Project of Software Engineering

School of Computer Science, Shandong University of Technology

Employee management system:

Soft Tech

Presented to
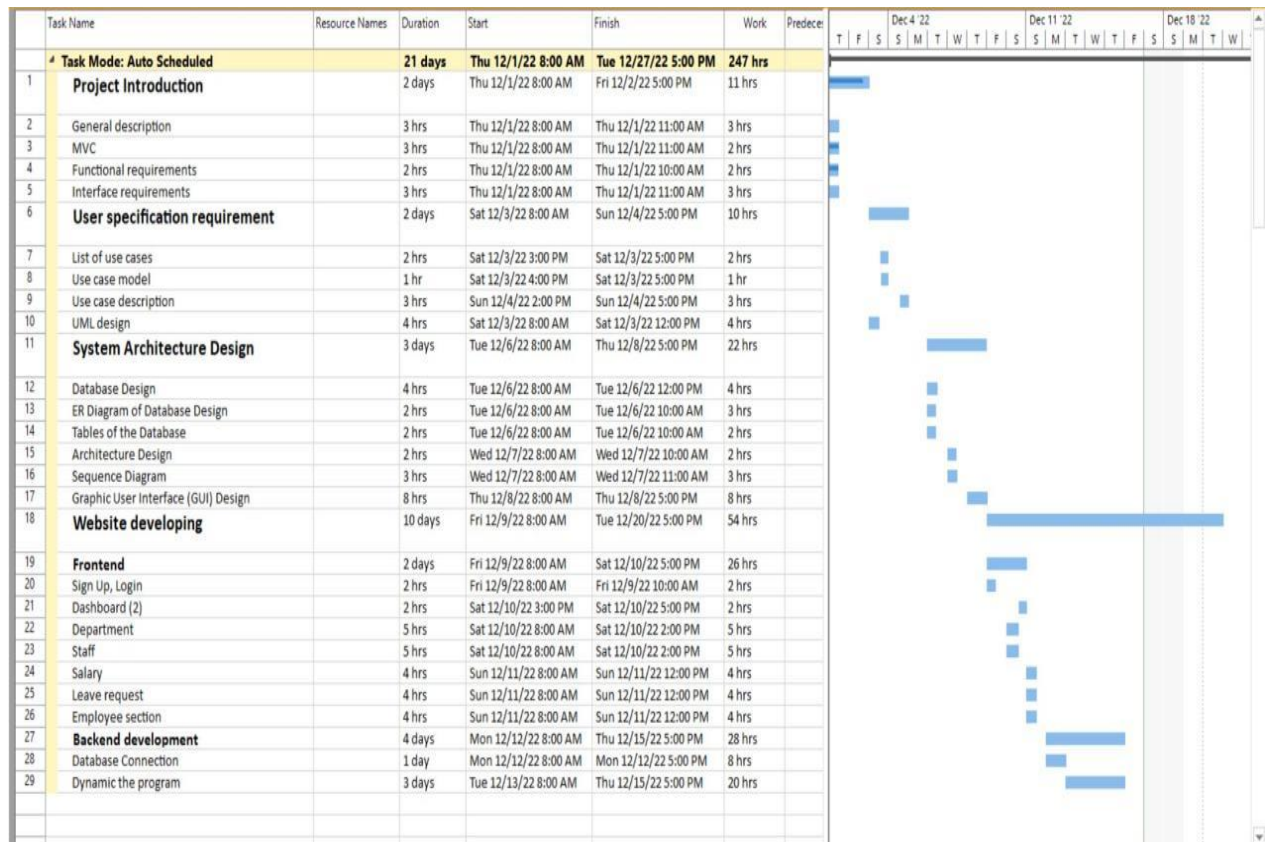
Dr. Zhu Ming

Prepared by

Rokaya Akter (如可亚)

Shuvo Md Touhidul Islam (树坡)

Tuser Md Taybul Islam (泰布尔)

Mohanto Tushar Chandra (杜沙尔)

