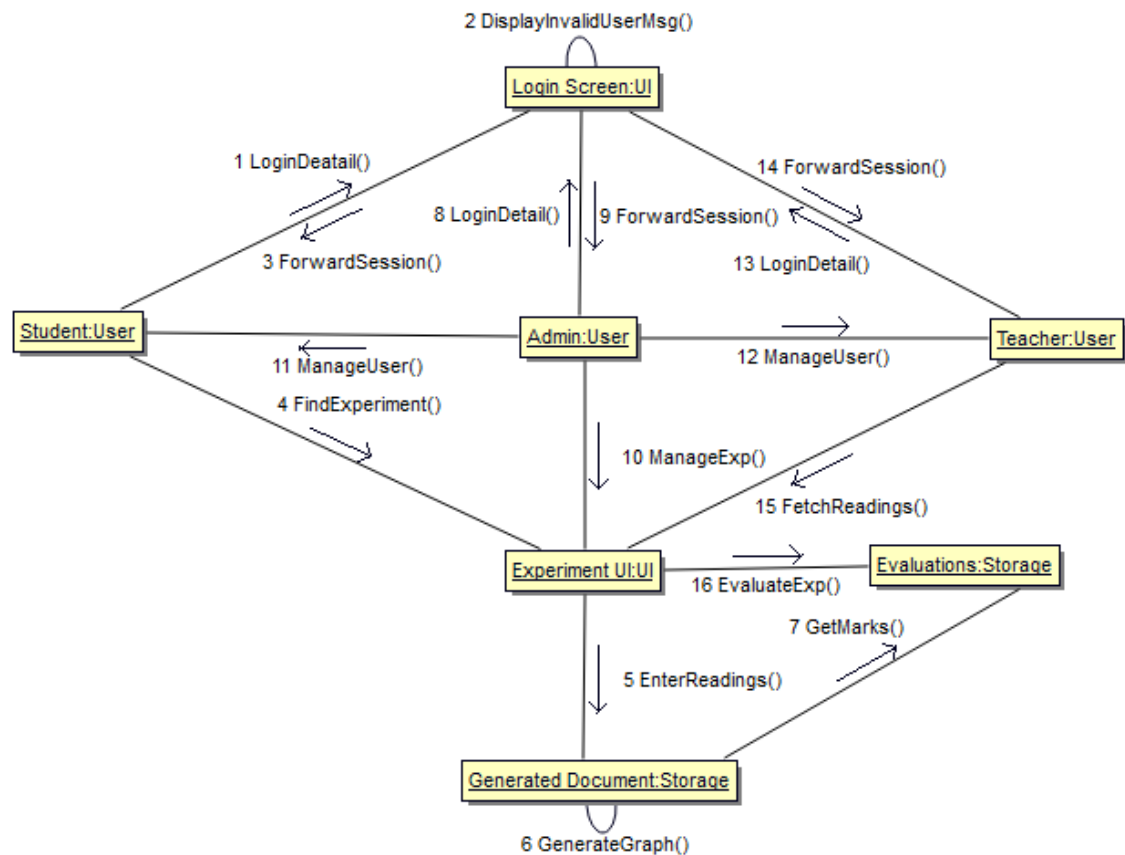


Figure 7: Collaboration Diagram

4.5 Sequence Diagram

A sequence diagram is a type of interaction diagram because it describes how—and in what order—a group of objects works together.

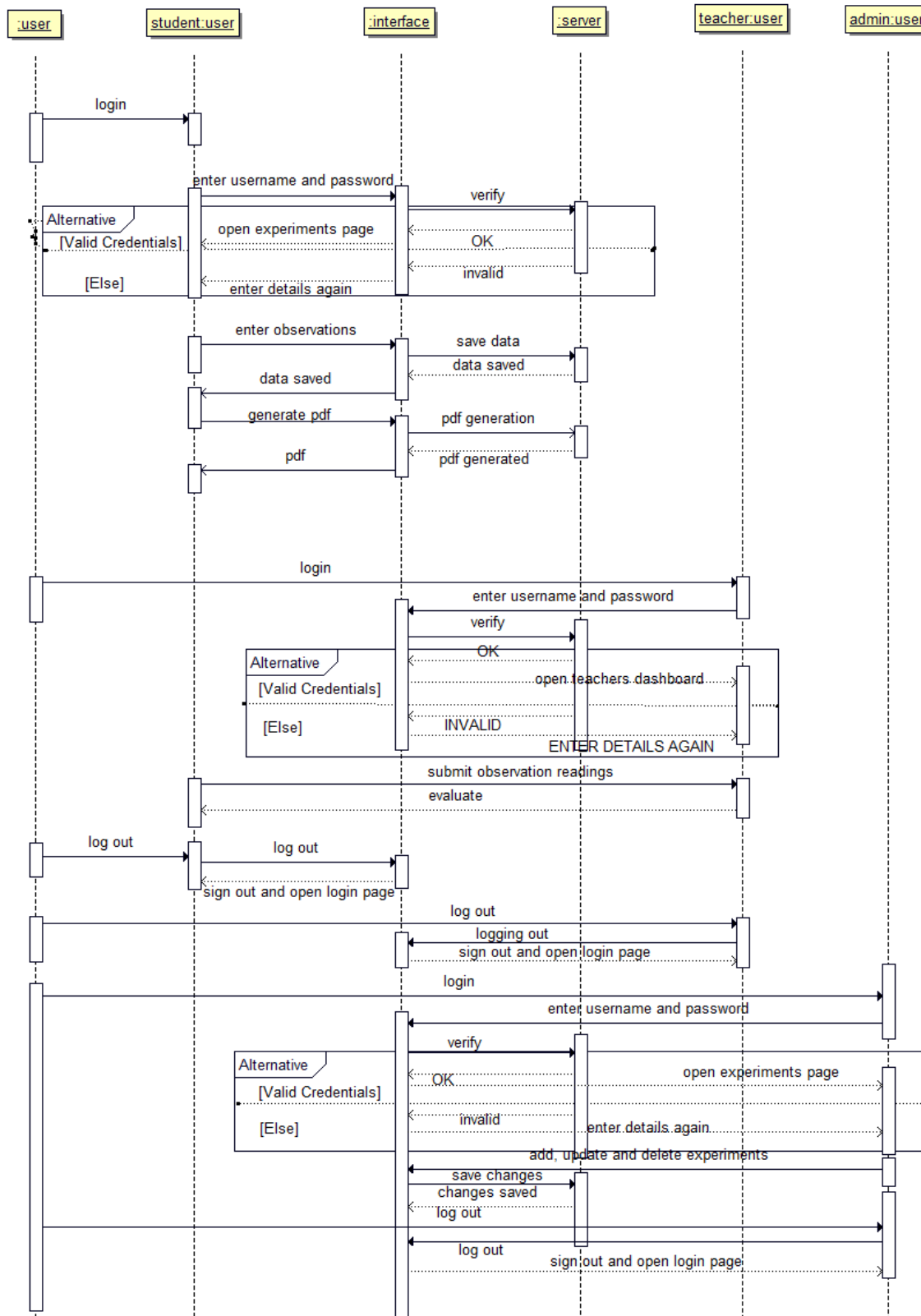


Figure 8 : Sequence Diagram

4.6 State Chart Diagram

A State chart diagram describes a state machine. State machine can be defined as a machine which defines different states of an object and these states are controlled by external or internal events.

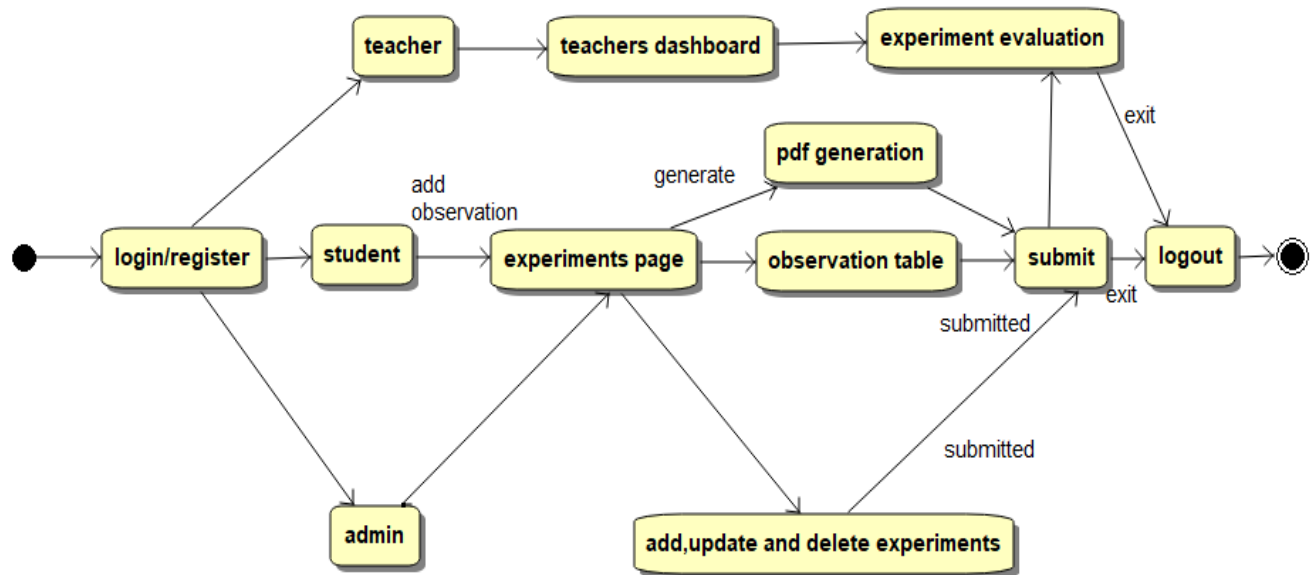


Figure 9 : State Chart Diagram

4.7 Component Diagram

Component diagrams are often drawn to help model implementation details and double-check that every aspect of the system's required function is covered by planned development.

