Student Attendance Taking System

By

Md. Saidul Islam Satil
151-35-891

A project submitted in partial fulfillment of the requirement for the degree of

Bachelor of Science in Software Engineering

Department of Software Engineering
DAFFODIL INTERNATIONAL UNIVERSITY

Fall – 2018

APPROVAL

This **Project** titled "**Student attendance taking system**", submitted by **Md. Saidul Islam Satil**, **151-35-891** to the Department of Software Engineering, Daffodil International University has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc in Software Engineering and approved as to its style and contents.

BOARD OF EXAMINERS

Dr. Touhid Bhuiyan
Professor and Head
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Chairman

K. M. Imtiaz-Ud-Din
Assistant Professor
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Internal Examiner 1

Internal Examiner 2

Md. Shohel Arman
Lecturer
Department of Software Engineering
Faculty of Science and Information Technology
Daffodil International University

Shafiul Hasan

Mr. Shafiul Hasan
Managing Director
Vivacom Solution, Dhaka

External Examiner

Declaration

I hereby declare that, this project has been done under the supervisor of Md. Rashed karim, Adjunct Faculty of Department of Software Engineering, Daffodil International University. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree.

Md. Saidul Islam Satil

Id: 151-35-891
batch: 16th
Department of software Engineering
Faculty of Science & information Technology

Daffodil International University

Certified by:

Md. Rashed Karim

Adjucnet Faculty

Department of Software Engineering

Daffodil International University

Acknowledgement

At first I want to thank Almighty for this beautiful life and for everything.

I also want to thank my parents for supporting me in every struggling moment of my life. I have failed so many times but they never lose faith from me. They always believed that I can. They always inspire me and give value to my opinion. If they were not with me, I could be a fallen star.

I am grateful to all of my teachers from primary school to University. Classrooms, faces of teachers, types and topics of teaching, decoration of classrooms everything have been changed with times. But one thing was unchanged from the classroom of blackboard to digital classroom. They have taught us about politeness, morality. And I am really so much grateful to all my teachers to understand me to be a good human being is more important to be a good engineer.

And I wat to thank my supervisor **Md. Rashed Karim** for giving me the chance to work this project. Without his proper guidelines and support it was so difficult to complete this project. In the time of working with this project I got some obstacles. For overcoming those obstacles, some of my friends were always beside me. I am very lucky to get few friends like them. I want to mention some of their names, Avee Chakraborti, Mrinal Mallik, Sudhin Sarkar Bishal, who always give me support to walk forward and I have learnt so many things from them.

And most importantly I also want to thank three million martyrs who sacrificed their lives for the freedom of this nation. And grateful to the government of Peoples Republic of Bangladesh for such a beautiful environment of education.

Table of Contents

List of Figures:

Chapter-1

Introduction

1. Introduction

1.1 About the system

Attendance system is a web based project. It is designed for taking the regular class attendance of students. This system will be helpful for both the teachers and students. Teachers will be able to take and check attendance and students will be able to submit their attendance very easily.

In our regular system, teachers call attendance and students response if they are attending in class. It wastes a long time from the class. Technologies have made everything easier, now a day's teachers are taking attendances in online. But it also taking a long time. In 40-45 minutes classes around 10 minutes are wasting in taking attendance. This system will reduce the time. Total attendance taking process will be complete within 2 minutes.

There will be three types of actor in this system. Admin, Teachers and Students. Admin will maintain the full system. He/she will generate teachers and students in sections according to batches, will provide courses as their demands.

For Taking attendance a teacher must have to login successfully first. After login successfully he/she will select the course with batch and sections. Then he will get the list of students who are registered for that course of that batch and section. Then he/she will click button for take attendance. After clicking Take attendance, a random string will be generated. Then teacher will say that string number to his/her students.

Then students will be able to submit that string number. But to submit that random string number at first they have to login in their profile successfully. After login successfully they can submit the random string number which is given by their teacher. The system will match the number. If matches, then the attendance of that student for that course will be given successfully.

And the main fact is security so that no one can give proxy attendance. This system will ask students to allow the system let to know his location. And it will show the teacher that how far he is staying from the classroom.

1.2 Purpose

Attendance system will have the following goals:

- Provide user friendly interface.
- Easy to store daily attendance.
- East to take and check attendance for the teachers.
- Easy to submit attendance for the students.

1.3 Scope

Attendance system will provide the following unique functionalities:

- Generate a random string
- Match the string
- Manual attendance entry

1.4 Vision

Vision is the art of see which is invisible to others. No project can build up without visions. The vision of this project is to make the attendance taking system easier. Now a day's teachers don't take present manually, they use student attendance taking portal. But still the process works manually. Because teachers call the roll, then students ensure that he is present in the class then teacher submit his/her attendance. This process waste a lot of times. I hope that this system will reduce the waste of time and the whole process will be organized more automatically.

1.5 Impotence

This system is necessary because every second of class time is valuable. But a lot of time waste to take roll call and check, this system will reduce the waste of time so that teachers and students are both can use each second of class time. Without that this system will reduce proving proxy attendance.

1.6 Proposed solution

In this system the super admin will be able to add teachers and students according to batches, sections and courses. Teachers and students will registration according their occupation. When a teacher will log in in his profile, a random string will be generated and the students have to submit the string within a fixed time.

1.7 Stakeholders

If any person or organization will be affected directly or indirectly by this project, then we can call them stakeholders.

1.7.1 Internal Stakeholders

1. Teachers
2. Students

1.7.2 External Stakeholders

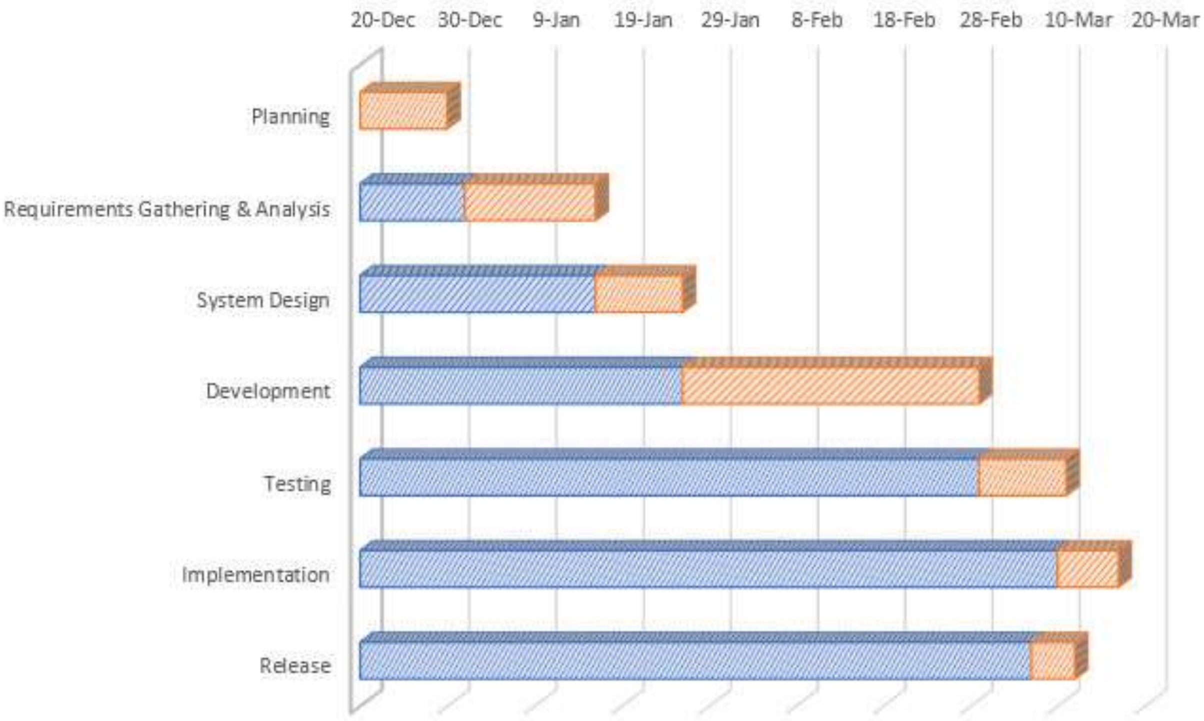
1. Educational Institutes

1.8 Project Scheduling

For developing a project need a proper planning at first. I also made a schedule for developing and executing my project successfully.

1.8.1 Gantt Chart

Stakeholders will get a clear view of this project by seeing the following Gantt chat.



1.8.2 Milestones/Deliverables

Milestone is a timeline of a project. Engineers have to finish a project within a fixed time, That's why they have to work with a fixed time period for the different steps of software development life cycle. The project milestones for this project are given bellow:

Task No	Task Name	Time
01	Requirements gathering and analysis	3 weeks
02	Sketching overall system	1 week
03	Database design	1 week
04	UI design	1 week
05	Implementation	3 weeks
06	Testing	2 weeks
07	Evolution the project	1 week

Chapter-2

Requirement specification

2. Functional Requirement

The whole project has to develop on the basis of the following requirements. Before identification the requirements we needed the comprehensive engagement and lighting quick coordination with the stakeholders. This accelerates the entire requirements management process by orchestrating the follow of information and process across the different team members and stakeholders. Again this is combined with hybrid agile and waterfall methodology tools. Functional requirements are the desired operations of a system. It describes how a system will act with users. How many facilities and functionalities end users will get from a system.

2.1 Registration

Registration is the task where a user will register themselves according to his playing role.

Requirement No	Requirements
01	Register with user name, password, user type and email.
02	Password should be given twice to match.
03	Password must be minimum 6 characters.
04	Approve a user’s information.
05	Email id of two students can’t be same.

2.2 Login & Retrieve Password

Requirement No	Requirements
01	While login match the username with user type
02	User will get functionalities of his/her type
03	Login time should be stored in the log file
04	While retrieve the password(if forgot) user should provide the user name and email id.
05	1. System will mail a link that user containing password.
06	With the new password link user will give his/her favorable password.
07	Update the new reset password with the log file.

2.3 Course Management

Requirement No	Requirements
01	Admin will define year.
02	Admin will define batch.
03	Admin will define semester.
04	Admin will define course.

05	Course will be attached to a semester, year, batch and program.
06	Students are enol to a course.
07	One students may enroll to more than one course.
08	Courses have to option to be taken in different semester.
09	A course must have a unique course code.
10	Admin will assign one or more teachers to a course
11	Admin will assign students to the sections containing that course and batch as their registration.

2.4 Attendance Management

Requirement No	Requirements
01	Teachers will generate a random string number for a particular course and a particular section.
02	Student will submit the same string code.
03	Have to submit the code within a fixed time.
04	System will match the submitted strings with the generate strings.
05	If the strings match, attendance of that student for that course for that day will be submitted.
06	Teachers will be able to see the distance of the students from classroom.

2.5 Attendance record

Requirement No	Requirements
01	Student can see only his attendance to a particular course
02	Students can see total attendance of a particular semester
03	Teachers can see student’s attendance to his/her course
04	Teacher can see the total attendance of all students in his/her course

2.6 External Interface Requirements

2.6.1 User Interface

User interface is very much important for a software because maximum time end users are non IT people. So engineers should make a beautiful user interface so that any types of user can use the software smoothly. In this student management system any backgrounded can be the users of this system. So, I have tried to make a beautiful and simple user interface for this project. In this system there will be different portal for each types of users. The admin has a lot of functionalities to done. He will be able to assign students and teachers in sections according to course, batch and section. So, he will get add teacher, add students, add sections, add batch, add year etc options. A teacher can choose a course with section and batch. So, he will get a dropdown list to select course with batch and section. Then, if any student doesn't have smartphone then a teacher will be able to input the attendance of that student manually. So, he will get a manually give attendance option. Student will simply submit the code generated by the course teacher.

2.6.2 Hardware Interface

This is the current version of the software; it will have no special hardware interface with other external system. It will run in a general-purpose computer system with general-purpose hardware and software.

2.6.3 Software Interface

To the end-user there is no need of any extra software to be installed.

2.6.4 Communication Interface

All types of communication will be secured.

2.7 Feasibility Studies

In this document we also provide some feasibility which will support the system and also give more litheness. For these we gave emphasize on the following topics.

2.7.1 Economical Feasibility

Economic analysis is most frequently used for evaluation of the effectiveness of the system. Most commonly known as cost/benefit analysis the procedure is to determine the benefit and saving that expected from a system and compare them with cost, decisions is made design and implement the system.

This part of feasibility study gives economic justification to the top management. A simple economic analysis that gives the actual comparison of cost and benefits is more meaningful in such cases.

In the system, the authority is most satisfied by the economic study. Because it need not require any additional hardware resources as well as it will save a lot of valuable class time.

2.7.2 Technical Feasibility

Technical feasibility centers on the existing manual system of the test management process and to what extent it can support the system. According to feasibility analysis procedure the technical feasibility of the system is analyzed and technical requirements such as software facilities, procedure, inputs are identified. It is one of the most important phases of the system development activities. The system offers greater levels of user friendliness combined with greater processing speed. Therefore, the cost of maintenance can be reduced.