22th December, 2022

Project Plan

	Task Name	Resource Names	Duration	Start	Finish	Work	Predeces
	Task Mode: Auto Scheduled		21 days	Thu 12/1/22 8:00 AM	Tue 12/27/22 5:00 PM	247 hrs	
1	Project Introduction		2 days	Thu 12/1/22 8:00 AM	Fri 12/2/22 5:00 PM	11 hrs	
2	General description		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	3 hrs	
3	MVC		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	2 hrs	
4	Functional requirements		2 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 10:00 AM	2 hrs	
5	Interface requirements		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	3 hrs	
6	User specification requirement		2 days	Sat 12/3/22 8:00 AM	Sun 12/4/22 5:00 PM	10 hrs	
7	List of use cases		2 hrs	Sat 12/3/22 3:00 PM	Sat 12/3/22 5:00 PM	2 hrs	
8	Use case model		1 hr	Sat 12/3/22 4:00 PM	Sat 12/3/22 5:00 PM	1 hr	
9	Use case description		3 hrs	Sun 12/4/22 2:00 PM	Sun 12/4/22 5:00 PM	3 hrs	
10	UML design		4 hrs	Sat 12/3/22 8:00 AM	Sat 12/3/22 12:00 PM	4 hrs	
11	System Architecture Design		3 days	Tue 12/6/22 8:00 AM	Thu 12/8/22 5:00 PM	22 hrs	
12	Database Design		4 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 12:00 PM	4 hrs	
13	ER Diagram of Database Design		2 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 10:00 AM	3 hrs	
14	Tables of the Database		2 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 10:00 AM	2 hrs	
15	Architecture Design		2 hrs	Wed 12/7/22 8:00 AM	Wed 12/7/22 10:00 AM	2 hrs	
16	Sequence Diagram		3 hrs	Wed 12/7/22 8:00 AM	Wed 12/7/22 11:00 AM	3 hrs	
17	Graphic User Interface (GUI) Design		8 hrs	Thu 12/8/22 8:00 AM	Thu 12/8/22 5:00 PM	8 hrs	
18	Website developing		10 days	Fri 12/9/22 8:00 AM	Tue 12/20/22 5:00 PM	54 hrs	
19	Frontend		2 days	Fri 12/9/22 8:00 AM	Sat 12/10/22 5:00 PM	26 hrs	
20	Sign Up, Login		2 hrs	Fri 12/9/22 8:00 AM	Fri 12/9/22 10:00 AM	2 hrs	
21	Dashboard (2)		2 hrs	Sat 12/10/22 3:00 PM	Sat 12/10/22 5:00 PM	2 hrs	
22	Department		5 hrs	Sat 12/10/22 8:00 AM	Sat 12/10/22 2:00 PM	5 hrs	
23	Staff		5 hrs	Sat 12/10/22 8:00 AM	Sat 12/10/22 2:00 PM	5 hrs	
24	Salary		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
25	Leave request		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
26	Employee section		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
27	Backend development		4 days	Mon 12/12/22 8:00 AM	Thu 12/15/22 5:00 PM	28 hrs	
28	Database Connection		1 day	Mon 12/12/22 8:00 AM	Mon 12/12/22 5:00 PM	8 hrs	
29	Dynamic the program		3 days	Tue 12/13/22 8:00 AM	Thu 12/15/22 5:00 PM	20 hrs	

Software Requirements Specification (SRS) Document

Introduction

1.1 Introduction

The purpose of this document is to define and describe the requirements of the Employee management system project and to spell out the system's functionality and its constraints. In our project, we followed the MVC pattern.

◆ This web application called "Soft Tech" allows users to create, store, and manage employee records.

◆ Additionally, the application features a payroll system and a leave application.

◆ It is a management method that aids in the management of both people and the many tasks and projects undertaken by the company.

◆ This management systems can handle and store employee information, including personal data, as well as information about various departments and organizational branches.

In this project we followed waterfall model:

Waterfall Model

1. User Requirement:
2. Software Requirement:
3. Architecture Design:
4. Detailed Design & Coding:
5. Testing:
6. Delivery:

1.2 Scope of this Document

The admin and the employee for the system are the user of the Soft Tech website, developed by our Team. Our constraints for this section includes our deadline for the document which is due 22-12-2022.

This web application contains two account access:

◆Admin

◆Employee

All three accounts contain different privileges and authentication

Admin: Here Admin is the owner of this Application (Company).He can manage the employee, their department, their performance statistics (attendance, leave request, query information and other functions)

Employee: Here employee can see their monthly task, payment Information, attendance, and they can contact with the leader.

1.3 Overview

The data is an Access Database with import/export capabilities from mysql database that will hold information on admin, employee, their details information, their department, their salary, their leave request, including their information management section for admin.

General description

2.1 website Functions

The website should make input of data and the entire checkout process easier and streamlined for the users (employees) and time efficient for the owner of a company.

2.2 Similar System Information

The website is being developed with Access, so there are a large number of similar databases or systems and they are used for a wide array of different purposes.

Technology we used(for frontend and backend) :

- ◆ Html
- ◆ CSS
- ◆ PHP
- ◆ MySQL

2.3 User Characteristics

The users include the employees of the Soft Tech as they input data. For this system, the user is required to know the website address.

2.6 General Constraints

Constraints include an easy to use interface for the program through forms, a Windows platform or, at bare minimum, a Mac with Access and Excel for Mac installed. For better view chrome browser is recommended.

Functional requirements

1. **Items provided to the Soft Tech shall be stored in the MySQL Database.**
    1. Items shall be stored on mysql database and have complete fields.

    2. Very high criticality

    3. Limited network / wi-fi availability could present a technical challenge

    4. The above stated factor is a risk we have encountered. Eliminate it by reducing the dependency of our program on these things.