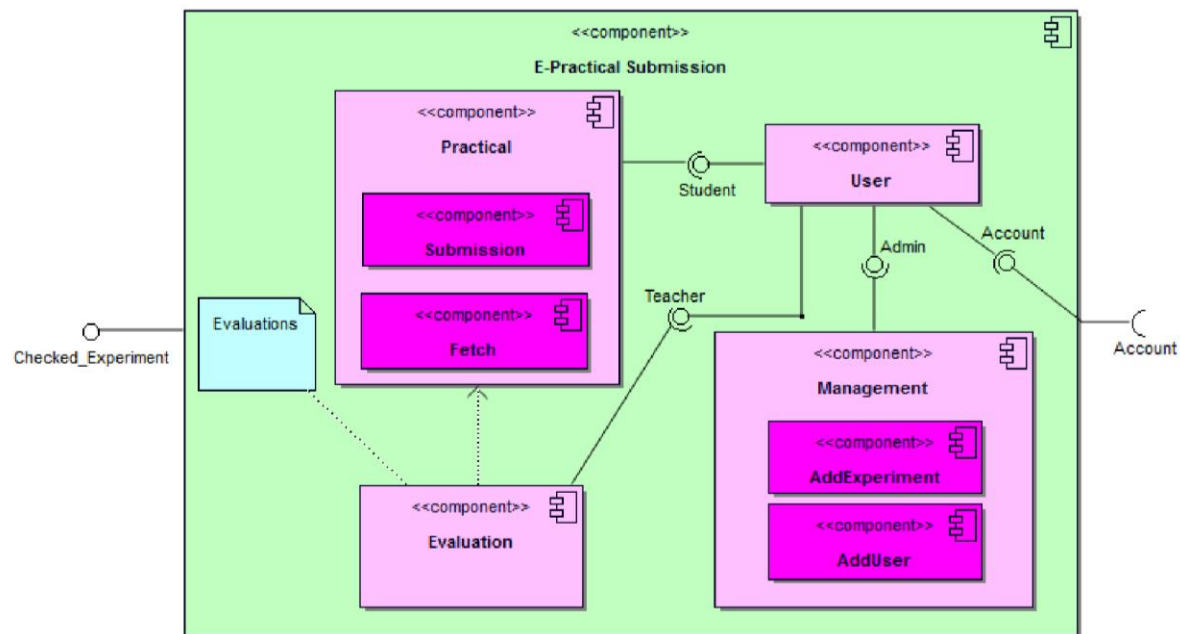


Figure 10: Component Diagram

4.8 Deployment Diagram

A deployment diagram is a UML diagram type that shows the execution architecture of a system, including nodes such as hardware or software execution environments, and the middleware connecting them.

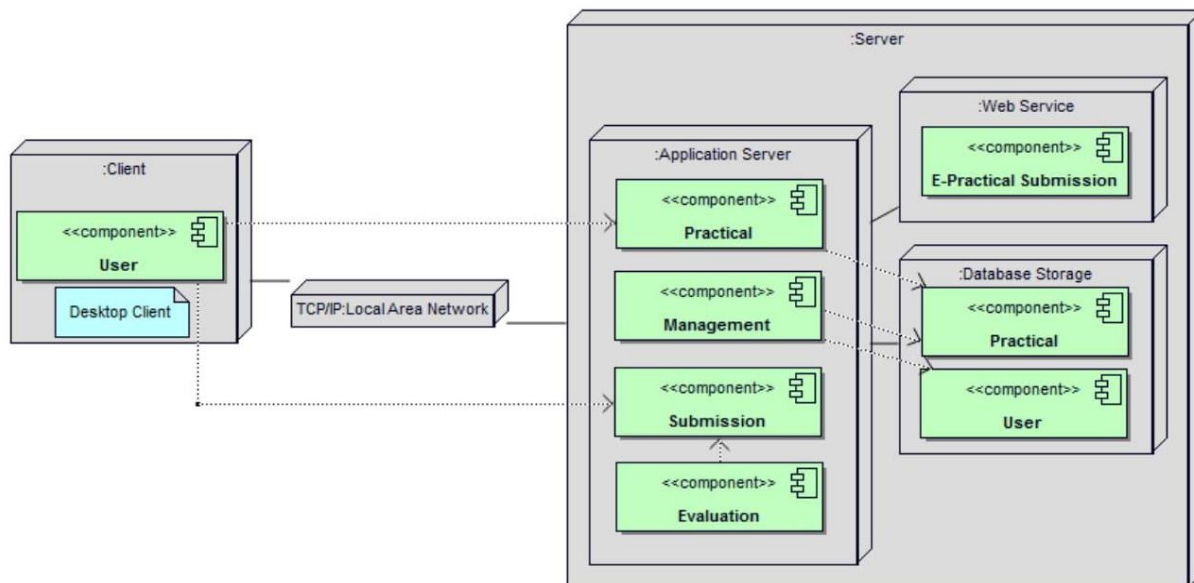


Figure 11 : Deployment Diagram

5 Testing

Software Testing is evaluation of the software against requirements gathered from users and system specifications. Testing is conducted at the phase level in software development life cycle or at module level in program code. Software testing comprises Validation and Verification.

5.1 Test Plan

A **Test Plan** is a detailed document that describes the test strategy, objectives, schedule, estimation, deliverables, and resources required to perform testing for a software product.

Test Plan helps us determine the effort needed to validate the quality of the application under test.

The test plan serves as a blueprint to conduct software testing activities as a defined process, which is minutely monitored and controlled by the test manager.

Testing Objective

The objective of the testing is to verify that the functionality of E-Practical Submissions System works according to the specifications.

The test will execute and verify the authentication as well as the main functionalities of the project, identify, fix and retest all high and medium severity defects considering all constraints.

Testing Strategy

To deliver a bug free software, a tester should cover maximum possible combinations of conditions. But it is not possible to completely exhaust all possible conditions. It is not only time consuming but also, it is a waste of resources.

The overall goal is to test that all the components work fine alone and together with other components. We will perform testing on the Login Form, Registration Form, Reset Password Form and Upload Image Form.

Parameter	Description
Identifier: Version Number	V 1.0.0
Test Items	<p>Subsystems to be tested are:</p> <ul style="list-style-type: none"> ● Registration route (Student, Teacher) ● Login route (Student, Teacher, Admin) ● Submit exp. readings (Student) ● Experiment Evaluation (Teacher) ● Modify user records (Admin)
Features to be tested	<ul style="list-style-type: none"> ● The data is getting saved to the database. ● The data is getting accessed from the database. ● The forms only accept valid data. ● All the routes redirect accordingly.
Approach	Each feature will be tested multiple times using relevant and non-relevant test cases.

Table No. 2: Test Plan for E-Practical Submissions

5.2 Test Case Report

Test Case #: 1.1

**System: E-Practical
Submissions**

Test Case Name: Register User

Subsystem: Registration Form

Design Date: 27/09/2020

Page: 1 of 5

Designed by: Group 3

Executed by: Group 3

Execution Date: 30/11/2020

Short Description: Testing Signup Page

Preconditions:

The user has accessed the E-Practical Submissions website by opening it in any browser.

The user does not have a prior account.

The system displays the 'Home' page of the EPS website.

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click the 'Signup' button	The system displays the signup page which has the registration form.	Pass	
2	Enter valid values in fields Username, Roll No, Email, Password, Course and click 'Submit' button.	The system displays a message of successful operation. The system displays the login page.	Pass	
3	Check post-condition 1			
4	Enter invalid values in fields Username, Roll No, Email, Password, Course and click 'Sign Up'	The system displays a message of unsuccessful operation and asks the user to correct the invalid fields with specific information on how to correct invalid fields.	Pass	
	Repeat step 2	The system displays a message of successful operation. The system displays the login page.	Pass	
5	Check post-condition 2			
6	Repeat steps 1,2 using some fields as empty and click 'Sign Up'	The system displays a message to fill empty fields.	Pass	
7	Fill the empty fields with valid data and click 'Sign Up'	The system displays a message of successful operation. The system displays the login page.	Pass	
8	Check post-condition 2			

Post-conditions:

The new user details are saved in the database.

The new user details are saved in the database.

Test Case #: 1.2

System: **E-Practical
Submissions**

Designed by: **Group 3**

Executed by: **Group 3**

Short Description: **Testing the Login Page**

Test Case Name: **Login User**

Subsystem: **Login Form Design**

Date: **27/10/2020**

Execution Date: **25/11/2020**

Page: **2 of 5**

Preconditions:

1. The user has accessed the E-Practical Submissions website by opening it in any browser.
2. The user has registered already and has his account login credentials.
3. The system displays the 'Home' page.

Step	Action	Expected System Response	Pass/ Fail	Comment
1	Click the 'Login' button	The system displays the login page which has the login form.	Pass	
2	Enter valid values in fields Username, Password and click 'Login'	The system displays a message of successful operation. The system displays the 'Dashboard' page.	Pass	
3	Check post-condition 1			
4	Enter invalid values in fields Username, Password and click 'Login'	The system displays a message of unsuccessful operation and asks the user to enter valid Username and Password.	Pass	
5	Enter valid values in fields Username, Password and click 'Login'	The system displays a message of successful operation. The system displays the 'Dashboard' page.	Pass	
6	Check post-condition 2			
7	Repeat steps 1,2 using some fields as empty and click 'Login'	The system displays a message to fill empty fields.	Pass	
8	Fill the empty fields with valid data and click 'Login'	The system displays a message of successful operation. The system displays the 'Dashboard' page.	Pass	
9	Check post-condition 3			

Post-conditions:

1. User is now logged in his account.
2. User is now logged in his account.
3. User is now logged in his account.

Test Case #: 1.3System: **E-Practical Submissions**Designed by: **Group 3**Executed by: **Group 3**Short Description: **Testing the submit exp. readings functionality**Test Case Name: **Submit Exp. Readings (User: STUDENT)**Subsystem: **Experiment**Design Date: **27/10/2020**Execution Date: **25/11/2020**Page: **3 of 5****Preconditions:**

1. The user has accessed the E-Practical Submissions website by opening it in any browser.
2. The user has a prior account and is already logged in with that account.
3. The system displays the 'Dashboard' page according to the type of user.