2.7.3 Behavioral Feasibility

People are inherently resistant to change and computer has been to known to facilitate changes. As estimate should be made of how strong the user is likely to move towards the development of computerized system. These are various levels of users in order to ensure proper authentication and authorization and security of sensitive data of the system.

Chapter-3

System Analysis

3. System Analysis

3.1 Actor goal list

3.1.1 Assumption

In this attendance system, it works student's attendance. It holds a collection of student's attendance, their absence.

3.2 Use Case Model

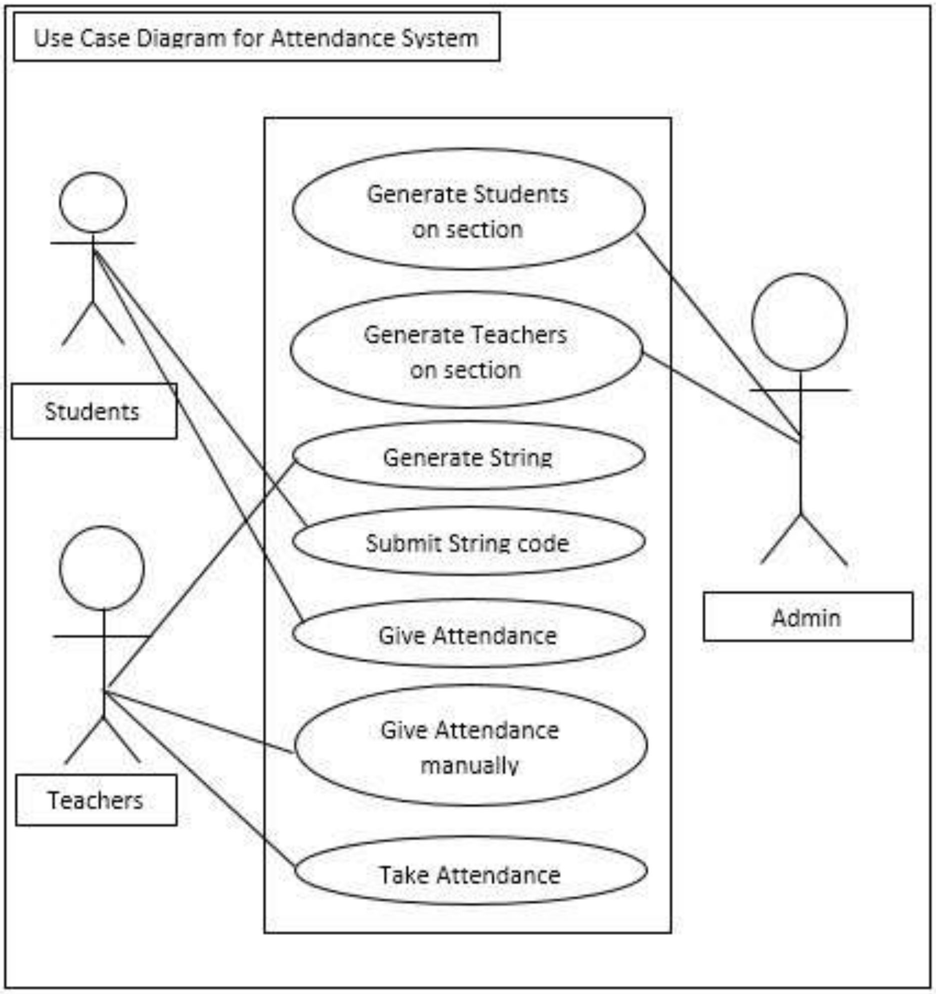


Figure 1: Use case diagram for attendance system.

3.3 Use Case Description (Brief)

3.3.1 Generate Students on section

In this system a super admin will add students in this system according to section. Students need to register every semester and they can register themselves at any section as their wish. Even they can take different subjects at different sections.

But someone need to assign them at a section. In this system Admin will do this job.

3.3.2 Generate Teachers on section

In every semester teacher select courses and sections. Admin will assign teachers in the section they registered.

3.3.3 Generate String code

This system is very smart and in this system teachers don’t need to take attendance by calling roll number. Teachers will just generate a random string and for generating random string teachers don’t need to do anything. The string will generate automatically.

3.3.4 Submit String code

In this System Students don’t need to wait when teacher will call their roll number. They will just submit the string which is generated by teacher within a fixed duration of time.

3.4 Use Case Description

3.4.1 Generate Teachers and Students in sections

Table 1: Description of Use case 1: Generate Teachers and Students in Section

<table>
<tr><td>Scope</td><td colspan="2">Generate Teachers & students in classes</td></tr>
<tr><td>Level</td><td colspan="2">User &amp; Admin</td></tr>
<tr><td>Primary Actor</td><td colspan="2">User &amp; Admin</td></tr>
<tr><td>Precondition</td><td colspan="2">Users &amp; Admin must have to registration and login first</td></tr>
<tr><td>Post Condition</td><td colspan="2">Users generate and submit strings and Admin generate them into sections.</td></tr>
<tr><td rowspan="2">Main flow</td><td>Actor</td><td>System</td></tr>
<tr><td>1. Get Mail Id and Password after registration.
2. Log in into dashboard. Mail id define as a teacher or as a student.</td><td>1. Record Mail Id and Password successfully.
2. Login into System successfully. Show dashboard based on registration.</td></tr>
<tr><td>Alternative scenario</td><td>Actor</td><td>System</td></tr>
</table>

	1. Login without registration or as different actor.	1. System returns Login unsuccessful.

3.4.2 Attendance Schedule

Table 2: Description of Use case 2: Attendance Schedule

Scope	Create attendance schedule	
Level	Teacher & Admin level	
Primary Actor	Teachers	
Precondition	Teachers make attendance schedule	
Post Condition	Teachers must be login first	
Main flow	Actor	System
	1. Get attendance schedule after registration records in device. Schedule(date, time, day)	1. System records attendance schedule successfully.
Alternative scenario	Actor	System
	1. Don’t get attendance schedule if registration records are not available in device.	1. System fails to record attendance schedule successfully.

3.4.3 Generate String code

Table 3: Description of Use case 3: Generate String

Scope	Generate String	
Level	Teachers and Admin	
Primary Actor	Teachers	
Precondition	Teachers submit their mail id & password	
Post Condition	Teachers must be login	
Main follow	Actor	System
	1. To generate the random string have to login successfully and have to select course, section & batch.	1. System will generate the random string.
Alternative scenario	Actor	System
	1. To generate the random string without login successfully or not to select course, section & batch.	1. System will fail to generate the string and process will be unsuccessful.

3.4.4 Submit String

Table 4: Description of Use case 3: Submit String

Scope	Submit String	
Level	Students & Admin	
Primary Actor	Students	
Precondition	Students submit their mail id & password	
Post Condition	Students must be login	
Main follow	Actor	System
	2. To submit the random string have	2. System will match the string and

	to login successfully and have to select course, section & batch.	will record attendance successfully.
Alternative scenario	Actor	System
	2. To submit the random string without login successfully or not to select course, section & batch.	2. System will fail to match the string and record attendance will be unsuccessful.

3.4.5 Give Attendance Manually

Table 5: Description of Use case 4: Give Attendance Manually

Scope	Manual attendance entry	
Level	Teachers & Admin level	
Primary Actor	Teachers	
Precondition	Teachers entry student's attendance	
Post Condition	Teaches must be login	
Main flow	Actor	System
	1. Get manual attendance entry after login first. Attendance (student's course, section, batch, date, time)	1. System record manual attendance entry successfully.
Alternative scenario	Actor	System
	1. Get manual attendance entry after	1. System fails to record manual

	login first. Attendance (student’s course, section, batch, date, time	attendance entry successfully.

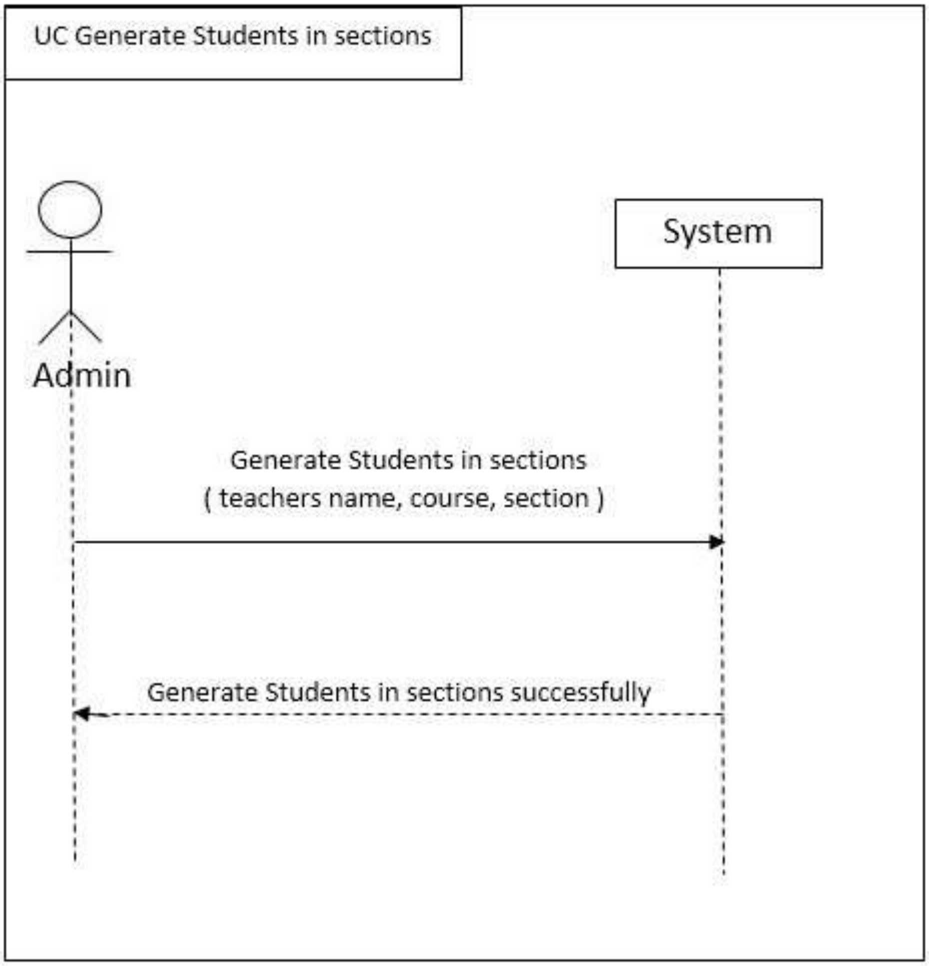
3.4.6 Take Attendance

Table 6: Description of Use case 5: Take Attendance

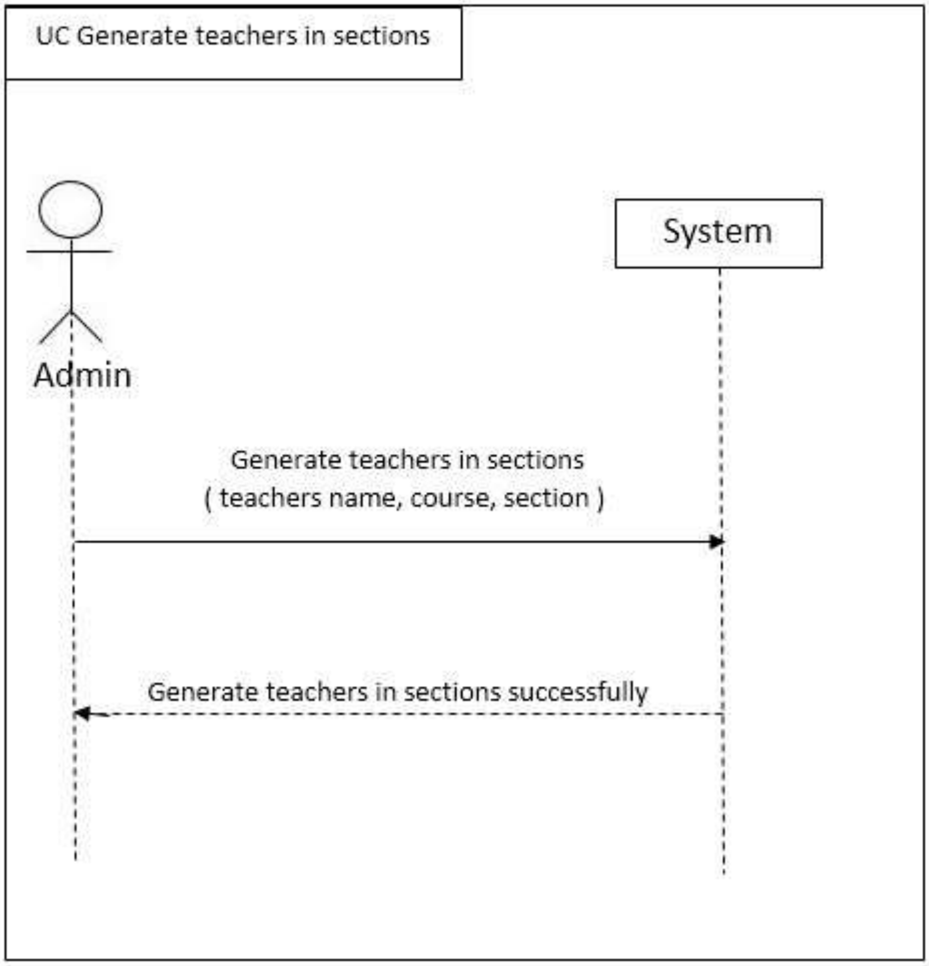
Scope	Take Attendance	
Level	Teachers & Admin level	
Primary Actor	Teachers	
Precondition	Teachers check student’s attendance	
Post Condition	Teaches must be login	
Main flow	Actor	System
	1. Check attendance after login first. Attendance (student’s course, section, batch, date, time)	1. System records attendance successfully.
Alternative scenario	Actor	System
	1. Check attendance after login first. Attendance (student’s course, section, batch, date, time)	1. System fails to record attendance successfully.