    5. This requirement is the basis of the project; all other aspects depend on it.

2. **The items shall be accessible via queries and reports.**
    1. Users of the database should be able to run reports on the data that has been put into the database. They should also be able to run queries.

    2. Very high criticality

    3. We do not foresee any technical issues preventing the implementation of this.

    4. Given the capabilities of Access, this requirement is able to be satisfied.

    5. This requirement depends on requirement number one.

3. **The data stored should be able to be manipulated through forms.**
    1. Items and other data should be able to be added and updated through the use of forms.

    2. Very high criticality

    3. We do not foresee any technical risks involved in this requirement.

    4. The only factor we can encounter here is the user of the system not being able to use it correctly. We will overcome this by training those who will be using it.

    5. This requirement is dependent on requirement one.

Item Description

ID AUC01

Name: AdminLogin

Actor(s): Admin

Trigger The user wants to use the login functionality/

Preconditions The admin open the website

Post-conditions The admin successfully login

Main Scenario 1. Admin enters an email address

2. The website shows the email entered by the admin

3. Admin enters a password

4. The website show the password entered previously with the email address

5. The admin press login button

6. The admin successfully login to the website

Alternative

Scenario 1. a. the admin enters a wrong email

b. the admin deletes the wrong email

c. the website shows warning message that the email is incorrect

d. the scenario continue with the main scenario at step 1.

1.1. List of Use Cases

Table 1-1 List of Use Cases

Actor	Use Case ID and Name	Use Case Description
Admin	AUC01**AdminLogin	The admin can get logged in
	AUC02**AdminLogout	The admin can get logged out
	AUC03**Dashboard	Admin can access the dashboard
	AUC04**Department	Admin can access the department section
	AUC05**AddDepartment	Admin can add any employee department
	AUC06**ManageDepartment	Admin can manage(edit & delete) any employee department
	AUC07**Staff	Admin can access the employee section
	AUC08**AddStaff	Admin can add any employee
	AUC09**ManageStaff	Admin can manage(edit & delete) any employee
	AUC10**Salary	Admin can pay through the salary section

	AUC10**AddSalary	Admin can set the amount for salary
	AUC11**ManageSalary	Admin can update the amount of salary
	AUC12**Leave	Admin can manage the leave request
	AUC12**ManageStaffsLeave	Admin can approve or reject the leave request
	AUC13**LeaveHistory	Admin can review employees leave details
Employee	EUC01**EmployeeLogin	Employee can get logged in
	EUC02**EmployeeLogout	Employee can get logged out
	EUC03**Dashboard	Employee can access in dashboard
	EUC04**Salary	Employee can view and take receipt of salary
	EUC05**Leave	Employee can apply for leave
	EUC06**LeaveHistory	Employee can review leave history