Step	Action	Expected System Response	Pass/Fail	Comment
1	Click the 'Open Experiment File' button	The system displays the Experiment file.	Pass	
2	Click on the 'Enter Readings and Result' button	The system displays the readings form on another page.	Pass	
3	Enter valid Roll No, Exp. No., Exp. Readings, and Exp. Result in the fields and click on 'Submit' button	The Roll No is validated with the user accounts database. The system displays the "Student Experiment Result/Grades table".	Pass	
4	Check post-condition 1			
5	Repeat steps 1-3 using invalid Roll No. or readings.	The system displays a message of unsuccessful operation and asks the user to use valid operators.	Pass	
6	Repeat steps 3-5 using valid Roll No. and readings.	The system displays a message of successful operation. The system displays the "Student Experiment Result/Grades table".	Pass	
7	Check post-condition 1			
8	Repeat step 1, and in step 2 the user clicks on "Download Exp File as PDF" button.	The system converts the experiment file to PDF format and downloads it to the student user.	Pass	
9	In step 1, the user clicks the 'Check Result' button.	The system displays a message of successful operation. The system displays the "Student Experiment Grades table".	Pass	
10	Check post-condition 2			

Post-conditions:

1. The readings of the experiment entered by the student are saved in the database.
2. The student grades stored in the database are accessed.

Test Case #: 1.4System: **E-Practical Submissions**Designed by: **Group 3**Executed by: **Group 3**Test Case Name: **View Readings and Evaluate (User: TEACHER)**Subsystem: **Experiment**Design Date: **27/10/2020**Execution Date: **25/11/2020**Page: **4 of 5****Short Description: Testing the View student readings and Evaluate functionalities for teacher****Preconditions:**

The user has accessed the E-Practical Submissions website by opening it in any browser.

The user has a prior account and is already logged in with that account.

The system displays the 'Teacher Dashboard' page according to the type of user.

Step	Action	Expected System Response	Pass/Fail	Comment
1	User clicks the 'Check All Submitted Assignments' button	The system displays the "Student Experiment Readings/Result" table with EDIT/DELETE rights.	Pass	
2	Click on the 'EDIT' button in front of any student.	The system displays the UPDATE form to mark grades of that student.	Pass	
3	Enter valid grades/marks and click on 'Update' button.	The student is evaluated successfully. The system displays the "Student Experiment Readings/Result" table with EDIT/DELETE rights.	Pass	
4	Check post-condition 1			
5	Repeat steps 1 and 2, and click on 'Download as PDF' button in step 2.	The system converts the Student Result table to PDF format and downloads it to the Teacher user.		
6	Repeat steps 1-3 using invalid grades/marks and click on 'Update' button in step 3.	The system displays a message of unsuccessful operation and asks the user to use valid operators.	Pass	
7	Repeat steps 1-3 using valid grades/marks and click on 'Update' button in step 3.	The system displays a message of successful operation. The system displays the "Student Experiment Readings/Result" table with EDIT/DELETE rights.	Pass	
8	Check post-condition 1			
9	Repeat steps 1, 2, 3 and click on "DELETE" button in front of any student in the table.	The system shows successful DELETE operation. The system displays the table with record deleted.	Pass	
10	Check post-condition 2			

Post-conditions:

1. The marks/grades of the student entered by the teacher are saved in the database.
2. The record of readings and marks of the student is deleted from the database.

Test Case #: 1.5System: **E-Practical Submissions**Designed by: **Group 3**Executed by: **Group 3**Short Description: **Testing the EDIT/DELETE Student/Teacher records functionality for ADMIN**Test Case Name: **EDIT Records (User: ADMIN)**Subsystem: **Experiment**Design Date: **27/10/2020**Execution Date: **25/11/2020**Page: **5 of 5****Preconditions:**

1. The user has accessed the E-Practical Submissions website by opening it in any browser.
2. The user has a prior account and is already logged in with that account.
3. The system displays the 'ADMIN Dashboard' page with ADMIN privileges.

Step	Action	Expected System Response	Pass/Fail	Comment
1	User clicks the 'EDIT Teacher/Student Records' button	The system displays the "Teacher/Student Records" table with EDIT/DELETE rights.	Pass	
2	Click on the 'EDIT' button in front of any teacher/student.	The system displays the UPDATE form to make changes to that teacher/student record.	Pass	
3	Enter valid Username, Email Id, Course and click on 'Update' button.	The record is updated successfully. The system displays the "Teacher/Student Records" table with EDIT/DELETE rights.	Pass	
4	Check post-condition 1			
6	Repeat steps 1-3 using invalid Username, Email Id, Course and click on 'Update' button in step 3.	The system displays a message of unsuccessful operation and asks the user to use valid operators.	Pass	
7	Repeat steps 1-3 using valid Username, Email Id, Course and click on 'Update' button in step 3.	The record is updated successfully. The system displays the "Teacher/Student Records" table with EDIT/DELETE rights.	Pass	
8	Check post-condition 1			
9	Repeat steps 1, 2, 3 and click on "DELETE" button in front of any teacher/student in the table.	The system shows successful DELETE operation. The system displays the table with record deleted.	Pass	
10	Check post-condition 2			

Post-conditions:

1. The Username, Email Id and Course records of the Teacher updated by the ADMIN are saved in the database
2. The record of Username, Email Id, Course of the Teacher/Student is deleted from the database.