3.5 System Sequence Diagram

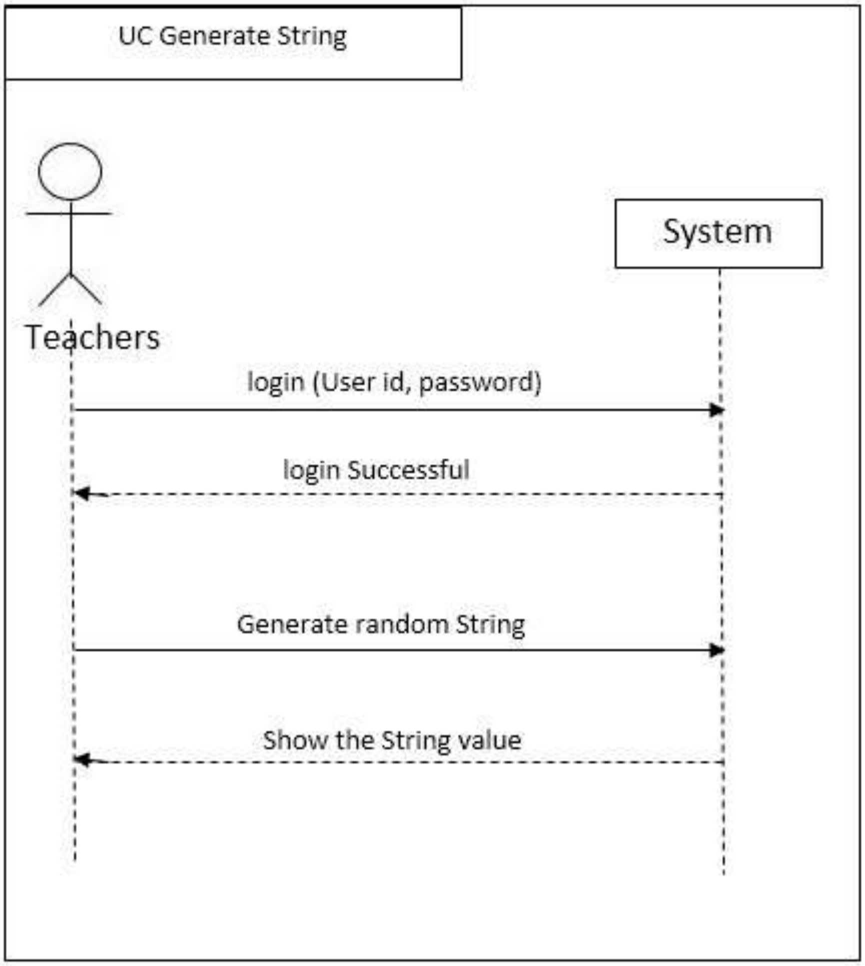
3.5.1 Generate Students in section


UC Generate Students in sections
System
Admin
Generate Students in sections
(teachers name, course, section)
Generate Students in sections successfully

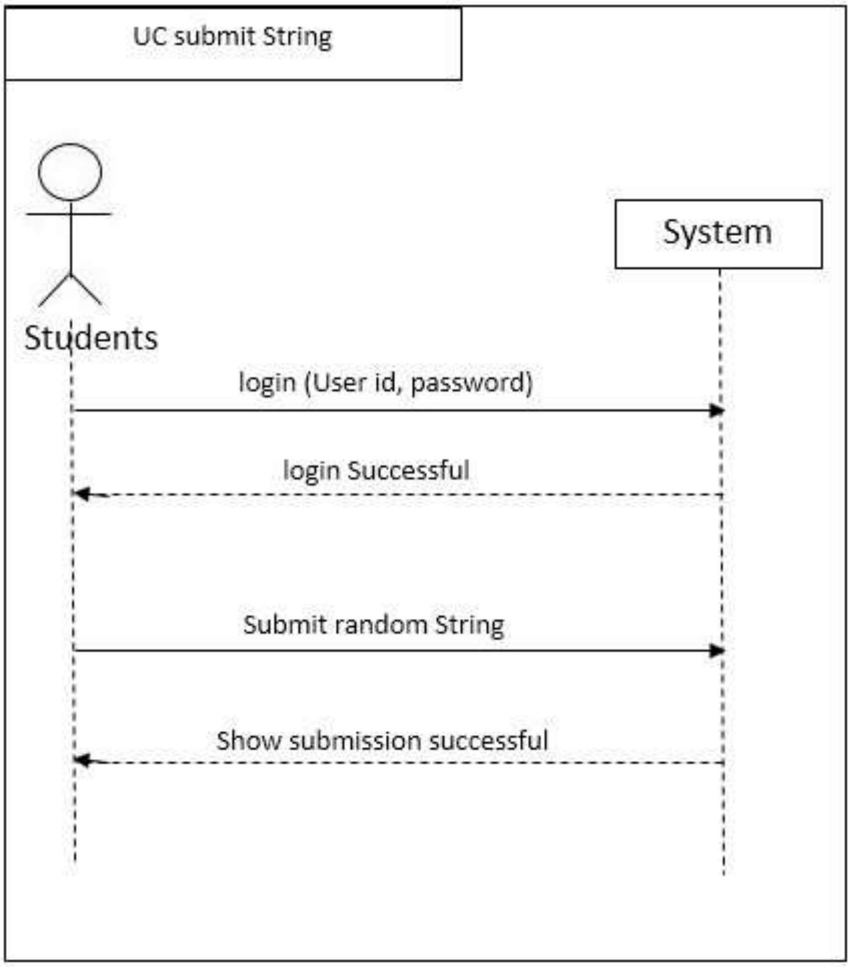
3.5.2 Generate Teachers in section


UC Generate teachers in sections
System
Admin
Generate teachers in sections
(teachers name, course, section)
Generate teachers in sections successfully