1.2. Use Case Model
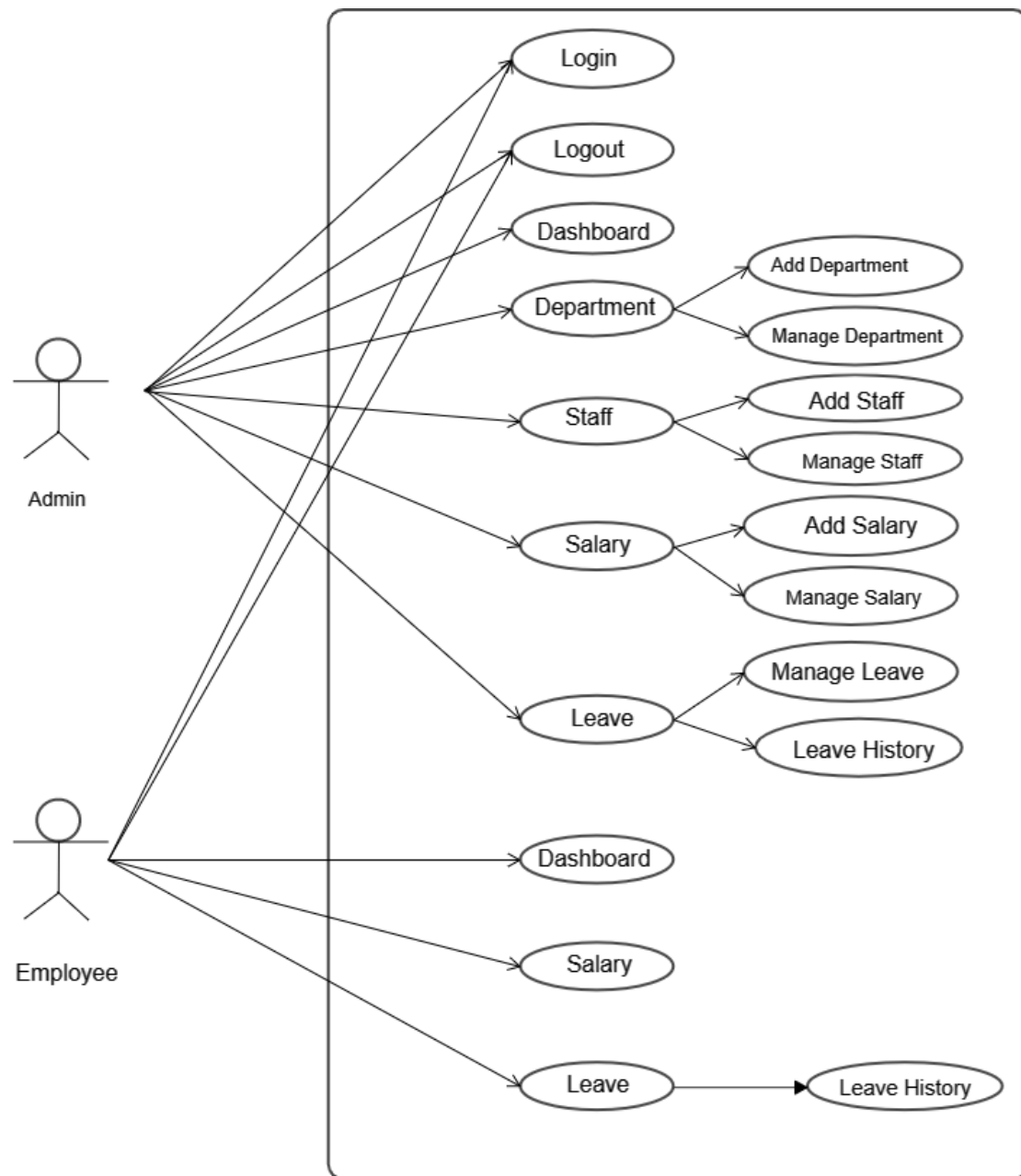
1.3 Description of Use Cases

1.3.1. AUC01 AdminLogin

Item	Description
ID	AUC01

Name	AdminLogin
Actor(s)	Admin
Trigger	The admin wants to use the login functionality
Preconditions	Open the website
Postconditions	The admin successfully logged into the website
Main Scenario	1. The admin enters a email 2. The website shows the email entered by the admin 3. User enter a password 4. The website show the password entered by the admin 5. The admin press the login button 6. The website verify the login details and successful login function
Alternative Scenario	a) The admin enters a wrong email b) The admin enters a wrong password c) The website shows "forgot password?" option if admin forget the password d) The website shows "create account" option if the admin don't have an account in the website
Other Requirements	

AUC02 AdminLogout

Item	Description
ID	AUC02
Name	AdminLogout
Actor(s)	Admin
Trigger	The admin wants to use the logout functionality
Preconditions	Open the website

Postconditions	The admin successfully log out from the website
Main Scenario	1. The admin press logout button 2. The website successfully logout from the website
Alternative Scenario	
Other Requirements	

AUC03 Dashboard

Item	Description
ID	AUC03
Name	Dashboard
Actor(s)	Admin
Trigger	The admin wants to use the dashboard functionality
Preconditions	The admin log into the website
Postconditions	The admin successfully use the dashboard functionality
Main Scenario	1. The admin log into the website 2. Admin can access into employee details 2. Admin can manage and review company information
Alternative Scenario	1. The admin move other pages

Item	Description
Other Requirements	

AUC04 Department

Item	Description
ID	AUC04
Name	Department
Actor(s)	Admin
Trigger	The admin wants to use the department functionality
Preconditions	The admin open the dashboard
Postconditions	The admin successfully use the department functionality
Main Scenario	1. The admin press the department option 2. Admin can preview the previous department history 3. Admin can create a new department 4. Admin can delete an existing department 5. Admin can modify the department information
Alternative Scenario	1. Admin can go back to the previous page
Other Requirements	

AUC05 Staff

Item	Description
ID	AUC05
Name	Staff

Actor(s)	Admin
Trigger	The admin wants to use the Staff functionality
Preconditions	The admin open the staff section
Postconditions	The admin successfully use the staff functionality
Main Scenario	1. The admin press the staff option 2. Admin can preview the previous staff history 3. Admin can assign a new satff 4. Admin can delete an existing staff information 5. Admin can modify the staff information
Alternative Scenario	1. Admin can go back to the previous page 2. Admin can move to other sections from the menu bar
Other Requirements	

AUC06 Salary

Item	Description
ID	AUC06
Name	Salary
Actor(s)	Admin
Trigger	The admin wants to use the Salary functionality
Preconditions	The admin open the Salary section
Postconditions	The admin successfully use the salary functionality
Main Scenario	1. The admin press the Salary option 2. Admin can preview the previous Salary history 3. Admin can assign salary for employee 4. Admin can upload a receipt for the salary