5.3 Screenshots

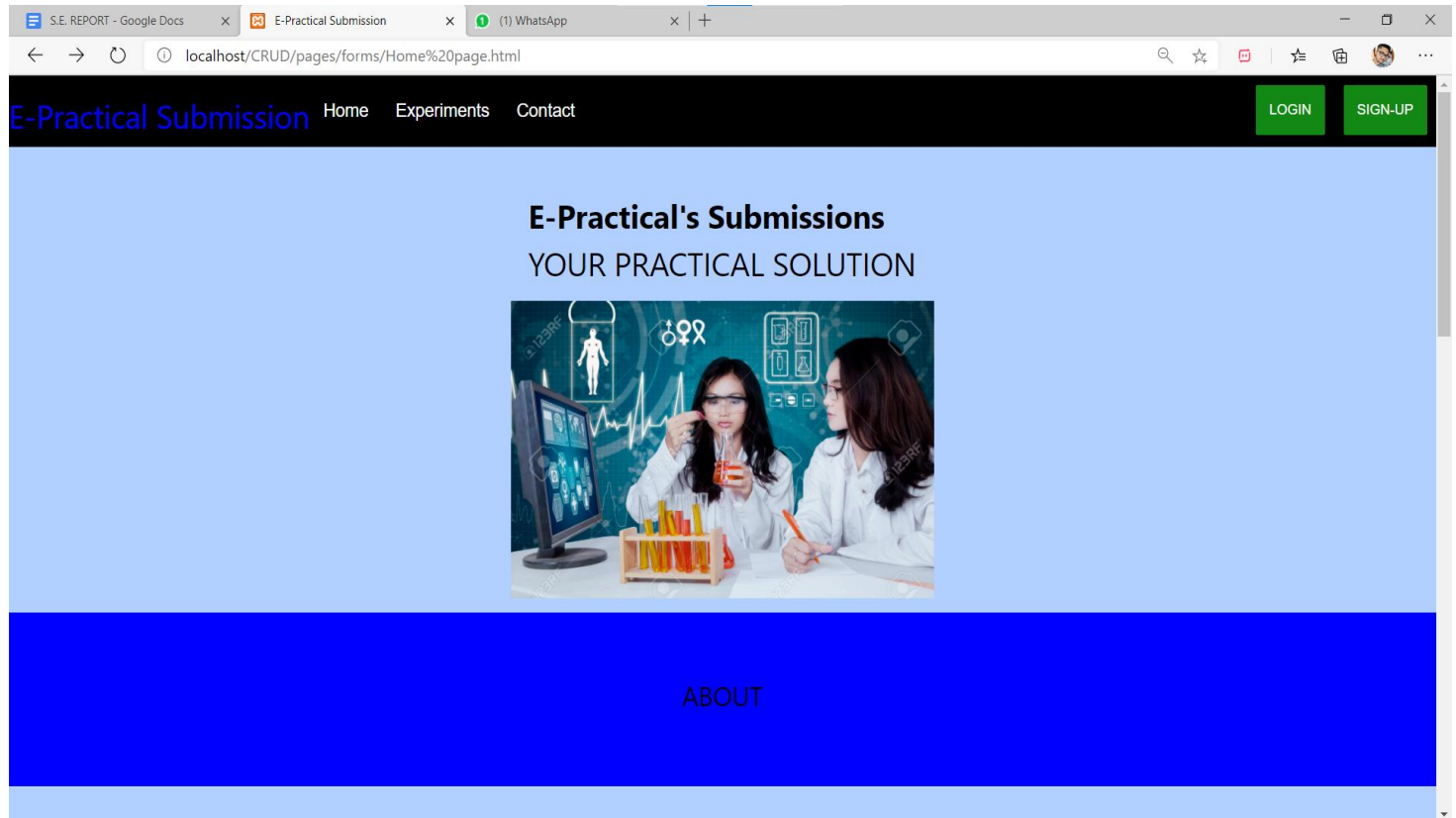


Figure 12 : Home Page

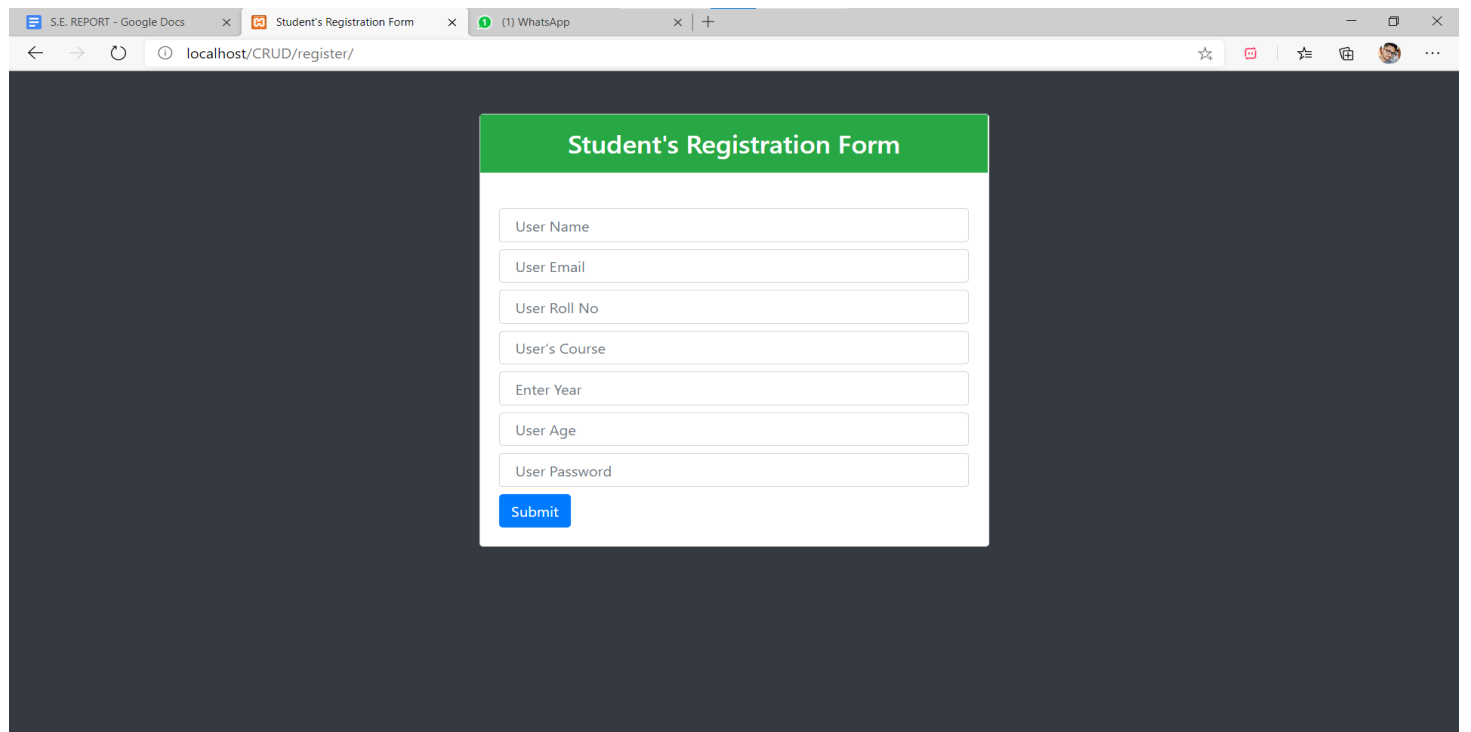
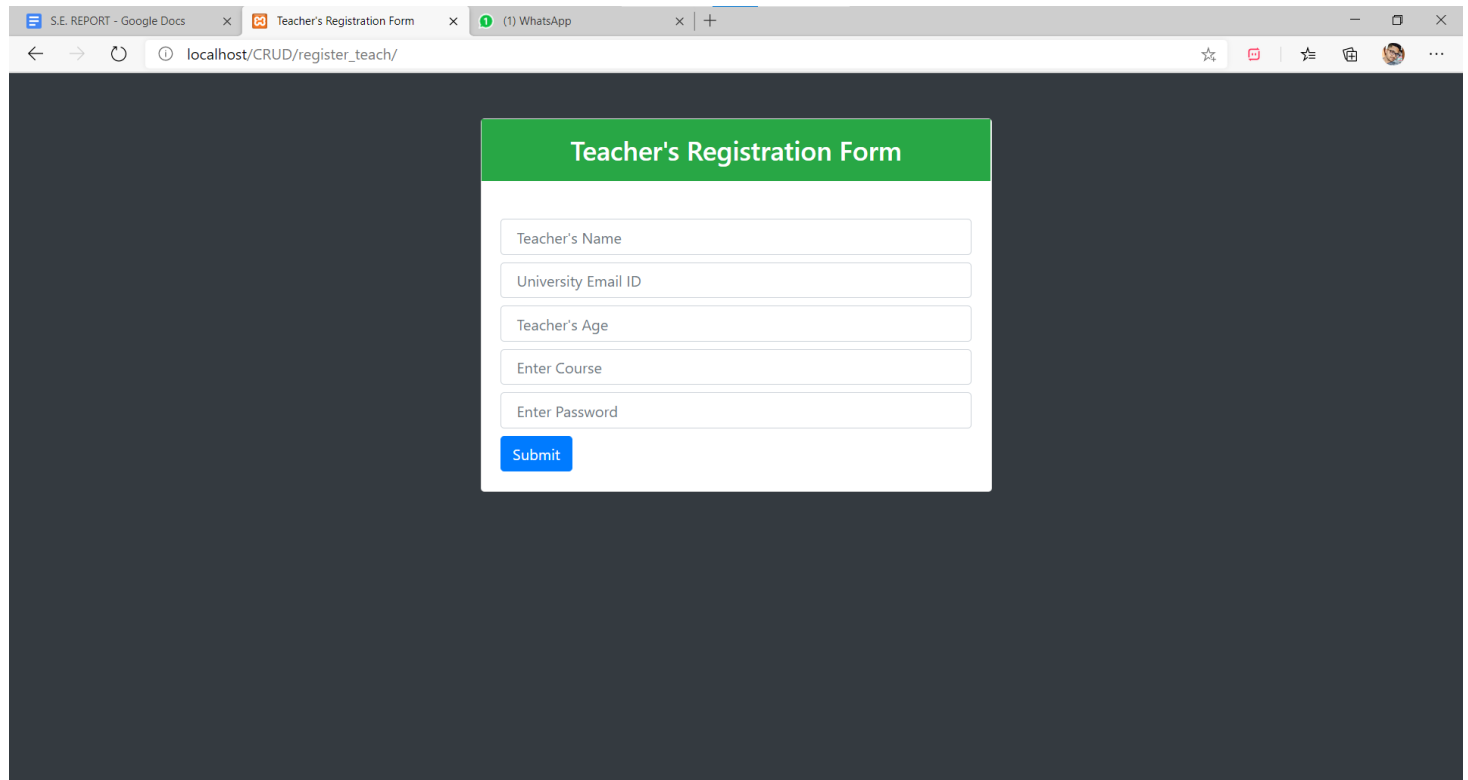
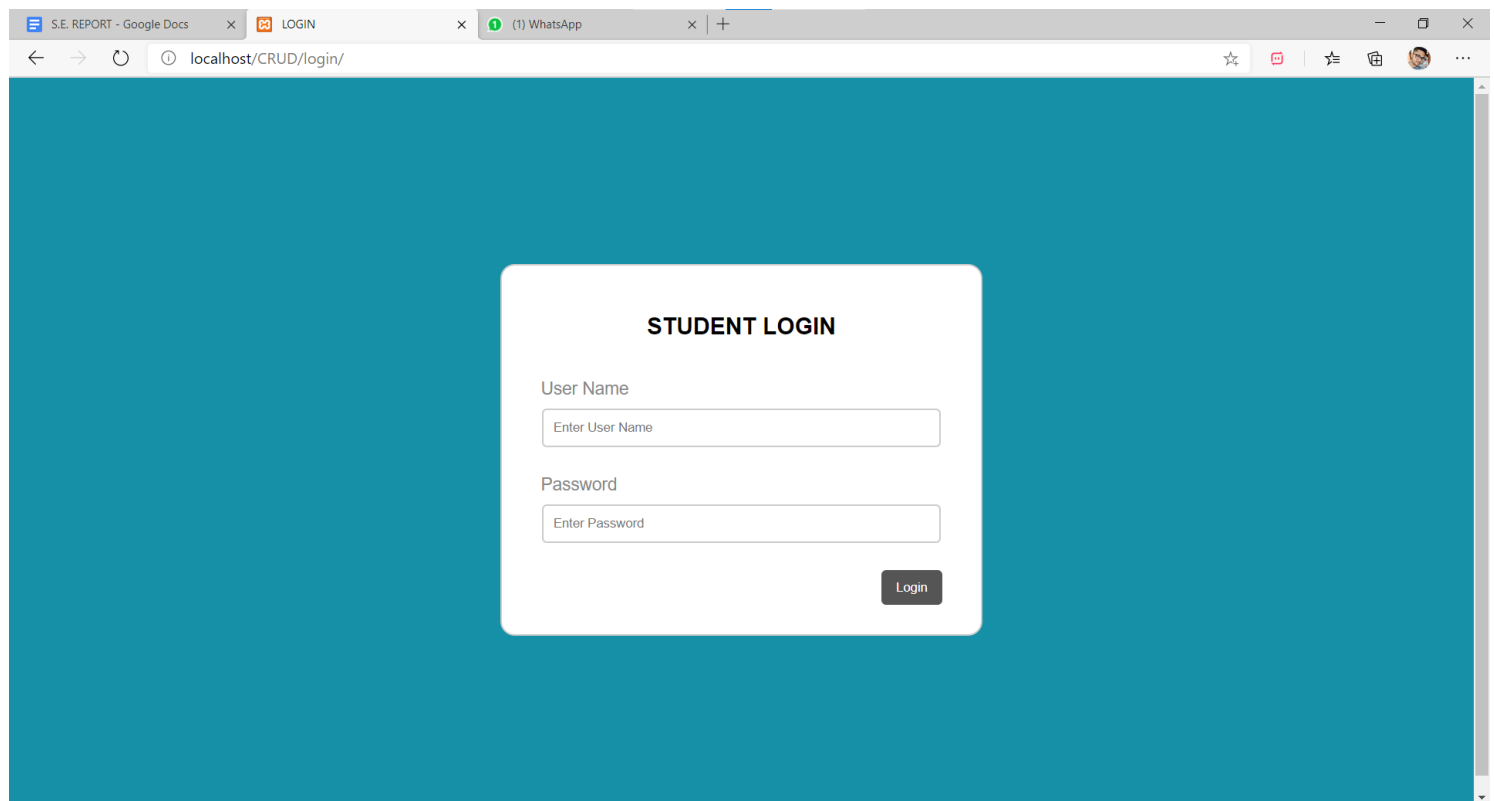