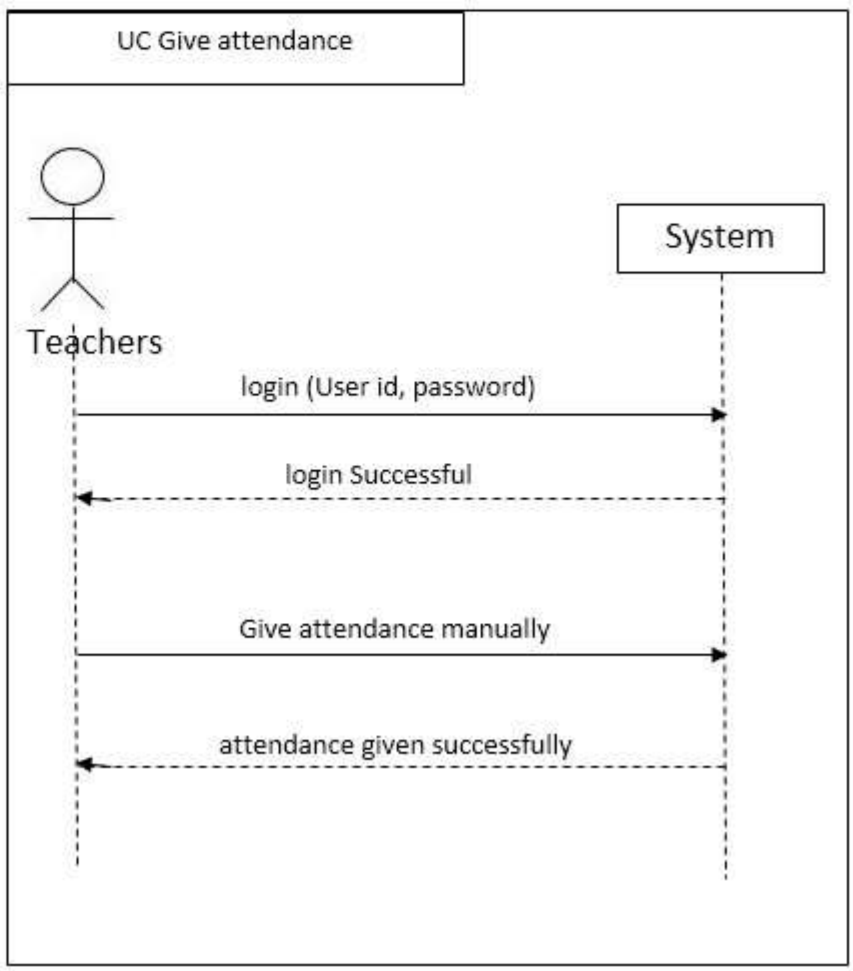
3.5.3 Generate String



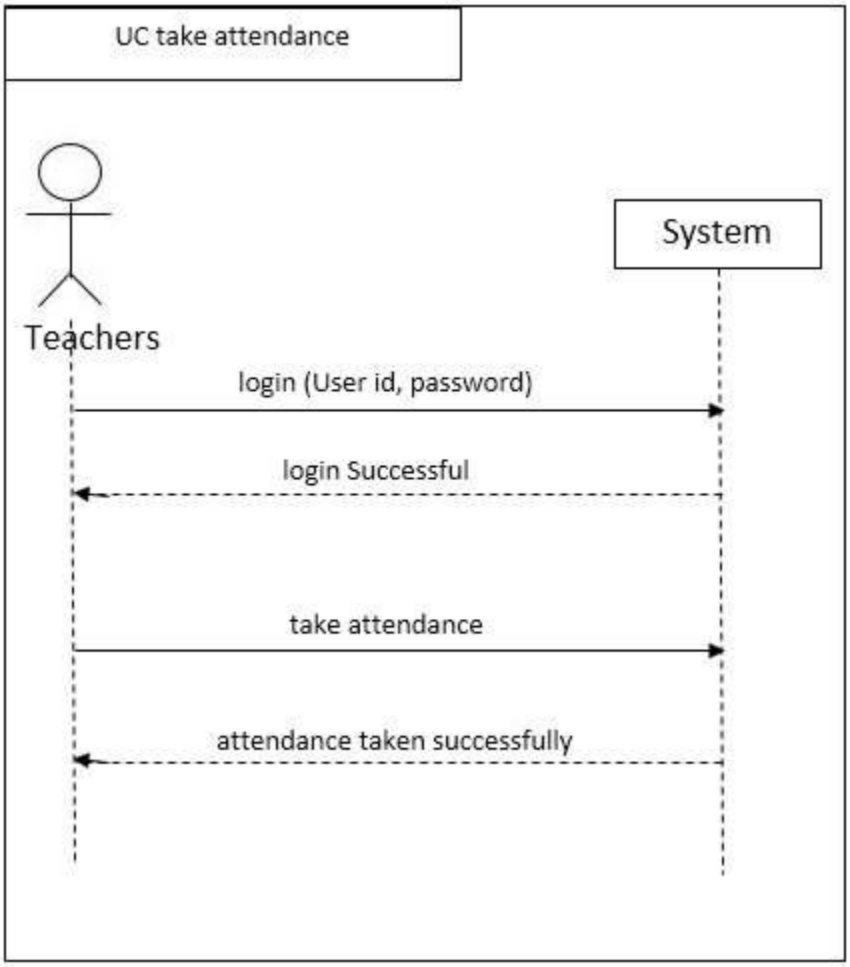
3.5.4 Submit String



3.5.5 Give attendance manually

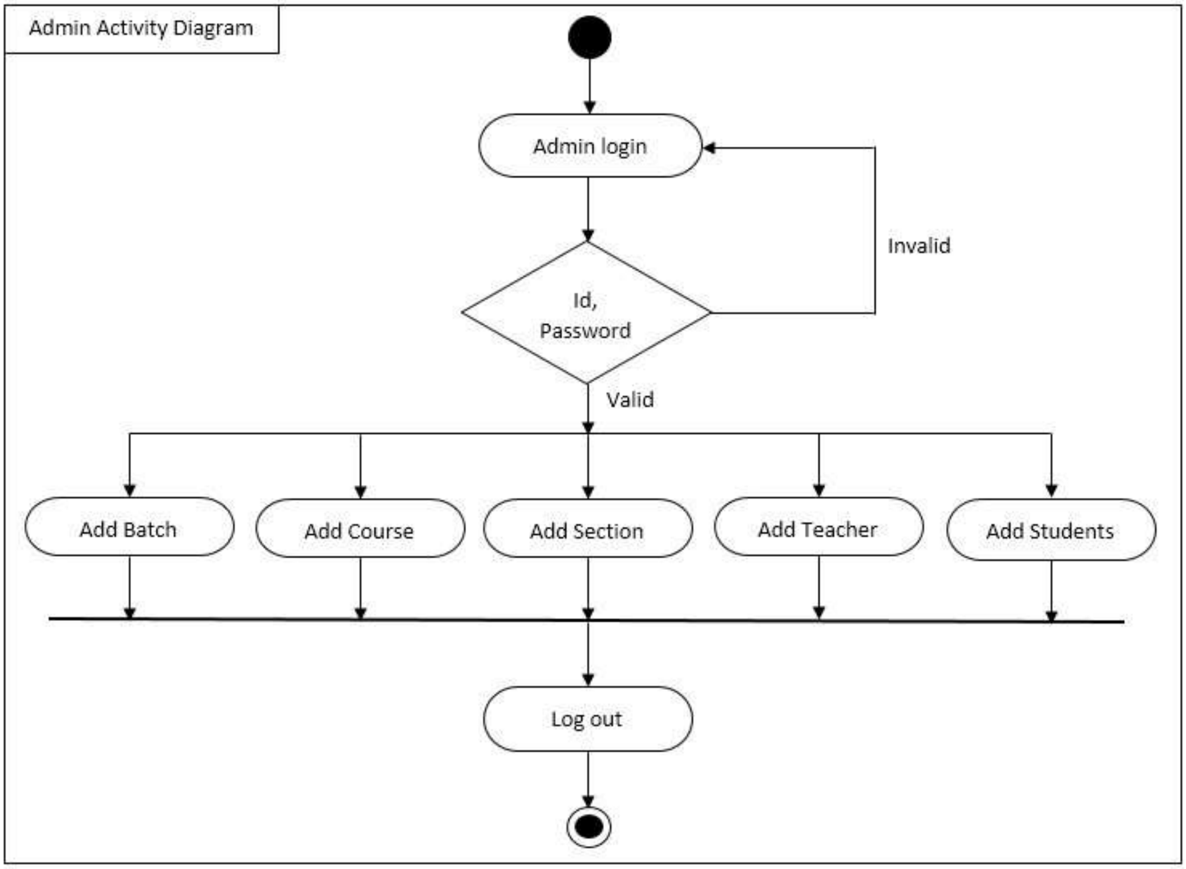


3.5.6 Take attendance

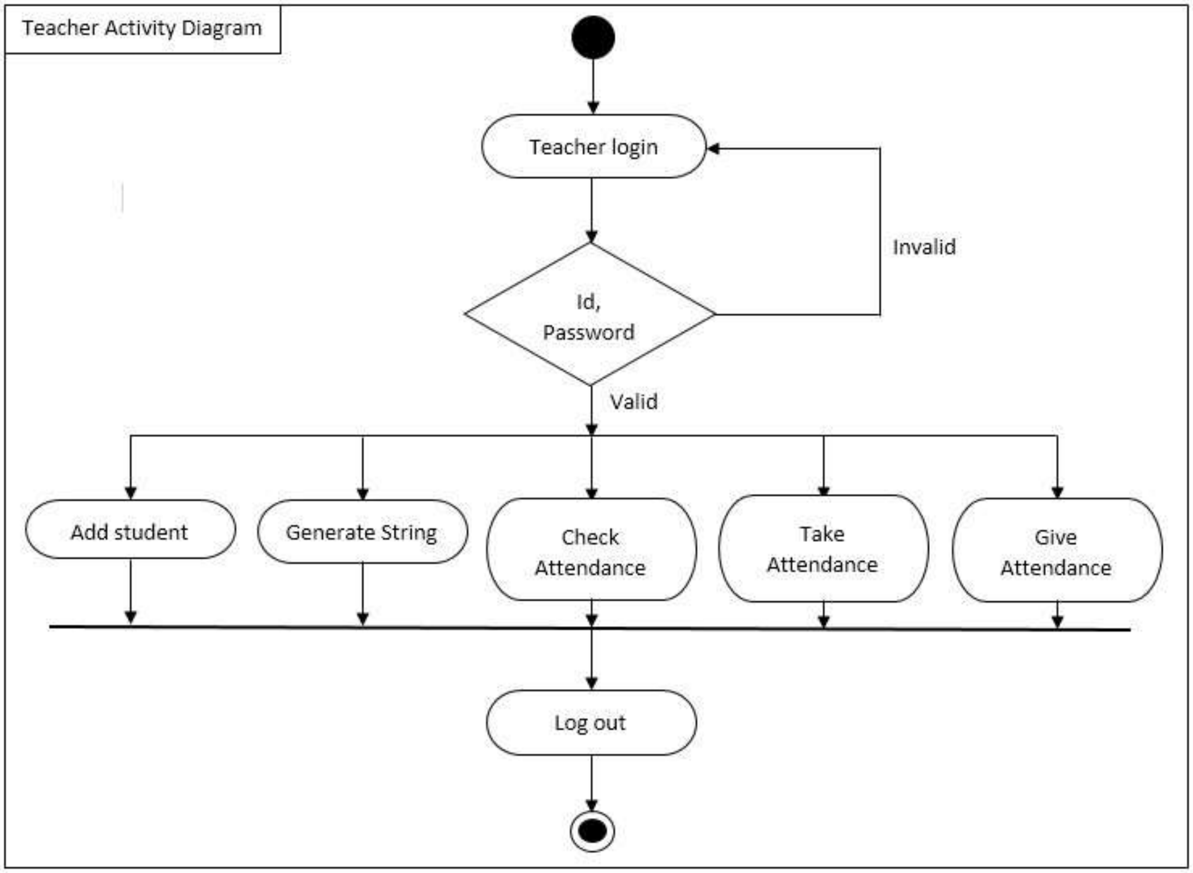


3.6 Activity Diagram

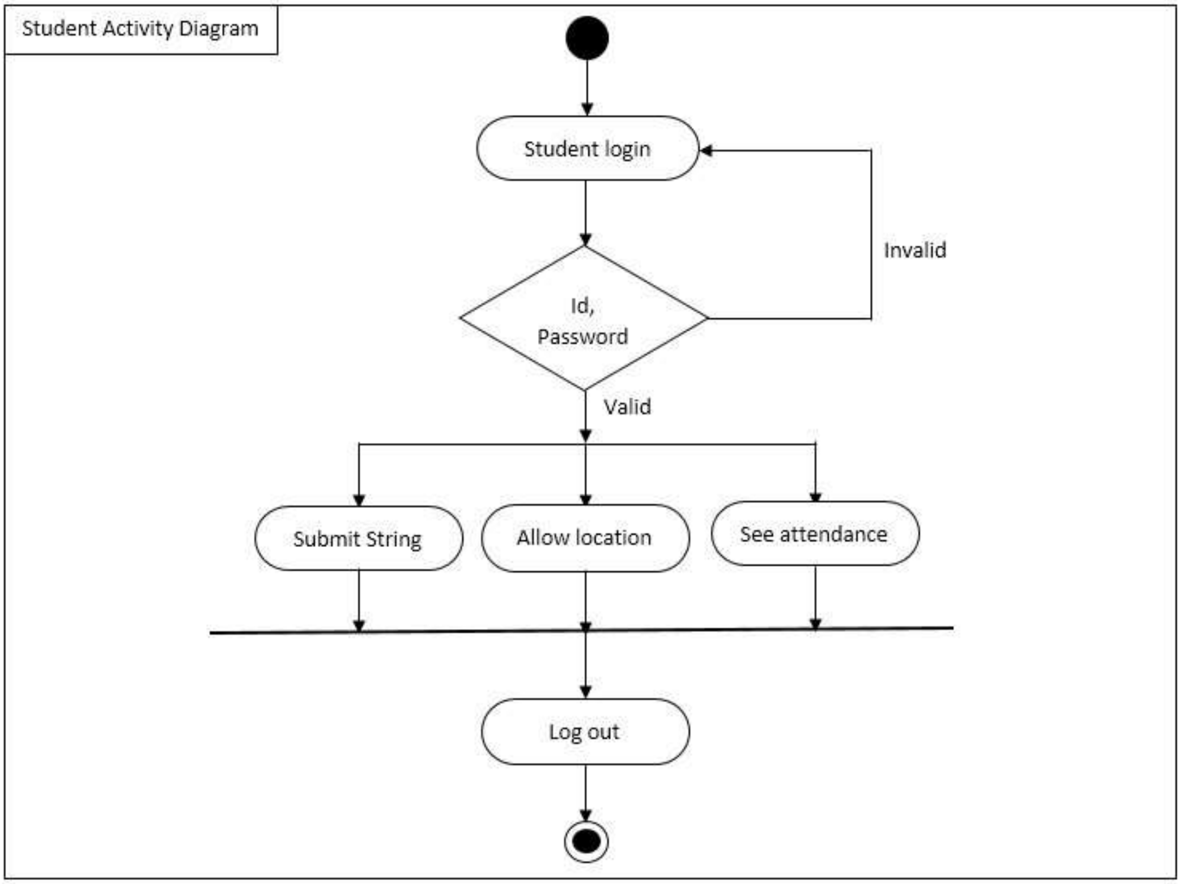
3.6.1 Admin Activity Diagram



3.6.2 Teachers Activity Diagram

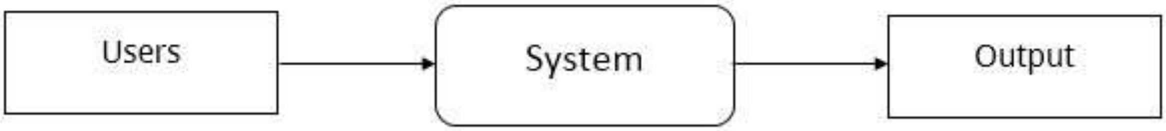


3.6.3 Student Activity Diagram

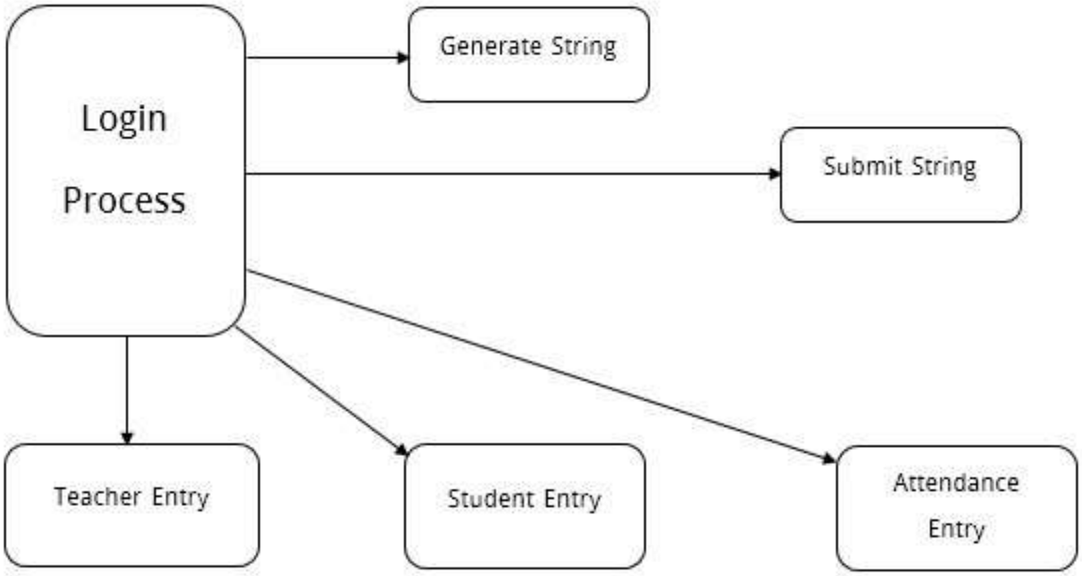


3.7 Data Flow Diagram (DFD)

3.7.1 DFD Level 0

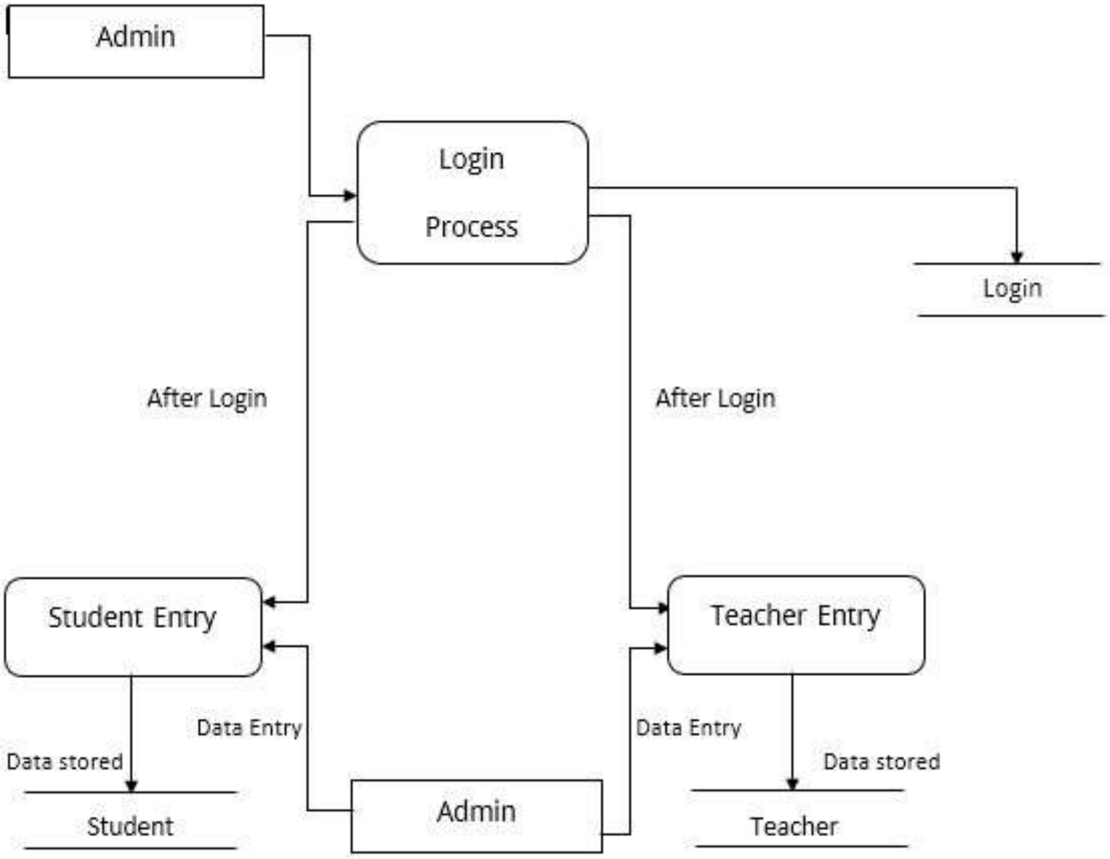


3.7.2 DFD Level 1

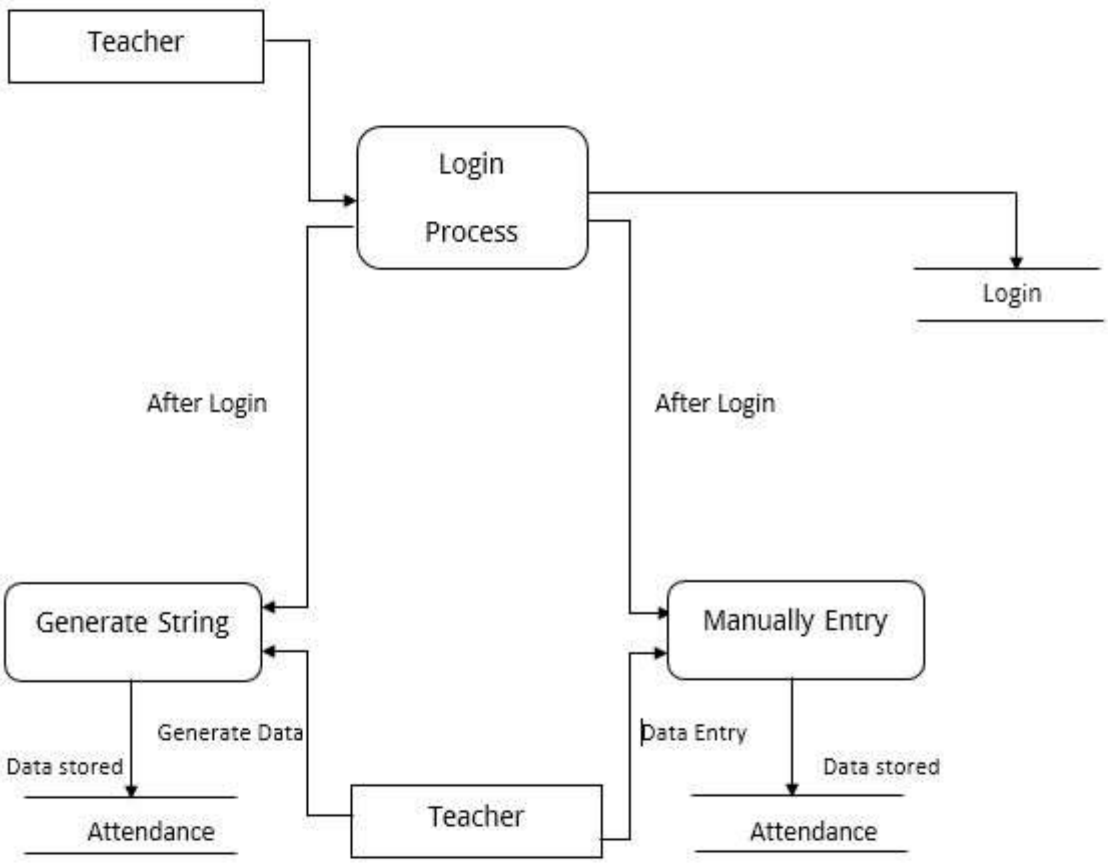


3.7.3 DFD Level 2

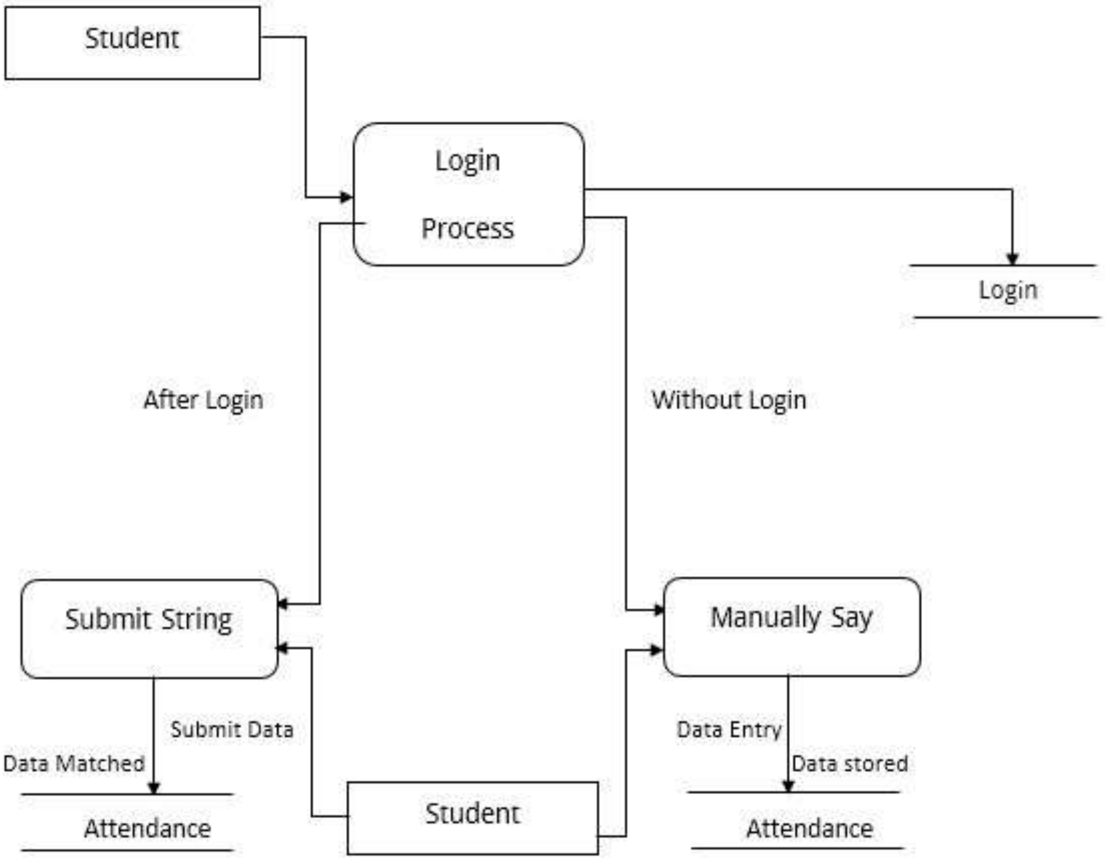
- **2.1**



- **2.2**


Teacher
Login
Process
Login
After Login
After Login
Generate String
Manually Entry
Generate Data
Data Entry
Data stored
Data stored
Attendance
Teacher
Attendance

- **2.3**


Student
Login
Process
Login
After Login
Without Login
Submit String
Manually Say
Submit Data
Data Matched
Data Entry
Data stored
Attendance
Student
Attendance

Chapter 4

Design & Development

4. Design and development

4.1 Application Interface Design

Website used to be the only offspring of the world wide web, but not they give birth to more applications. This new generation does not have the platform constraints or instillation requirements, of the former, making web apps quite an attractive option for business, web application interface design is a form of web design, but the focus is on the function. These applications offer simple, intuitive interfaces that are user-responsive, which allows users to get things done with less time and effort. To achieve this, below are some points to consider and include in your application interface design.

4.1.1 Multiple Features

My project is complete solution handle of student's attendance case. It helps both the teachers and students to take and submit daily attendance smoothly. It will also help to keep daily attendance reports perfectly. Teachers can easily trace the location of a student also. So students will not provide proxy.

4.2 Design & Implementation Constraints

Realization of a concept or idea into a configuration, drawing, model, module, pattern, plan or specification and which helps achieve the item's designated objectives.

4.2.1 Software Language of framework

For fond-end I use HTML, Bootstrap, JavaScript. My design looks like premium template.

And for back-end I use PHP and a PHP framework name Laravel.

- I Used Laravel framework for my project.
- It's a server site language.
- It follows MVC pattern.

4.2.2 Information management

It manages all kinds of information. Information's are stored in database server.

4.2.3 Data Entry

For data entry purposes I used some tools. Tools are given bellow"

- Microsoft Office
- Draw.io
- Mockplus

4.3 User Interface Technology

My project is made in Laravel framework using HTML, CSS, Bootstrap and JavaScript and all data are stored in the database.