Item	Description
Alternative Scenario	3. Admin can go back to the previous page 4. Admin can move to other sections from the menu bar
Other Requirements	

AUC07 Leave

Item	Description
ID	AUC07
Name	Leave
Actor(s)	Admin
Trigger	The admin wants to use the leave functionality
Preconditions	The admin open the leave section
Postconditions	The admin successfully use the leave functionality
Main Scenario	1. The admin press the leave option 2. Admin can preview the previous leave history 3. Admin can approave leave request for employee 4. Admin can delete leave request for employee
Alternative Scenario	5. Admin can go back to the previous page 6. Admin can move to other sections from the menu bar
Other Requirements	

EUC01 EmployeeLogin

Item	Description

ID	EUC01
Name	EmployeeLogin
Actor(s)	Employee
Trigger	The employee wants to use the employeeLogin functionality
Preconditions	The employee open the login option
Postconditions	The employee successfully use the login functionality
Main Scenario	1. The employee enters an email 2. The website shows the email entered by the employee 3. The employee enter a password 4. The website show the password entered by the employee 5. The employee press the login button 6. The website verify the login details and successful login function
Alternative Scenario	a) The employee enters a wrong email b) The employee enters a wrong password c) The website shows "forgot password?" option if employee forget the password d) The website shows "create account" option if the employee don't have an account in the website
Other Requirements	

EUC02 EmployeeLogout

Item	Description
ID	EUC02
Name	EmployeeLogout
Actor(s)	Employee
Trigger	The employee wants to use the employeeLogout functionality

Preconditions	The employee open the log out option
Postconditions	The employee successfully use the logout functionality
Main Scenario	1. The employee press logout button 2. The website successfully logout from the website
Alternative Scenario	
Other Requirements	

EUC03 Dashboard

Item	Description
ID	EUC03
Name	Dashboard
Actor(s)	Employee
Trigger	The employee wants to use the Dashboard functionality
Preconditions	The employee open the dashboard option
Postconditions	The employee successfully use the dashboard functionality
Main Scenario	1. The employee log into the website 2. Employee can access into his details information 2. Employee can manage and review their information
Alternative Scenario	

Item	Description
Other Requirements	

EUC04 Salary

Item	Description
ID	EUC04
Name	Salary
Actor(s)	Employee
Trigger	The employee wants to use the Salary functionality
Preconditions	The employee open the Salary option
Postconditions	The employee successfully use the Salary functionality
Main Scenario	1. The employee press salary option 2. Employee can view their salary information 2. Employee can get the receipt of the salary
Alternative Scenario	a) Employee can go back to the previous page b) Employee can move to other sections from the menu bar
Other Requirements	

EUC05 Leave

Item	Description
ID	EUC05
Name	Leave
Actor(s)	Employee

Trigger	The employee wants to use the leave functionality
Preconditions	The employee open the leave option
Postconditions	The employee successfully use the leave functionality
Main Scenario	1. The employee press leave option 2. Employee can ask for leave approval
Alternative Scenario	c) Employee can go back to the previous page d) Employee can move to other sections from the menu bar
Other Requirements	

EUC06 Leave

Item	Description
ID	EUC06
Name	Leave
Actor(s)	Employee
Trigger	The employee wants to use the leave functionality
Preconditions	The employee open the leave option
Postconditions	The employee successfully use the leave functionality
Main Scenario	1. The employee press leave option 2. Employee can ask for leave approval 3. Employee can watch their leave history

Alternative **Scenario**	e) Employee can go back to the previous page f) Employee can move to other sections from the menu bar
Other Requirements	