Figure 13 : Student Signup Page

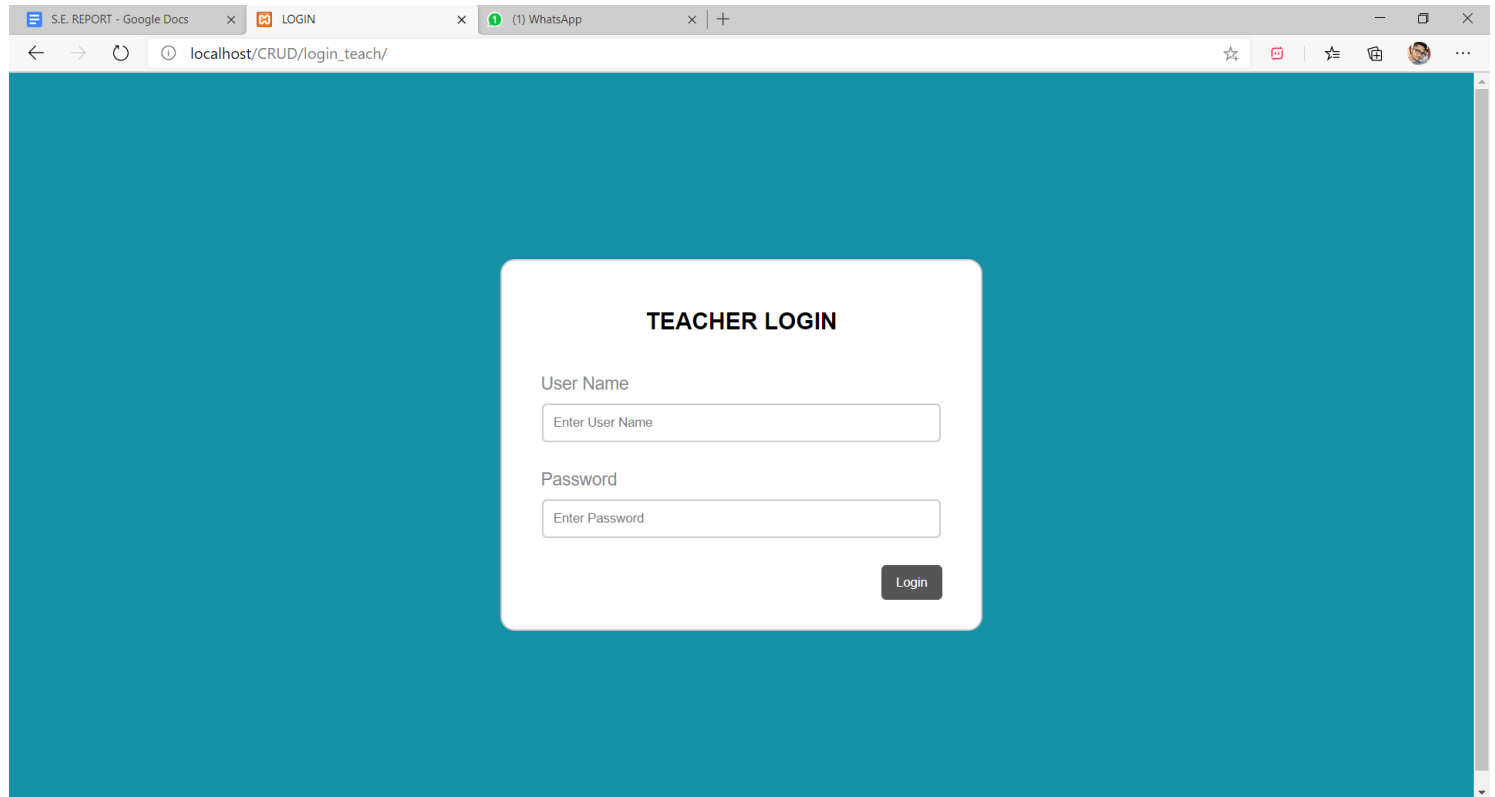


The screenshot shows a web browser window with the URL `localhost/CRUD/register_teach/`. The page features a dark gray background. In the center, there is a white rectangular form titled "Teacher's Registration Form" with a green header bar. The form contains five input fields: "Teacher's Name", "University Email ID", "Teacher's Age", "Enter Course", and "Enter Password". Below these fields is a blue "Submit" button.

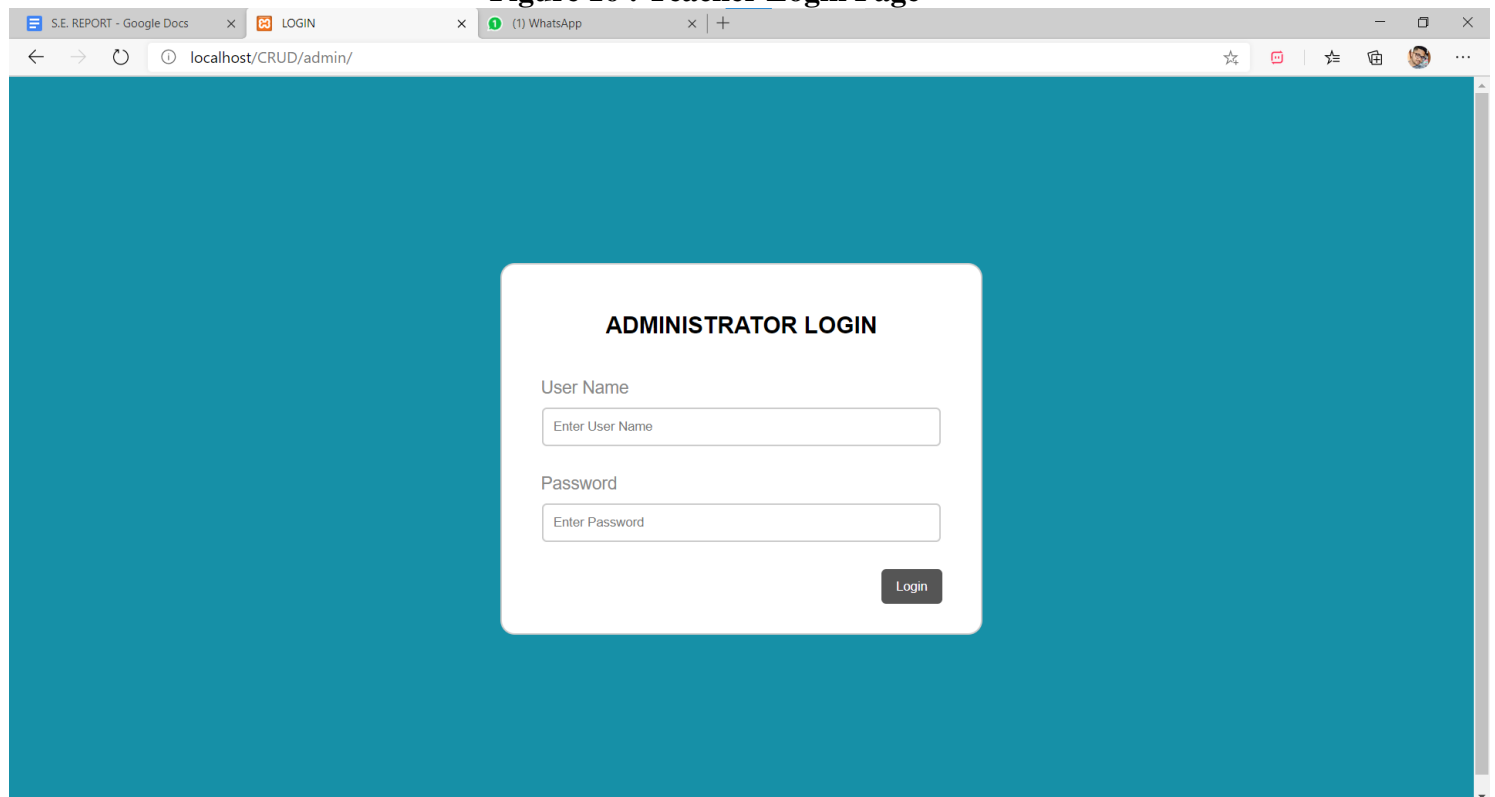
Figure 14 : Teacher Signup Page

The screenshot shows a web browser window with the URL `localhost/CRUD/login/`. The page has a solid teal background. In the center, there is a white rounded rectangular form titled "STUDENT LOGIN". The form contains two input fields: "User Name" and "Password". Below the "Password" field is a dark gray "Login" button.

Figure 15 : Student Login Page



A screenshot of a web browser displaying the 'TEACHER LOGIN' page. The browser's address bar shows 'localhost/CRUD/login_teach/'. The page has a teal background. In the center, there is a white rounded rectangle containing the title 'TEACHER LOGIN' in bold. Below the title are two input fields: 'User Name' with the placeholder text 'Enter User Name', and 'Password' with the placeholder text 'Enter Password'. A dark grey 'Login' button is positioned at the bottom right of the white box.

Figure 16 : Teacher Login Page

A screenshot of a web browser displaying the 'ADMINISTRATOR LOGIN' page. The browser's address bar shows 'localhost/CRUD/admin/'. The page has a teal background. In the center, there is a white rounded rectangle containing the title 'ADMINISTRATOR LOGIN' in bold. Below the title are two input fields: 'User Name' with the placeholder text 'Enter User Name', and 'Password' with the placeholder text 'Enter Password'. A dark grey 'Login' button is positioned at the bottom right of the white box.