4.3.1 Implementation Tools & Platforms

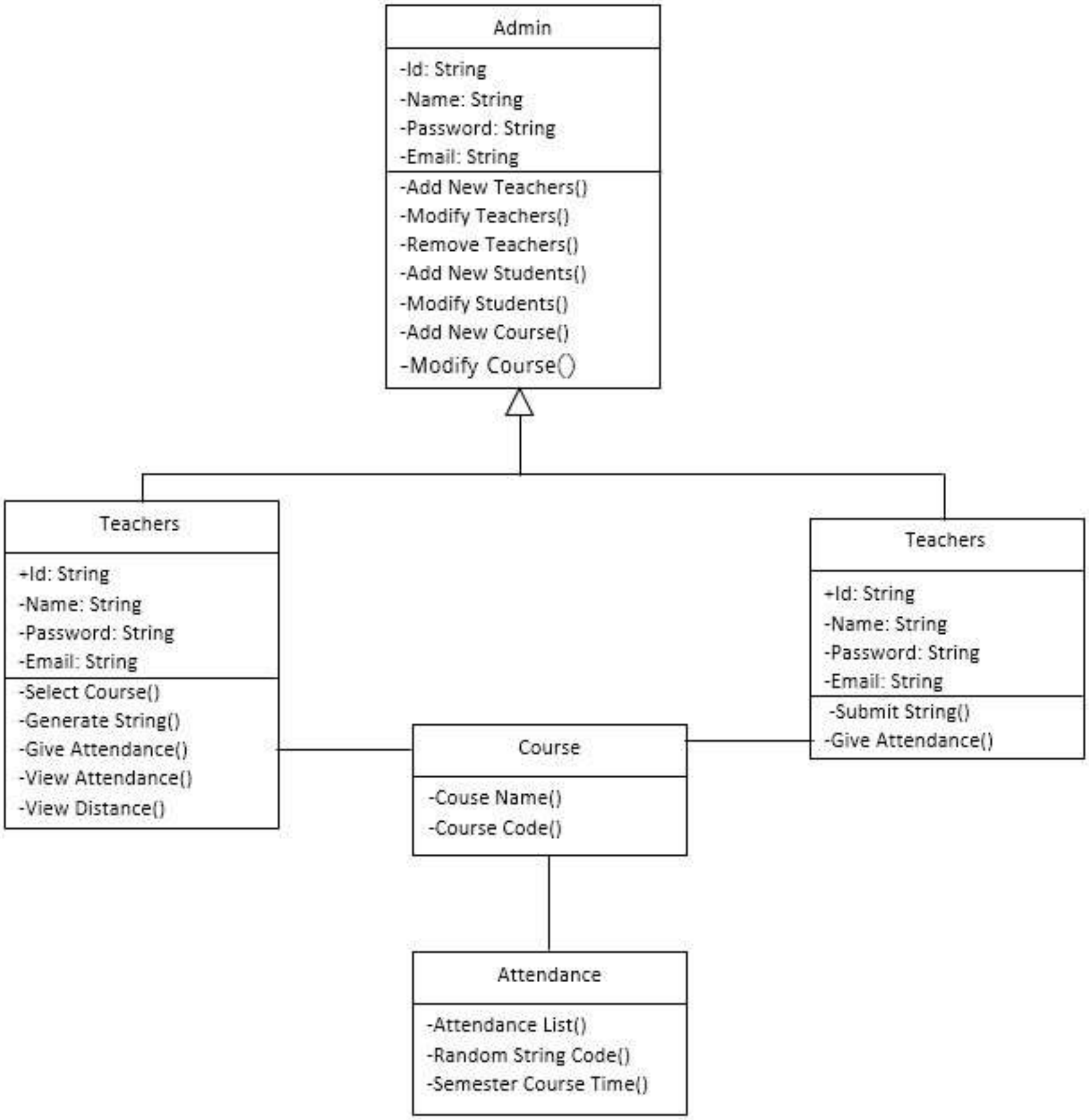
Recommended server configuration

- PHP support (Laravel)
- XAMPP (v3.2.2 or later)
- MySQL
- Apache

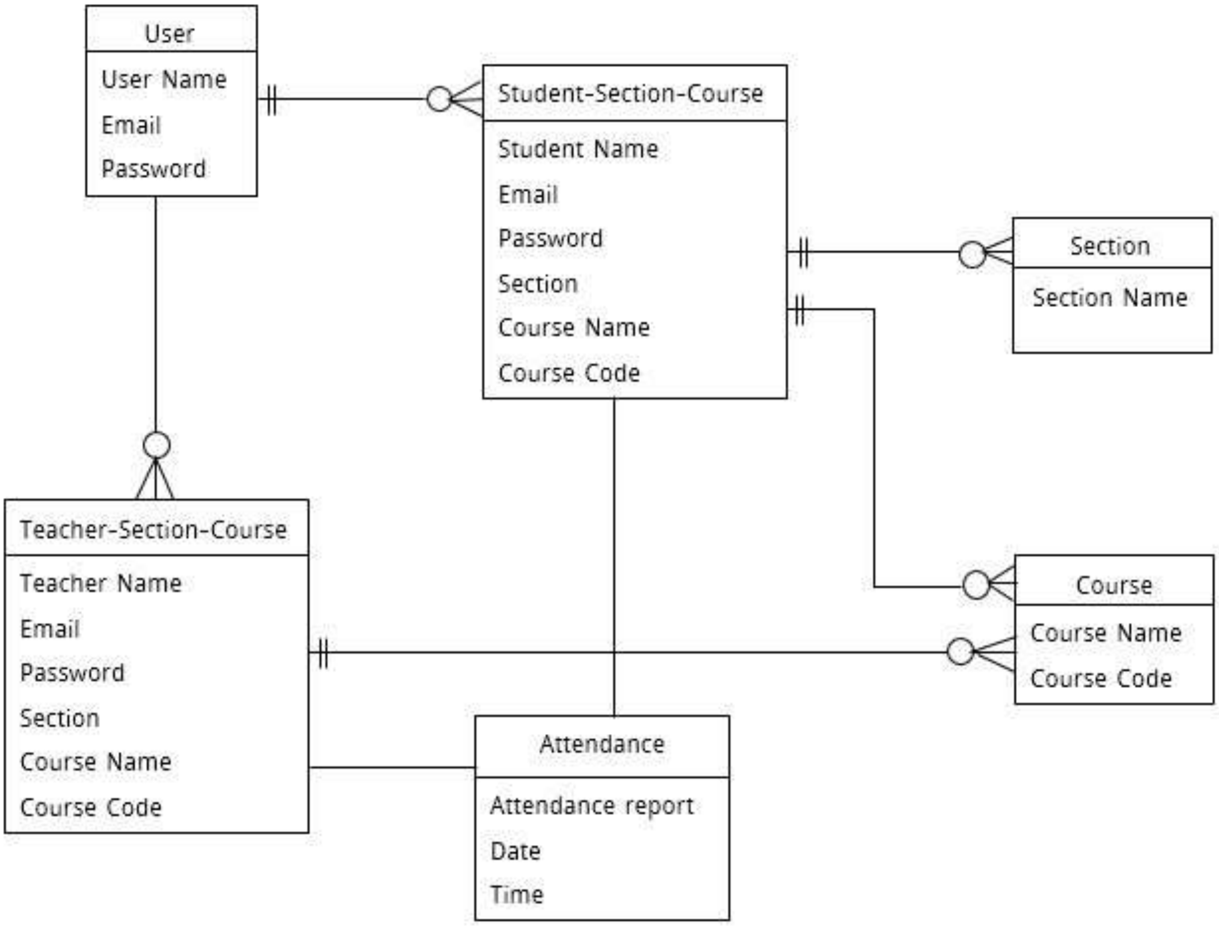
Other versions or platforms may work but they may have been tested. I welcome feedback from users who have successfully run the system in different platforms.

4.4 Proposed Architecture

4.4.1 Class Diagram



4.4.2 Database Diagram


User
User Name
Email
Password
Student-Section-Course
Student Name
Email
Password
Section
Course Name
Course Code
Section
Section Name
Teacher-Section-Course
Teacher Name
Email
Password
Section
Course Name
Course Code
Course
Course Name
Course Code
Attendance
Attendance report
Date
Time

Chapter-5

Test Plan

5. Test Plan

5.1 Testing Features

Features testing is the process of making change in software system to add one or more new features to make modifications in the already existing features. Each of these feature is said to have a characteristic that is designed to be useful, intuitive and effective.

5.1.1 Features to be Tested

Features	Priority	Description
Login	1	Login as authentic user.
Generate String code	2	When teacher will generate the string code.
Submit String code	2	Students will submit the same string
Match String code	3	Matching the generating and submitting string
Show distance	2	Showing the distance of student from classroom
Database	2	Database is much important to store information.

Here, 1= Low priority, 2= Medium Priority, 3= High priority

5.2 Testing Strategy

A testing strategy is a general approach to the testing process rather than a method of devising particular system or component tests. Different testing strategies may be adopted depending on the type of system to be tested and the development process used.

5.2.1 Test approach

A test approach is the test strategy implementation of a project, defines how testing would be carried out. Test approaches have two techniques.

- Proactive- An approach in which the test design process in initiated as early as possible to find and fix the defects before build it.
- Reactive- An approach in which the testing is not started until after design and coding are completed.

5.2.1.1 Black box testing

Black box testing is also called functional testing that ignores the internal mechanism of a system or component and focus on the outputs generated in response to selected inputs and execution conditions. We have decided to perform equivalence partitioning and boundary value analysis for this system.

5.2.1.2 Equivalence class partitioning

In considering the inputs for our equivalence testing, the following types will be used.

- Legal Input values: Test values within boundaries of the specification equivalence classes. This will be input data the program excepts and is programmed to transformed into usable values.
- Illegal Input Values: Test equivalence classes outside the boundaries of the specification. This will be input data the program may be presented, but that will produce any meaningful output.

5.2.1.3 Boundary value analysis

The acceptable range of the values for the application was set at the development time. At the time of testing developer will define the boundary value and generate test case for performing the boundary value analysis.

5.2.1.4 White box testing

White box testing is a software testing method in which internal structures and functionality testing of a system. Basically a programmer work as white box tester.

5.2.1.5 Pass/Fail Criteria

The criteria for selecting pass or fail a step is given bellow.

- According to the given scenario the expected result need to take place then the scenario will be selected as pass otherwise that criteria will be failed.
- System should be hundred present pass. If an item tested 10 times, 9 times perfectly worked and single time don't work properly then it will be considered as fail case.
- System crash will be considered as fail case.
- After submitting a query in the system, if expected page won't appear then it will be considered as fail case.

5.3 Testing Schedule

Test Phase	Time
Test plan creation	1 week
Test specification creation	2 weeks
Unit testing	5 weeks
Component testing	1 week
Test Phase	**Time**
Integration testing	1 week
Use case diagram	1 week
User interface testing	1 week
Load testing	1 week
Performance testing	1 week
Release to production	1 week

5.4 Testing Environment

Testing environment is a setup of software and necessary hardware for the testing team to execute test cases. In other words, it supports test execution with hardware, software and network configured.

- System and applications
- Test data
- Database server
- Front and running environment.
- Client operating system
- Browser
- Hardware includes Server Operation System
- Network
- Documentation required like reference documents/configuration guides/ installation guides/ user manuals.
-

5.5 Test Cases

5.5.1 Login

<table>
<tr><td><b>Test Case #1</b></td><td><b>Test Case Name:</b> Login</td></tr>
<tr><td><b>System:</b> Student Attendance taking System</td><td><b>Subsystem:</b> Admin, Teachers, Students</td></tr>
<tr><td><b>Designed by</b>: Md. Saidul Islam Satil</td><td><b>Designed date:</b> 02-03-2019</td></tr>
<tr><td><b>Executed by</b>: Md. Saidul Islam Satil</td><td><b>Executed date:</b> 10-03-2019</td></tr>
<tr><td colspan="2"><b>Short description:</b> The user is registered and trying to login to the Audit application website when the system will check validity.</td></tr>
<tr><td colspan="2"><b>Pre-conditions:</b>
<b>1.</b> When any user tries to go any pages as his/her registration type, they will be asked to login first. They have to submit the email id and password for login successfully.
<b>2.</b> Assume that Username is ‘Satil123@gmail.com’ and password is ‘satil123’</td></tr>
</table>

Step	**User name**	**Password**	**Expected response**	**Pass/ Fail**	**Comment**
1	Satil	123	Wrong user name and password		
2		xyz	User name can’t be blank		
3	Password	admin	Invalid user name and password		
4	admin		Password can’t be blank		
5	xys123	Satil123	Invalid user name		
6	Satil123@gmail .com	Satil123	Successfully login		
7.	123satil@yahoo .com	satil	Invalid user name and password		

8	……..	hgjkl	Invalid user name and password		
9			User name and password can’t be blank		
10	Satil123@gmail .com	hhoo	Invalid password		
Post Condition: Registered members will be successfully login in the system.					

5.5.2 Generate String Code

Test case #2	**Test case name:** Generate sting code
System: Student Attendance taking System	**Subsystem:** Take attendance
Designed by: Md. Saidul Islam Satil	**Designed Date:** 25-03-2019
Executed by: Md. Saidul Islam Satil	**Executed date:** 30-03-2019
Short Description: After login teacher must have to select section and course to generate the code number and take the attendance.	
Pre-condition: For this test case, we consider section as “b” and course as “Security”.	

hlStep	**Section**	**Course**	**Try to take attendance**	**Expected response**	**Pass/ fail**	**Comment**
1	b		Yes	Fulfil the course		
2			Yes	Fulfil the section and course		
3		Security	Yes	Fulfil the section		
4	b	security	Yes	Show the string code		
Post condition: Teachers will be successfully generating the string code.						

5.5.3 Submit String Code

Test case #3	**Test case name:** Submit sting code
System: Student Attendance taking System	**Subsystem:** Give attendance
Designed by: Md. Saidul Islam Satil	**Designed Date:** 11-03-2019
Executed by: Md. Saidul Islam Satil	**Executed date:** 15-03-2019
Short Description: After login student must have to select section and course to submit the code number and give the attendance.	
Pre-condition: For this test case, we consider section as “c” and course as “Security”.	

Step	**Section**	**Course**	**Try to Submit String code**	**Expected response**	**Pass/ fail**	**Comment**
1	b		Yes	Fulfil the course		

2			Yes	Fulfil the section and course		
3		Security	Yes	Fulfil the section		
4	b	security	Yes	Show the string code		
Post condition: Teachers will be successfully generating the string code.						

5.5.4 Match String Code

Test case #4	**Test case name:** Match sting code
System: Student Attendance taking System	**Subsystem:** Allow attendance
Designed by: Md. Saidul Islam Satil	**Designed Date:** 16-03-2019
Executed by: Md. Saidul Islam Satil	**Executed date:** 04-03-2019
Short Description: After generating a string code by the teacher, students have to submit that code. Then system will match both the code. If they are same the attendance of that student for that course and the particular day will be given.	
Pre-condition: For this test case, we consider that the generating string code by the teacher is “XSyuZmoN”.	

Step	**Generating String code**	**Submitting string code**	**Expected response**	**Pass/ fail**	**Comment**
1	XSyuZmoN	HnYzij	Mismatch with the key		
2	XSyuZmoN		Null not allowed. Submit the code.		
3	XSyuZmoN	XSyuZmoN	Code matched. Attendance Given		
Post condition: Attendance taking is working successfully.					