Scheduling: Grant chart

	Task Name	Resource Names	Duration	Start	Finish	Work	Predece
	Task Mode: Auto Scheduled		**21 days**	**Thu 12/1/22 8:00 AM**	**Tue 12/27/22 5:00 PM**	**247 hrs**	
1	**Project Introduction**		2 days	Thu 12/1/22 8:00 AM	Fri 12/2/22 5:00 PM	11 hrs	
2	General description		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	3 hrs	
3	MVC		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	2 hrs	
4	Functional requirements		2 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 10:00 AM	2 hrs	
5	Interface requirements		3 hrs	Thu 12/1/22 8:00 AM	Thu 12/1/22 11:00 AM	3 hrs	
6	**User specification requirement**		2 days	Sat 12/3/22 8:00 AM	Sun 12/4/22 5:00 PM	10 hrs	
7	List of use cases		2 hrs	Sat 12/3/22 3:00 PM	Sat 12/3/22 5:00 PM	2 hrs	
8	Use case model		1 hr	Sat 12/3/22 4:00 PM	Sat 12/3/22 5:00 PM	1 hr	
9	Use case description		3 hrs	Sun 12/4/22 2:00 PM	Sun 12/4/22 5:00 PM	3 hrs	
10	UML design		4 hrs	Sat 12/3/22 8:00 AM	Sat 12/3/22 12:00 PM	4 hrs	
11	**System Architecture Design**		3 days	Tue 12/6/22 8:00 AM	Thu 12/8/22 5:00 PM	22 hrs	
12	Database Design		4 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 12:00 PM	4 hrs	
13	ER Diagram of Database Design		2 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 10:00 AM	3 hrs	
14	Tables of the Database		2 hrs	Tue 12/6/22 8:00 AM	Tue 12/6/22 10:00 AM	2 hrs	
15	Architecture Design		2 hrs	Wed 12/7/22 8:00 AM	Wed 12/7/22 10:00 AM	2 hrs	
16	Sequence Diagram		3 hrs	Wed 12/7/22 8:00 AM	Wed 12/7/22 11:00 AM	3 hrs	
17	Graphic User Interface (GUI) Design		8 hrs	Thu 12/8/22 8:00 AM	Thu 12/8/22 5:00 PM	8 hrs	
18	**Website developing**		10 days	Fri 12/9/22 8:00 AM	Tue 12/20/22 5:00 PM	54 hrs	
19	**Frontend**		2 days	Fri 12/9/22 8:00 AM	Sat 12/10/22 5:00 PM	26 hrs	
20	Sign Up, Login		2 hrs	Fri 12/9/22 8:00 AM	Fri 12/9/22 10:00 AM	2 hrs	
21	Dashboard (2)		2 hrs	Sat 12/10/22 3:00 PM	Sat 12/10/22 5:00 PM	2 hrs	
22	Department		5 hrs	Sat 12/10/22 8:00 AM	Sat 12/10/22 2:00 PM	5 hrs	
23	Staff		5 hrs	Sat 12/10/22 8:00 AM	Sat 12/10/22 2:00 PM	5 hrs	
24	Salary		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
25	Leave request		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
26	Employee section		4 hrs	Sun 12/11/22 8:00 AM	Sun 12/11/22 12:00 PM	4 hrs	
27	**Backend development**		4 days	Mon 12/12/22 8:00 AM	Thu 12/15/22 5:00 PM	28 hrs	
28	Database Connection		1 day	Mon 12/12/22 8:00 AM	Mon 12/22/22 5:00 PM	8 hrs	
29	Dynamic the program		3 days	Tue 12/13/22 8:00 AM	Thu 12/15/22 5:00 PM	20 hrs	

System Architecture Design

1. Introduction

1.1 System Introduction

The purpose of this document is to describe the detailed design of Employee management system. Its main contents include:

System function introduction:

Implementing Methodology

Model View Controller or MVC as it is popularly called, is a software design pattern for developing web applications. A Model View Controller pattern is made up of the following three parts:

- Model - The lowest level of the pattern which is responsible for maintaining data.

- View - This is responsible for displaying all or a portion of the data to the user.

- Controller - Software Code that controls the interactions between the Model and View.

MVC is popular as it isolates the application logic from the user interface layer and supports separation of concerns. Here the Controller received all request for the application and then works with the Model to prepare any data needed by the View. The VIre then uses the data prepared by the Controller to generate a final presentable response. The MVC abstraction can be graphically represented as follows. In this project we followed MVC pattern.

Model View Controller Flow diagram

1.2 Readers of this document

The readership of this document includes:

1. Admin

2. Employee

1.3 References

1. Dr. Zhu Ming, "Software Engineering Course", Shandong University of Technology.

2. Database Design

2.1. ER Diagram of Database Design

login_tbl

| id |
| username |
| password |
| usertype |

department_tbl

| Id |
| department_name |
| added_on |

staff_tbl

| id |
| staff_name |
| gender |
| email |
| mobile |
| dob |
| doj |
| address |
| city |
| state |
| country |
| department_id |
| pic |
| added_by |
| updated_on |
| added_on |

leave_tbl

| id |
| staff_id |
| leave_reason |
| description |
| leave_from |
| leave_to |
| applied_on |