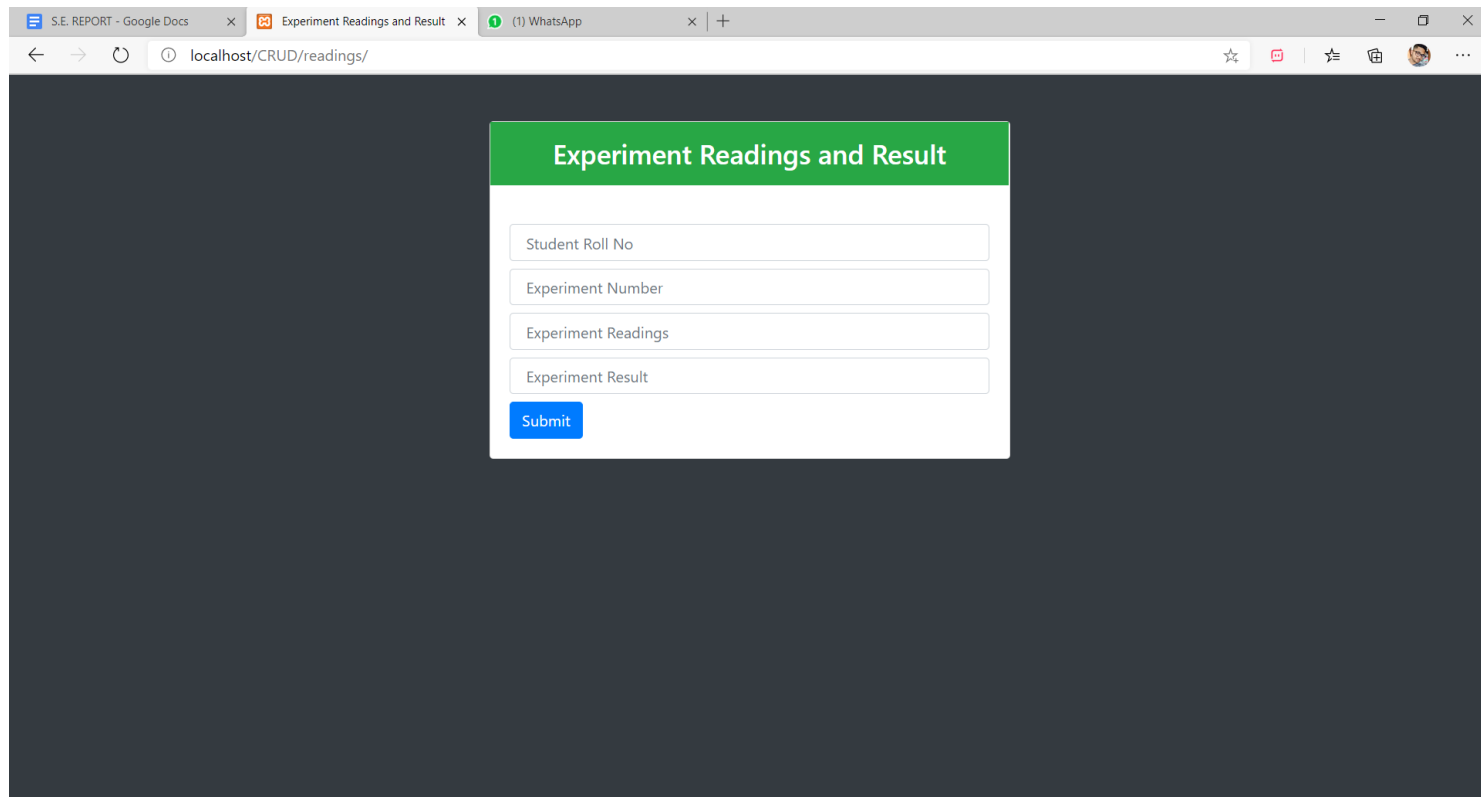
Figure 17 : Administrator Login Page

The screenshot shows a web application titled "Student Dashboard". The sidebar on the left contains the following links: Dashboard, Experiments (selected), New experiment, Previous submissions, Extras, Login, Register, Forgot Password, and a LOGOUT button. The main content area is titled "New Experiment" and includes a form with the following fields: "Subject Code" (a dropdown menu), "Experiment" (a dropdown menu), and "Teacher" (a dropdown menu). Below the form are three blue buttons: "Open Experiment File", "Check Result", and "Download Experiment file as PDF". The footer of the page displays the copyright notice "Copyright © 2020-2021 E-Practical Submissions. All rights reserved." and the version "Version 3.1.0-pre".

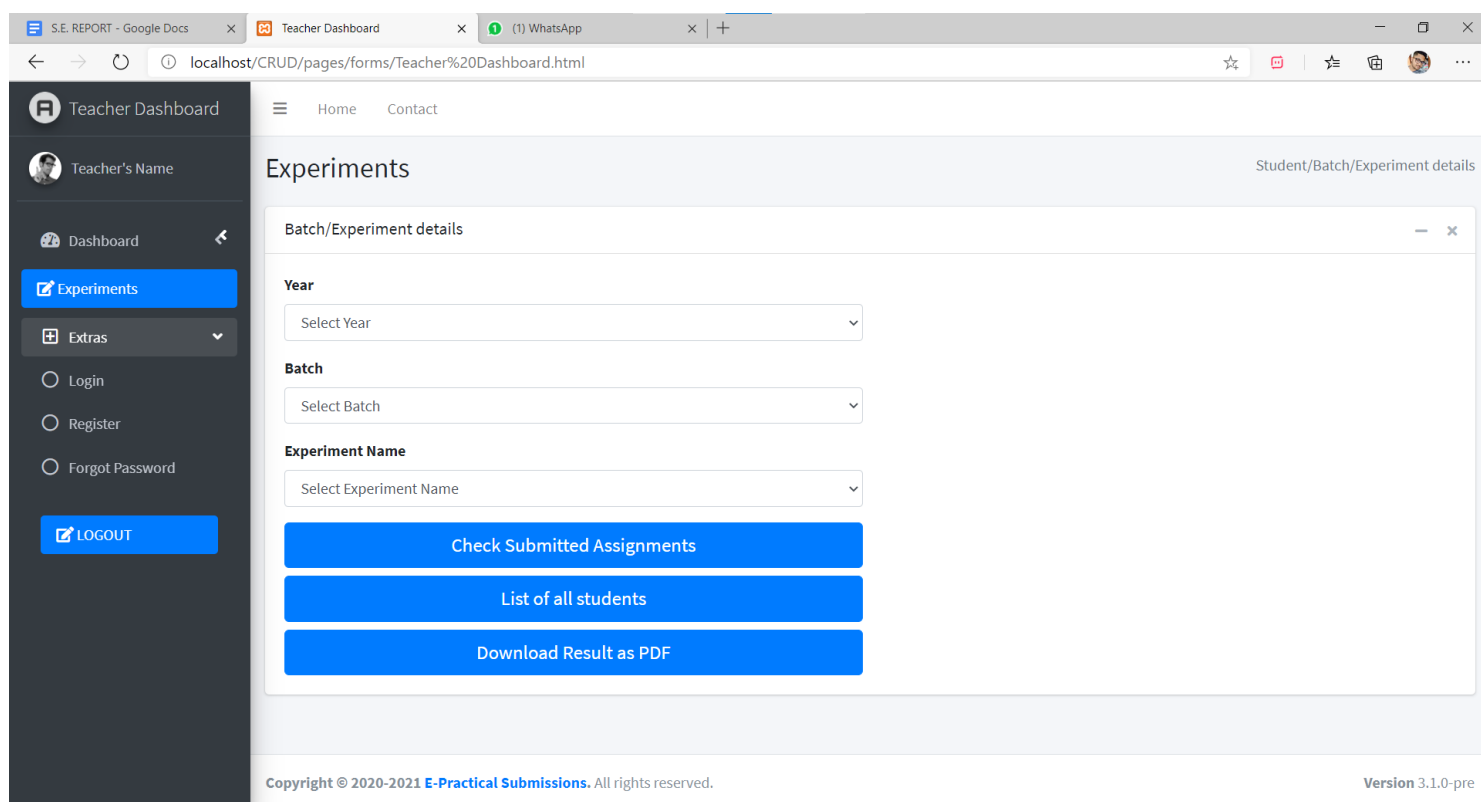
Figure 18 : Student Dashboard Page

The screenshot displays the "PHYSICS EXPERIMENT:1" page. It features a red header with the title. Below the header is a circuit diagram showing a battery, a rheostat, a plug key, and two ammeters connected in series. The page is divided into sections with blue headers: "AIM", "APPARATUS REQUIRED", and "THEORY". The "AIM" section states: "To determine the resistance per cm of a given wire by plotting a graph for potential difference versus current." The "APPARATUS REQUIRED" section lists: "A wire of unknown resistance", "Battery", "Voltmeter", "Milliammeter", "Rheostat", "Plug key", "Connecting wires", and "Piece of sandpaper". The "THEORY" section includes a photograph of the experimental setup and text explaining Ohm's law: "According to Ohm's law, the electric current flowing through a conductor is directly proportional to the potential difference across its ends, provided the physical state (pressure, temperature and dimensions) of the conductor remains unchanged. If i is the current flowing through the conductor and V is the potential difference across its end, then $V \propto i$ and hence $R = \frac{V}{i}$ ".

Figure 19 : Experiment Page

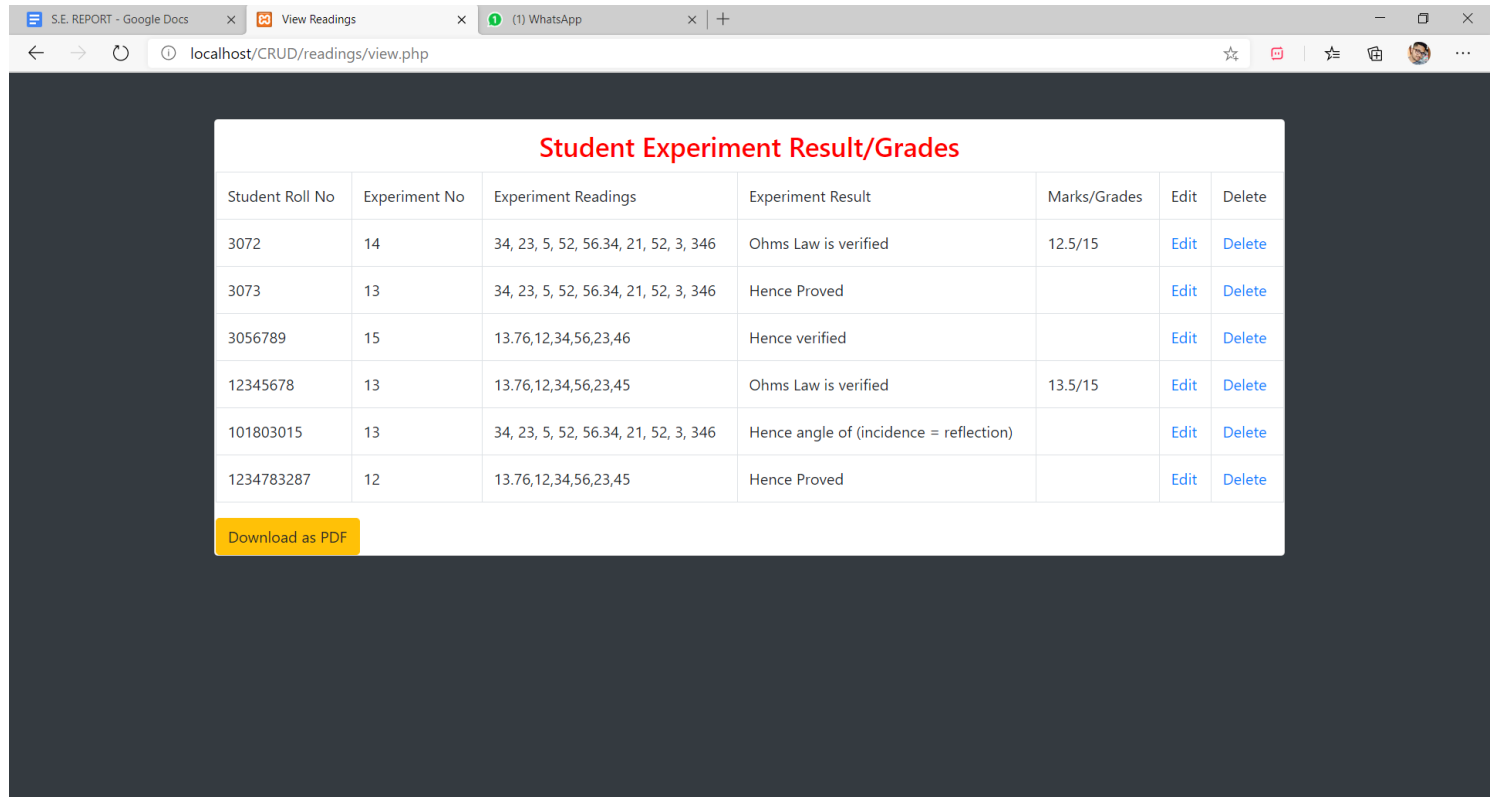


The screenshot shows a web browser window with three tabs: 'S.E. REPORT - Google Docs', 'Experiment Readings and Result', and '(1) WhatsApp'. The address bar shows 'localhost/CRUD/readings/'. The main content area features a dark grey background with a white form titled 'Experiment Readings and Result' in a green header. The form contains four input fields: 'Student Roll No', 'Experiment Number', 'Experiment Readings', and 'Experiment Result'. A blue 'Submit' button is located at the bottom of the form.