5.5.5 Show distance

Test case #5	**Test case name:** Show distance
System: Student Attendance taking System	**Subsystem:** location tracing
Designed by: Md. Saidul Islam Satil	**Designed Date:** 04-04-2019
Executed by: Md. Saidul Islam Satil	**Executed date:** 04-04-2019
Short Description: After login successfully and at the time of submitting the string code, system will ask to allow location to the student so that teacher can understand how far the student is staying from the classroom.	
Pre-condition: At the time of development, we will define the location of university from the classroom. After a student allowing to trace his/her location system will calculate how far he/she is staying from the classroom.	

Step	**Allow location**	**Show Distance**	**Expected response**	**Pass/ fail**	**Comment**
1	No	Location not allowed	Location not allowed		
2		Location not allowed	Location not allowed		

3	Yes	Yes	The distance is “___” km from the classroom.		
Post condition: Location tracing and showing distance is working successfully.					

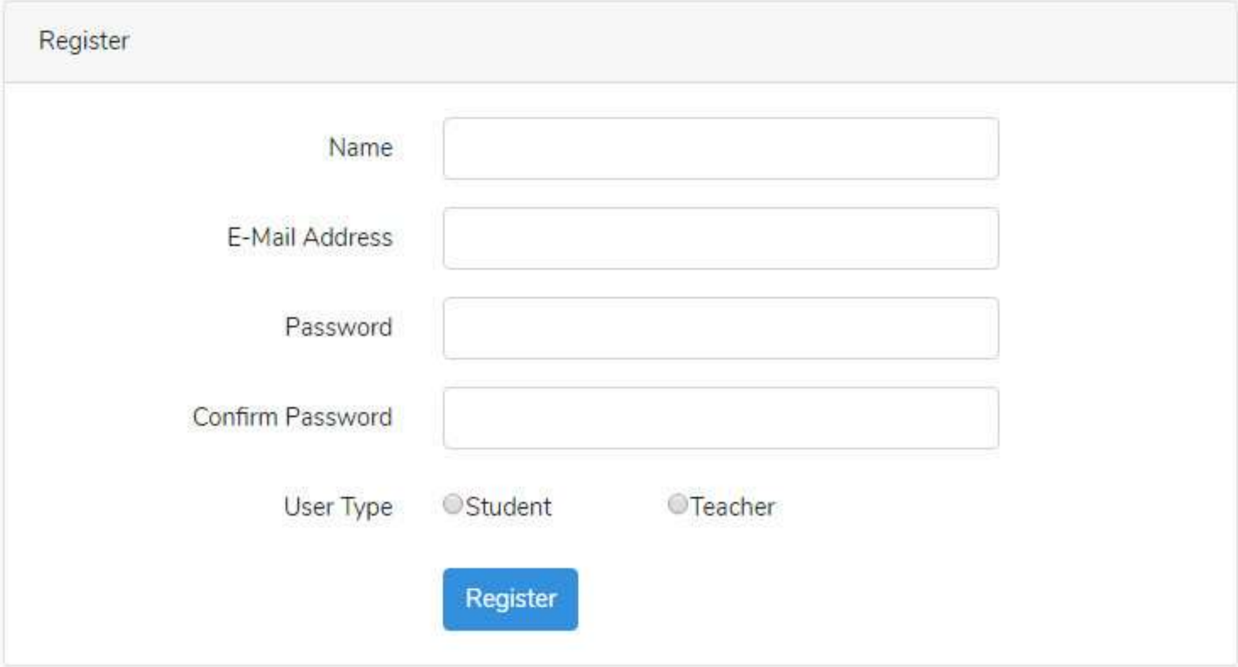
Chapter- 6

User Manual

6. User Manual

6.1 Registration page

User must have to register themselves as their occupation.



Register

Name

E-Mail Address

Password

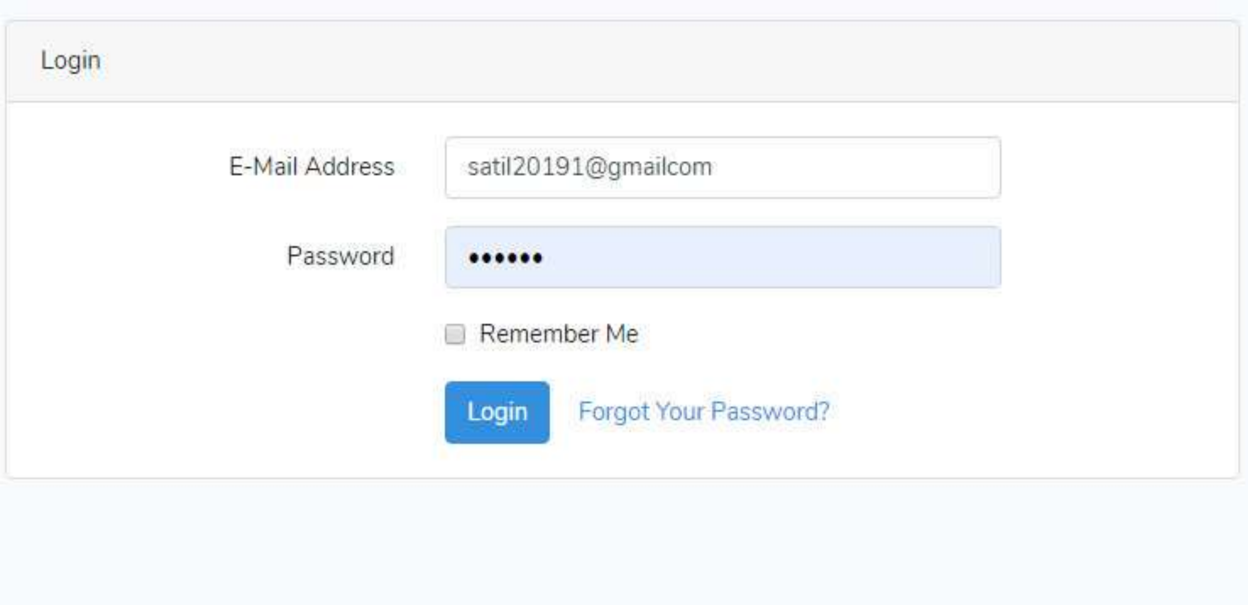
Confirm Password

User Type ○Student ○Teacher

This is the registration page. Here a user will register himself. He must have to select a type that is he student or teacher.

6.2 Login Page

Students must have to login first. After login successfully system will show him/her the page as user type.



Login

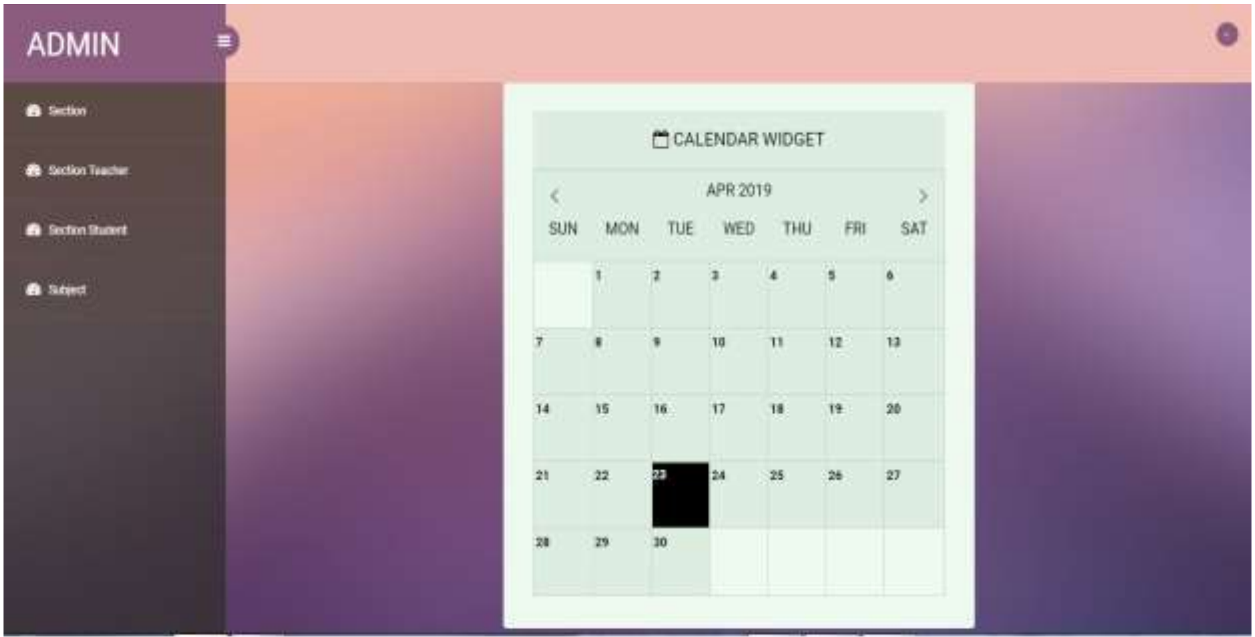
E-Mail Address satil20191@gmailcom

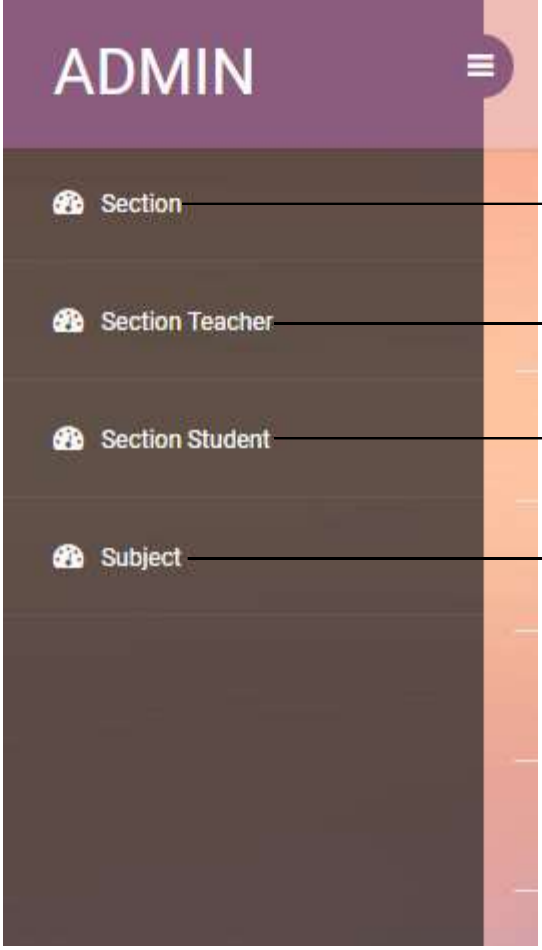
Password ••••••

□ Remember Me

6.3 Admin Dashboard

This will be the admin Dashboard. Here a admin will be able to add students, remove students, add teachers, remove teachers, add courses and remove courses.





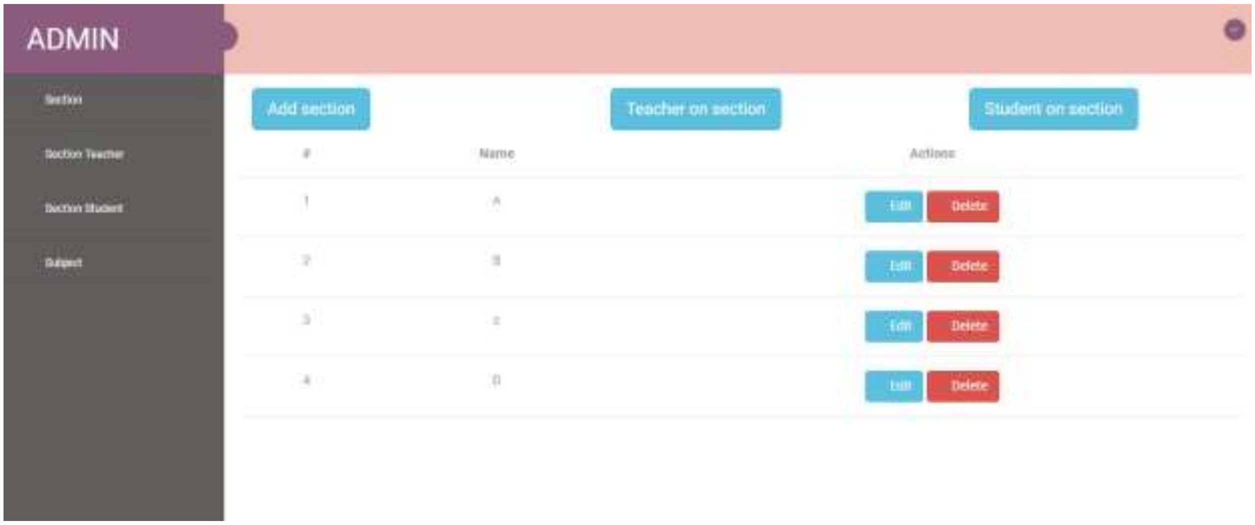
Here Admin will be able to add sections

Here Admin will be able to assign teachers in sections, with courses.