salary_tbl

| id |
| staff_id |
| basic_salary |
| allowance |
| total |
| added_by |
| updated_on |
| added_on |
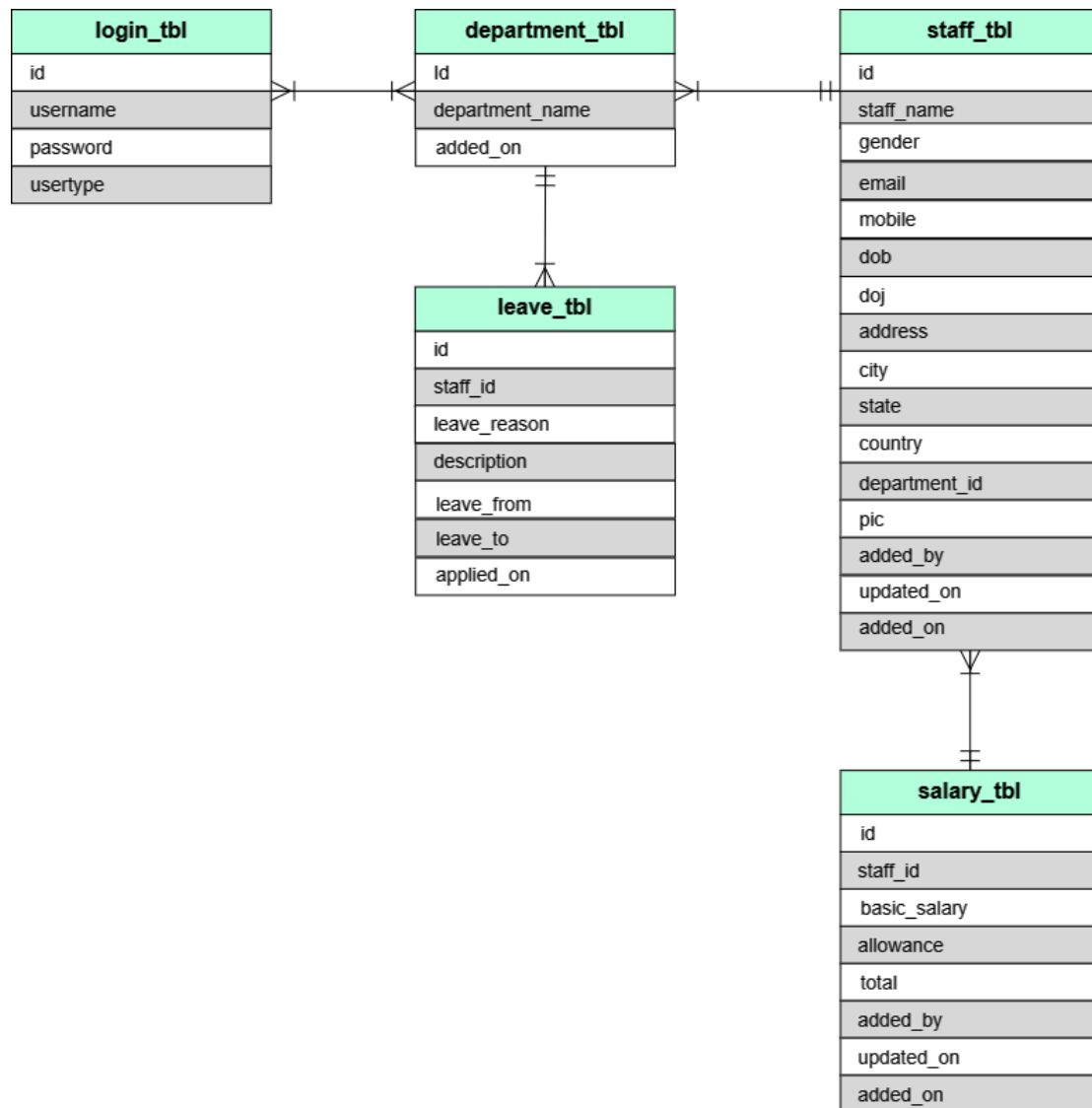
2.2 Tables of the Database

2.2.1 login_tbl

Field Name	Type of data	Length
id	int	11
username	varchar	80
password	varchar	80

usertype	int	11

2.2.2 department_tbl

Field Name	Type of data	Length
id	int	11
Department_name	varchar	100
Added_on	timestamp	

2.2.3 leave_tbl

Field Name	Type of data	Length
id	int	11
Staff_id	int	11
Leave_reason	varchar	90
description	text	
Leave_from	date	
Leave_to	date	
Applied_on	date	

2.2.4 staff_tbl

Field Name	Type of data	Length
id	int	11
Staff_name	varchar	150
gender	varchar	10
email	varchar	150
mobile	bigint	20
dob	date	
doj	date	
address	text	

Field Name	Type of data	Length
city	varchar	100
state	varchar	100
country	varchar	100
Department_id	int	11
pic	varchar	150
Added_by	int	11
Updated_on	date	
Added_on	timestamp	

2.2.5 salary_tbl

Field Name	Type of data	Length
id	int	11
Staff_id	int	11
Basic_salary	bigint	20
allowance	bigint	20
total	bigint	20
Added_by	int	11
Updated_on	date	
Added_on	timestamp	