Figure 20 : Student Readings Submission Form

The screenshot shows a web browser window with three tabs: 'S.E. REPORT - Google Docs', 'Teacher Dashboard', and '(1) WhatsApp'. The address bar shows 'localhost/CRUD/pages/forms/Teacher%20Dashboard.html'. The page has a dark sidebar on the left with the following menu items: 'Teacher Dashboard', 'Teacher's Name', 'Dashboard', 'Experiments' (highlighted in blue), 'Extras', 'Login', 'Register', 'Forgot Password', and 'LOGOUT' (highlighted in blue). The main content area has a light grey header with 'Experiments' and 'Student/Batch/Experiment details'. Below the header, there is a 'Batch/Experiment details' section with three dropdown menus: 'Year' (Select Year), 'Batch' (Select Batch), and 'Experiment Name' (Select Experiment Name). Below these dropdowns are three blue buttons: 'Check Submitted Assignments', 'List of all students', and 'Download Result as PDF'. The footer contains the text 'Copyright © 2020-2021 E-Practical Submissions. All rights reserved.' and 'Version 3.1.0-pre'.

Figure 21 : Teacher Dashboard Page

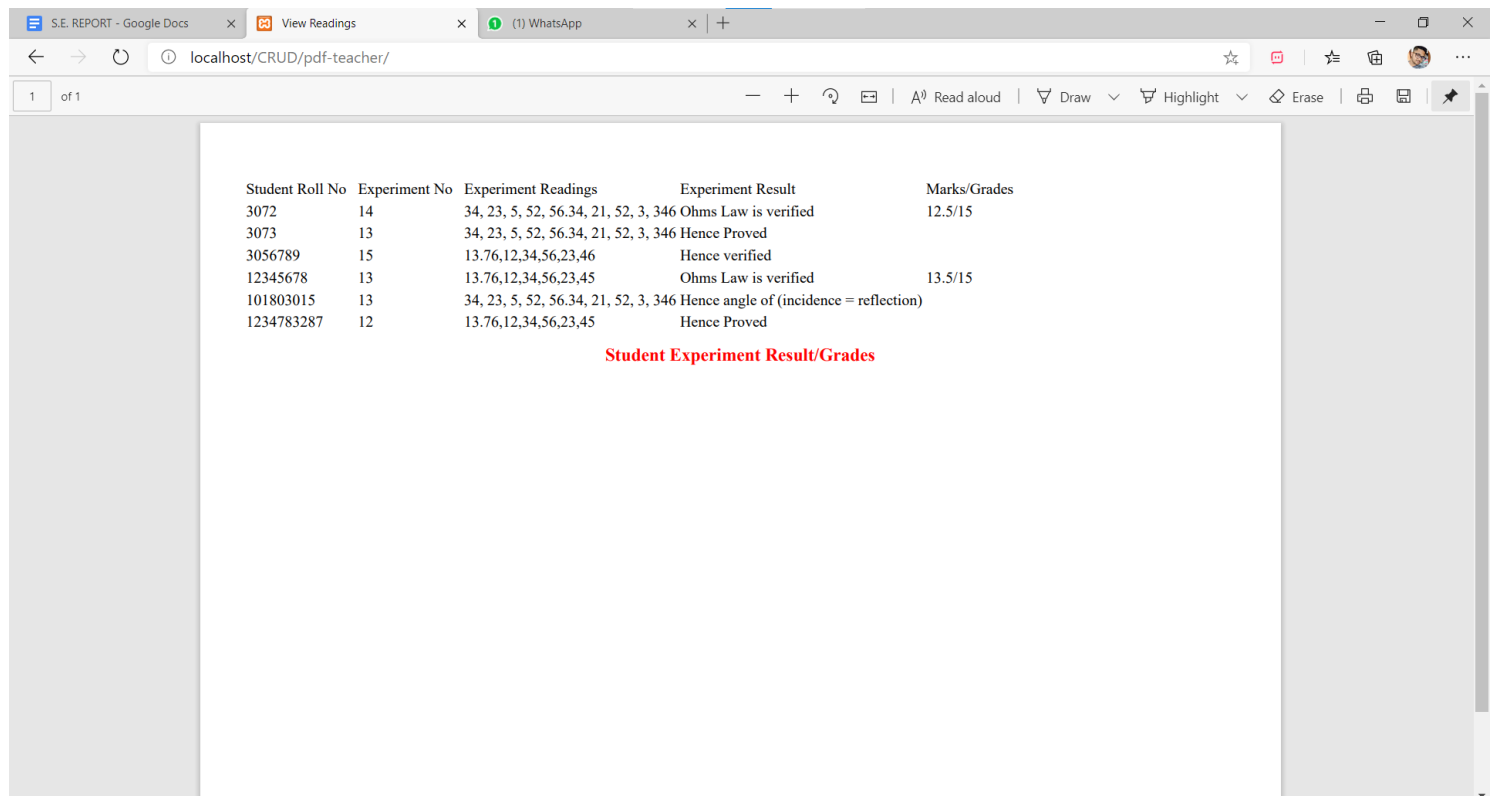


The screenshot shows a web browser window with the address bar displaying 'localhost/CRUD/readings/view.php'. The page title is 'Student Experiment Result/Grades'. Below the title is a table with 7 columns: Student Roll No, Experiment No, Experiment Readings, Experiment Result, Marks/Grades, Edit, and Delete. The table contains 7 rows of data. Each row has a blue 'Edit' link and a blue 'Delete' link in the last two columns. Below the table is a yellow button labeled 'Download as PDF'.

Student Roll No	Experiment No	Experiment Readings	Experiment Result	Marks/Grades	Edit	Delete
3072	14	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Ohms Law is verified	12.5/15	Edit	Delete
3073	13	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Hence Proved		Edit	Delete
3056789	15	13.76,12,34,56,23,46	Hence verified		Edit	Delete
12345678	13	13.76,12,34,56,23,45	Ohms Law is verified	13.5/15	Edit	Delete
101803015	13	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Hence angle of (incidence = reflection)		Edit	Delete
1234783287	12	13.76,12,34,56,23,45	Hence Proved		Edit	Delete

[Download as PDF](#)

Figure 22 : Student Grades Table with Edit Access



The screenshot shows a PDF viewer window with the address bar displaying 'localhost/CRUD/pdf-teacher/'. The PDF content displays the same table as Figure 22, but without the 'Edit' and 'Delete' links. The table is titled 'Student Experiment Result/Grades' in red text.

Student Roll No	Experiment No	Experiment Readings	Experiment Result	Marks/Grades
3072	14	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Ohms Law is verified	12.5/15
3073	13	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Hence Proved	
3056789	15	13.76,12,34,56,23,46	Hence verified	
12345678	13	13.76,12,34,56,23,45	Ohms Law is verified	13.5/15
101803015	13	34, 23, 5, 52, 56.34, 21, 52, 3, 346	Hence angle of (incidence = reflection)	
1234783287	12	13.76,12,34,56,23,45	Hence Proved	