Here Admin will be able to assign students in sections, with courses.

Here Admin will be able to add new courses.

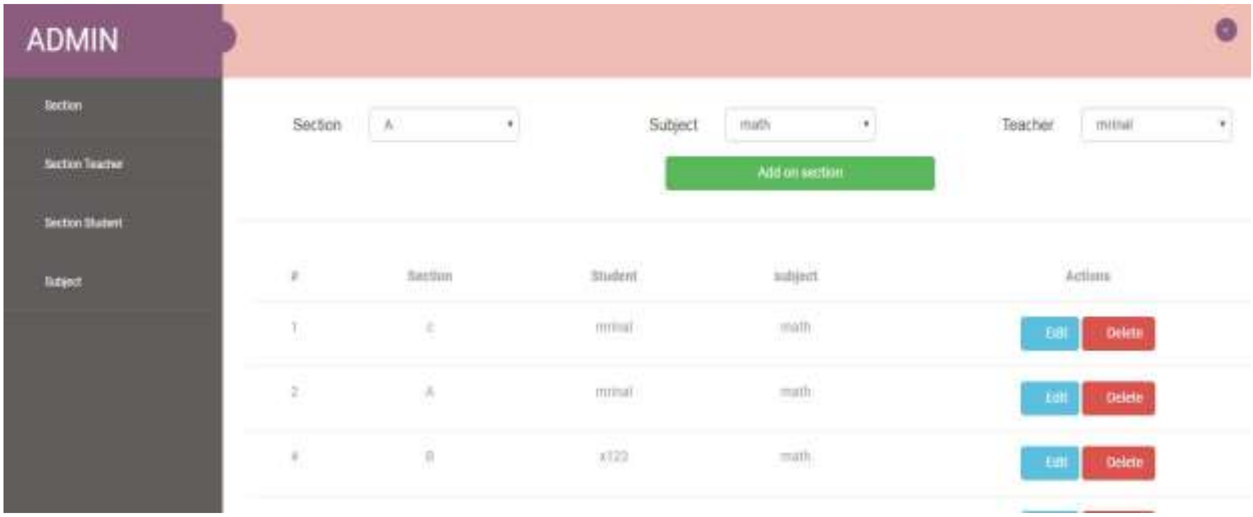
6.4 Add Sections



#	Name	Actions
1	A	Edit Delete
2	B	Edit Delete
3	C	Edit Delete
4	D	Edit Delete

Here Admin will be able to add sections, also able to see the students and teacher list of a section.

6.5 Add Teacher

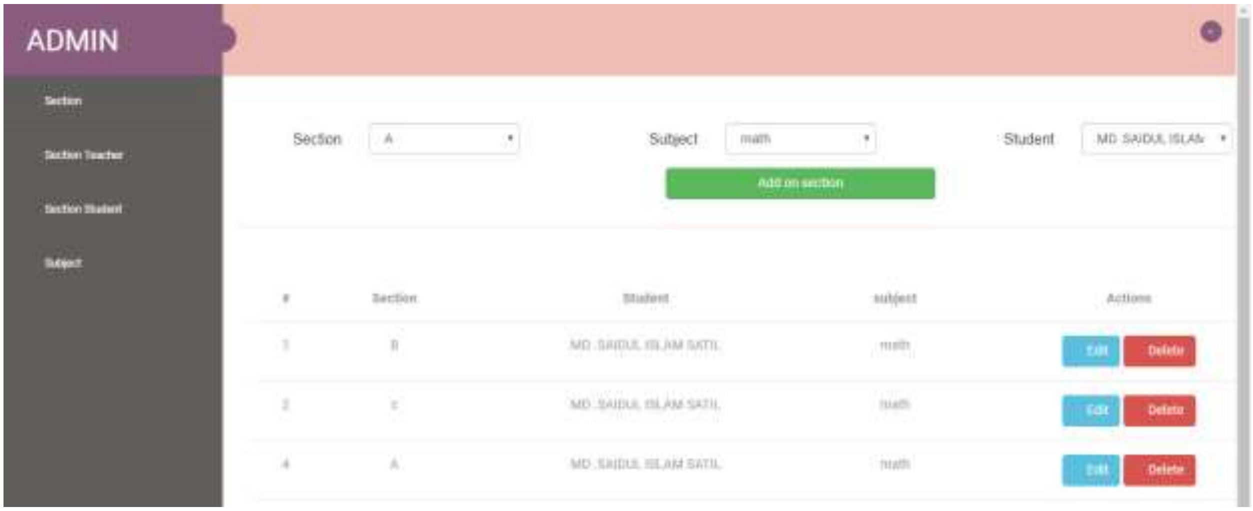


Section: A
Subject: math
Teacher: mrinal

#	Section	Student	subject	Actions
1	C	mrinal	math	Edit Delete
2	A	mrinal	math	Edit Delete
4	B	x123	math	Edit Delete

Here Admin will be able to assign teachers in sections, with courses.

6.6 Add Students



Section: A

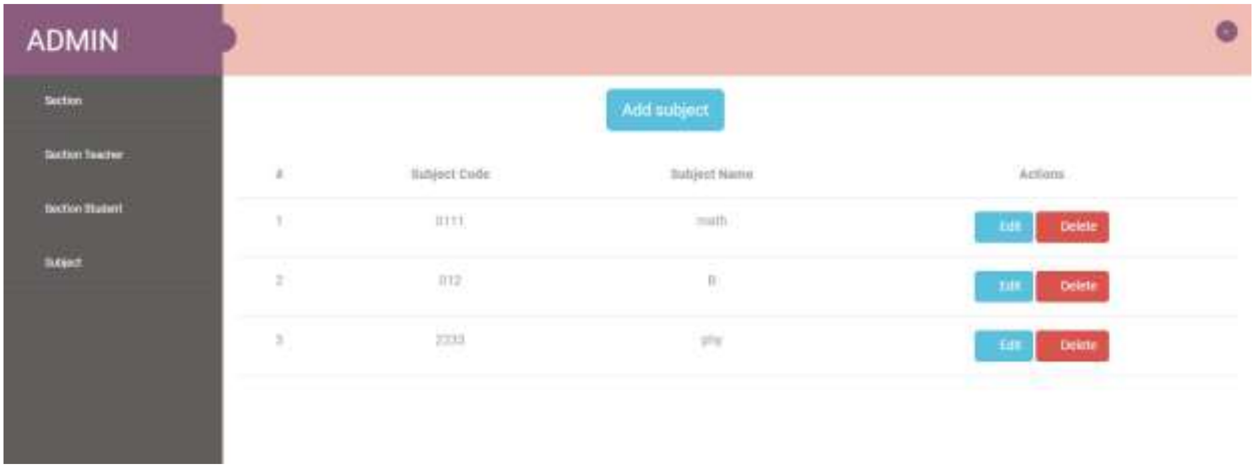
Subject: math

Student: MD. SAIDUL ISLAN

#	Section	Student	subject	Actions
1	B	MD. SAIDUL ISLAM SATIL	math	Edit Delete
2	c	MD. SAIDUL ISLAM SATIL	math	Edit Delete
4	A	MD. SAIDUL ISLAM SATIL	math	Edit Delete

Here Admin will be able to assign students in sections, with courses.

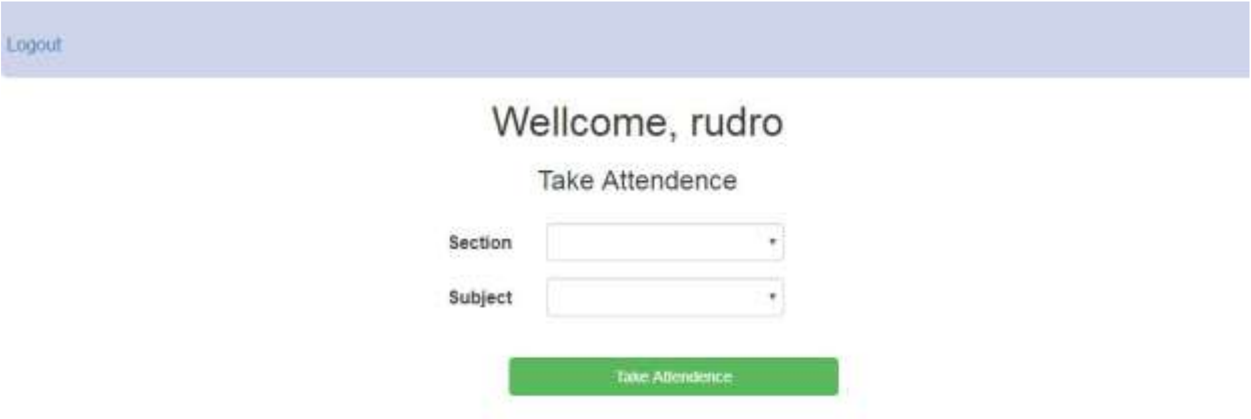
6.7 Add Courses



#	Subject Code	Subject Name	Actions
1	0111	math	Edit Delete
2	012	B	Edit Delete
3	2333	phy	Edit Delete

Here Admin will be able to add new courses with course code.

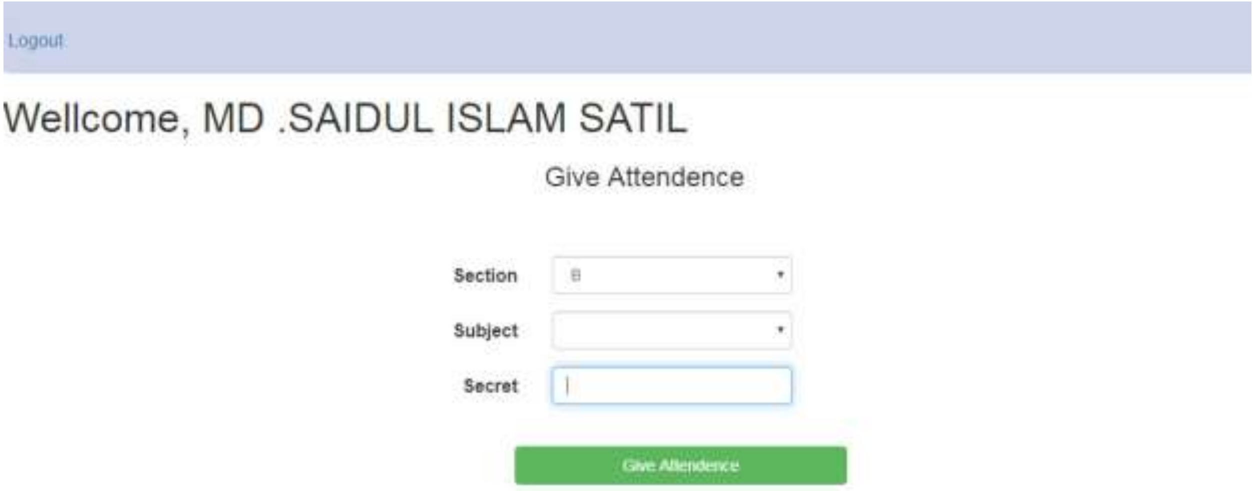
6.8 Teachers Dashboard



Wellcome, rudro

To generate the random string code, teacher must have to fulfil the section and course field. When a teacher will click the button take attendance after fulfil the section and subject field then he/she will get a random string number.

6.9 Student's Dashboard



Wellcome, MD .SAIDUL ISLAM SATIL

Section B

Students must have to fulfil the section and subject field. Then they will submit the code. If match then the attendance will be added to the system, if the string doesn’t match it will show error.

Chapter- 7

Conclusion

7.1 Project Summary

The project has been started from December. From that beginning time I have to work hard to know the client's requirements clearly. I have visited to so many schools, colleges, Universities. I have discussed with the honorable teachers with my software department. I have analysis the class time and how much time waste after attendance taking. Then I had decided to develop this project.

From starting to end, I have tried my best to make is the best. I think storing data in database is very important. That's why I did this first and relationships with tables. After that I design UI. This project's UI is very simple and clean which help for the user's experience. Then I started coding and executing the project.

If I did not test the project, there will stay some bug on this project which will ruin full project. That's why I gave Importance to test the whole project then I solved some bug which I got after testing this project.

7.2 Limitations

Nothing is perfect in this world. So it is impossible to build a system without any limitations. There are also few limitations. They are:

- Not fully responsive
- Not highly secure
- Chance of proxy attendance is not zero

7.3 Future Scope

There is so many scopes to make this project better. Hope that in next version of this system we will be able to add a system crasher so that after submitting the string the site can hang for a particular time so that no one can submit other's attendance.
And for student's facility we develop a mobile app for this system.

7.4 References

- www.google.com
- www.wikipedia.com
- http://github.com/
- http://Youtube.com/
- www.W3school.com/
- www.laravel.com