Student Experiment Result/Grades

Figure 23 : Student Result in PDF format

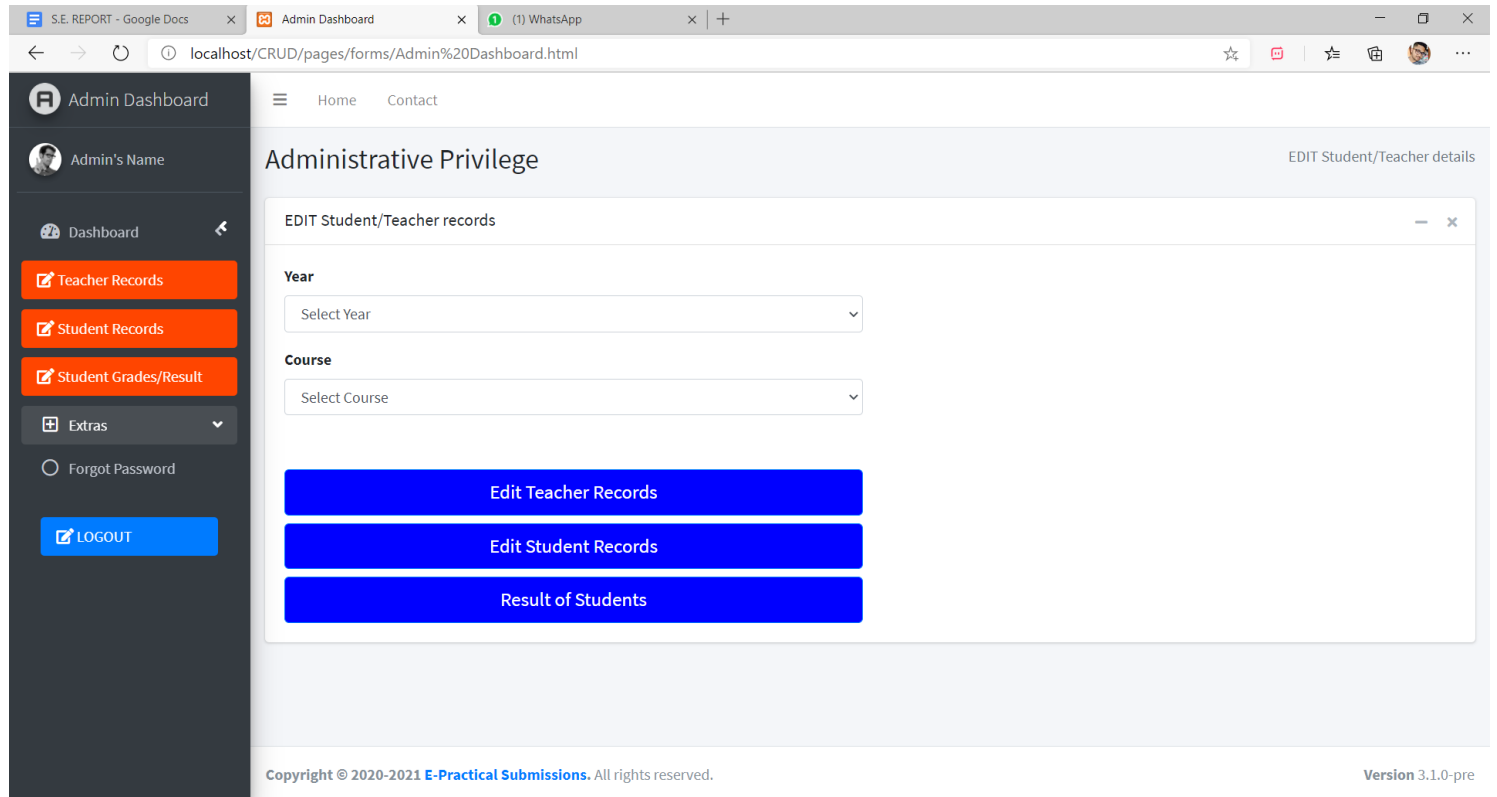


Figure 24 : Admin Dashboard Page

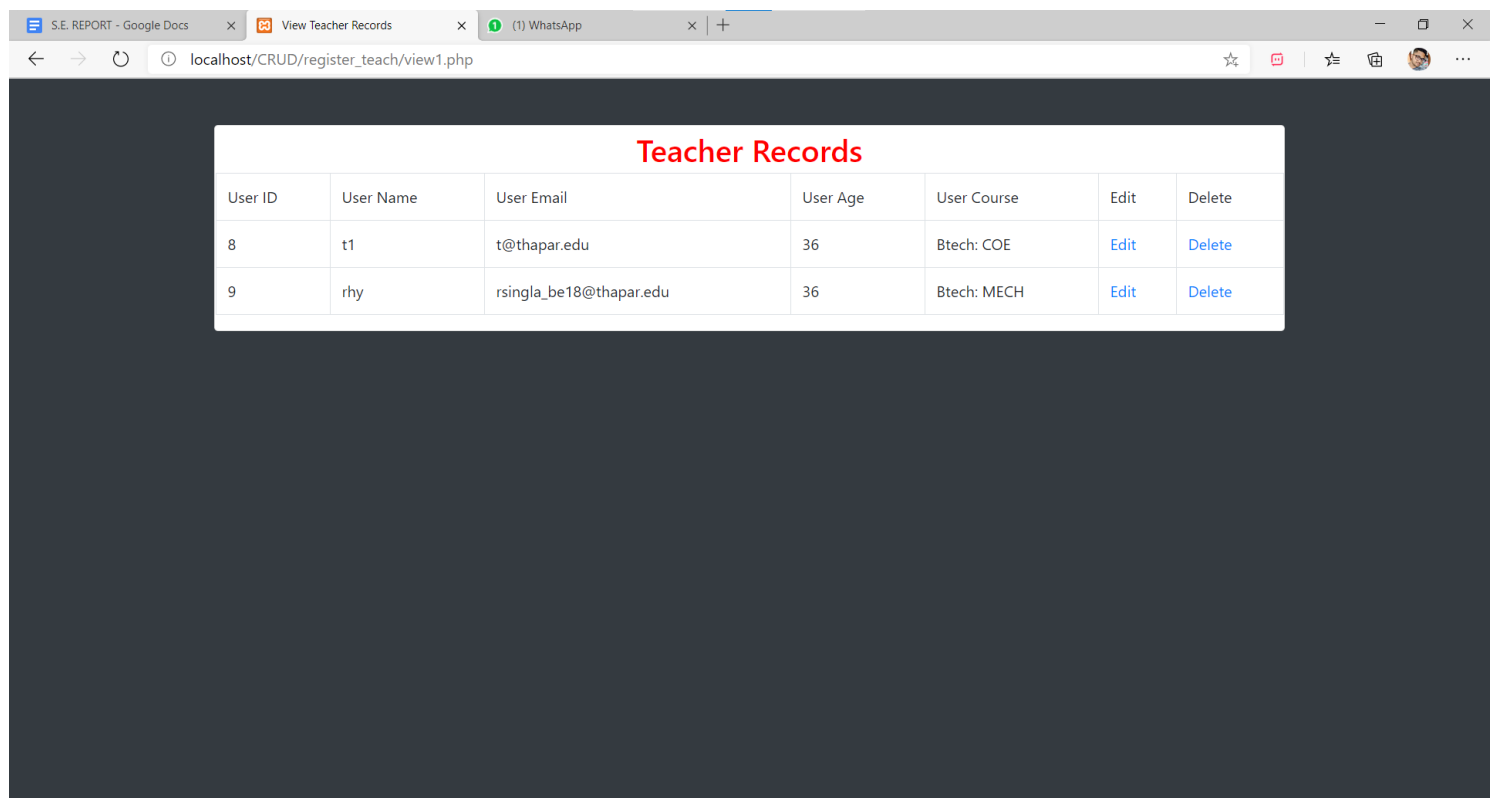
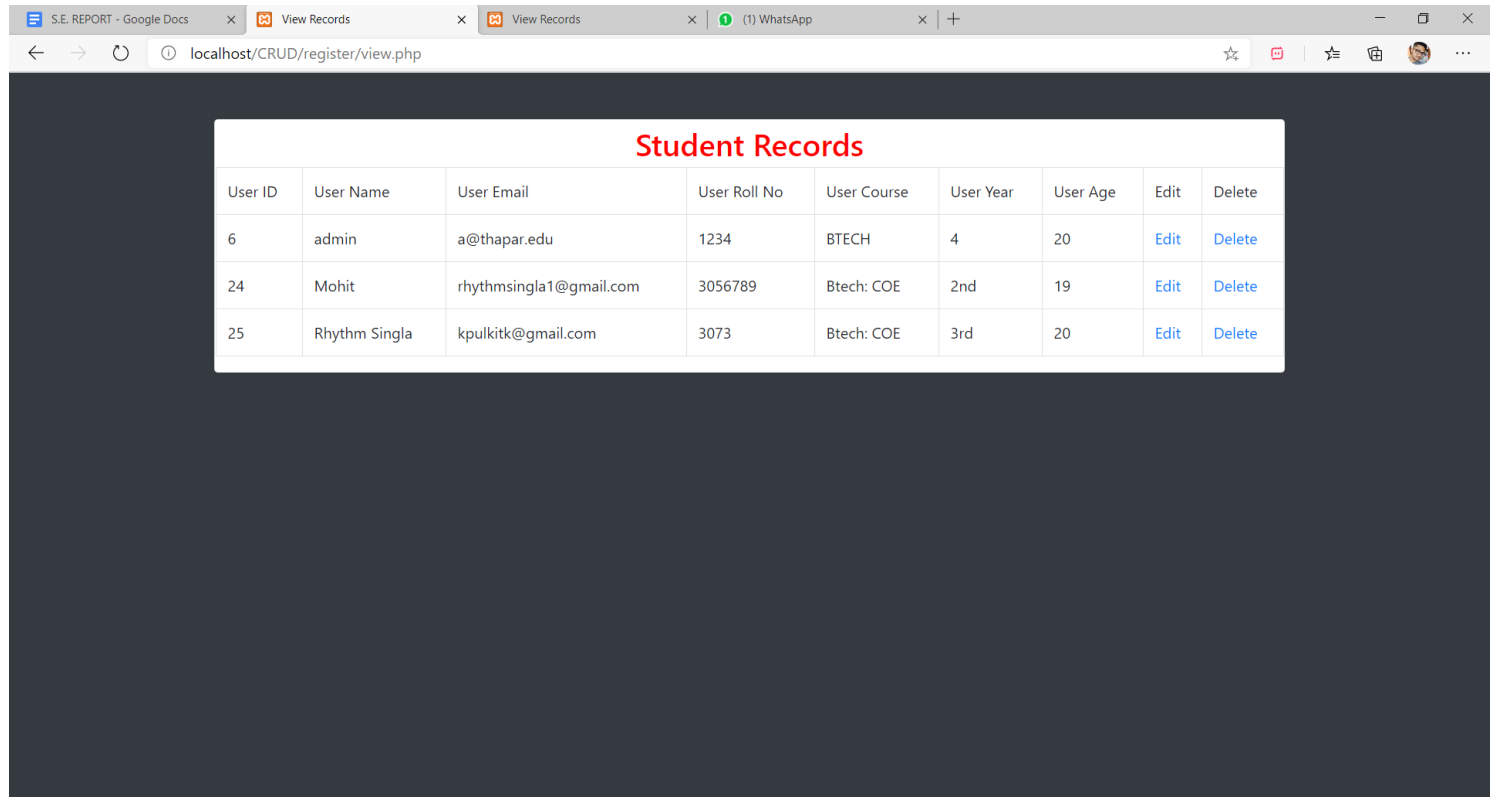


Figure 25 : Teacher Records Table with EDIT Access



The screenshot displays a web application interface for viewing student records. The browser's address bar indicates the URL is `localhost/CRUD/register/view.php`. The page features a table titled "Student Records" with the following data:

User ID	User Name	User Email	User Roll No	User Course	User Year	User Age	Edit	Delete
6	admin	a@thapar.edu	1234	BTECH	4	20	Edit	Delete
24	Mohit	rhythmsingla1@gmail.com	3056789	Btech: COE	2nd	19	Edit	Delete
25	Rhythm Singla	kpulkitk@gmail.com	3073	Btech: COE	3rd	20	Edit	Delete

Figure 26 : Student Records Table with EDIT Access