3. Architecture Design

3.1 Class diagram
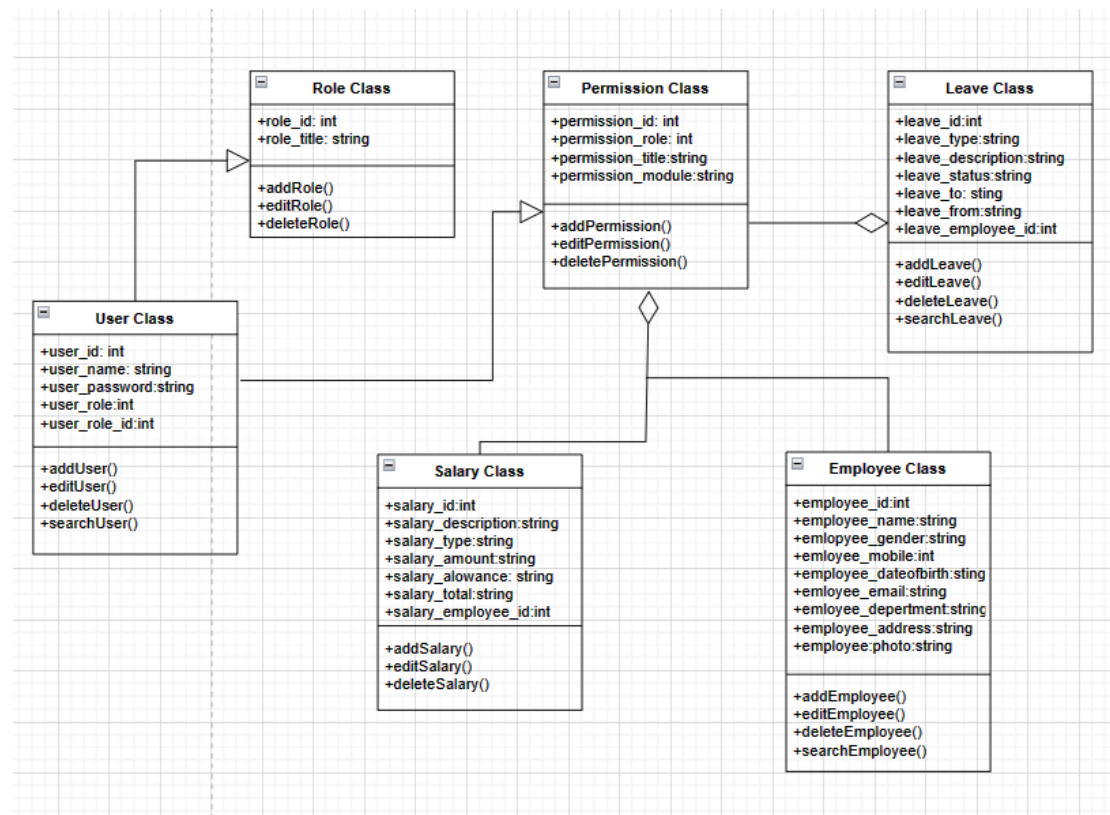
3.1.2 Sequence diagram
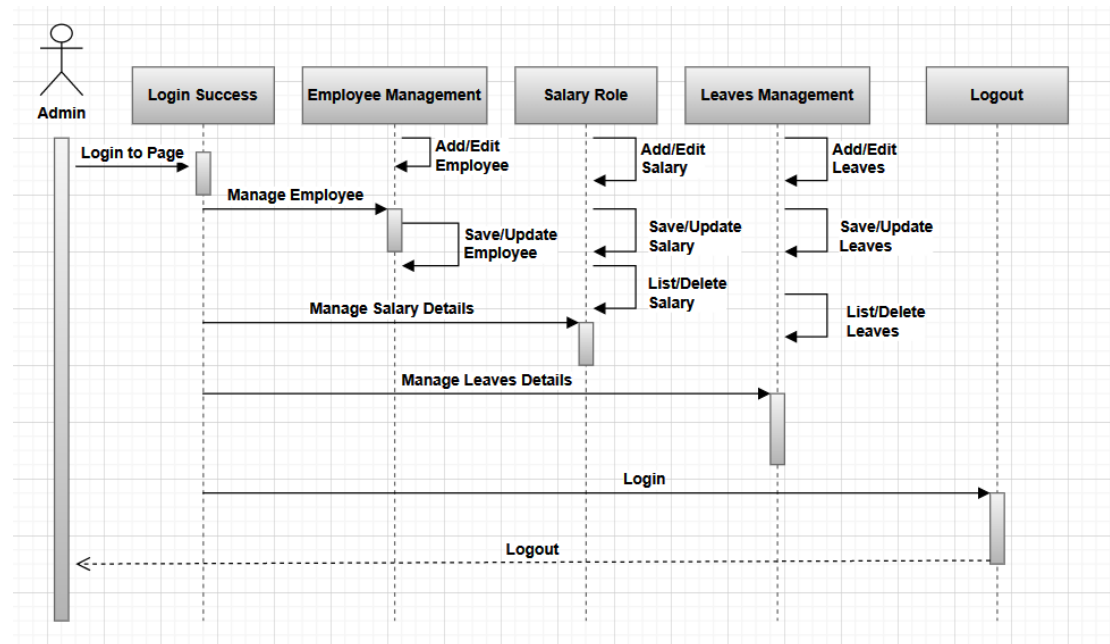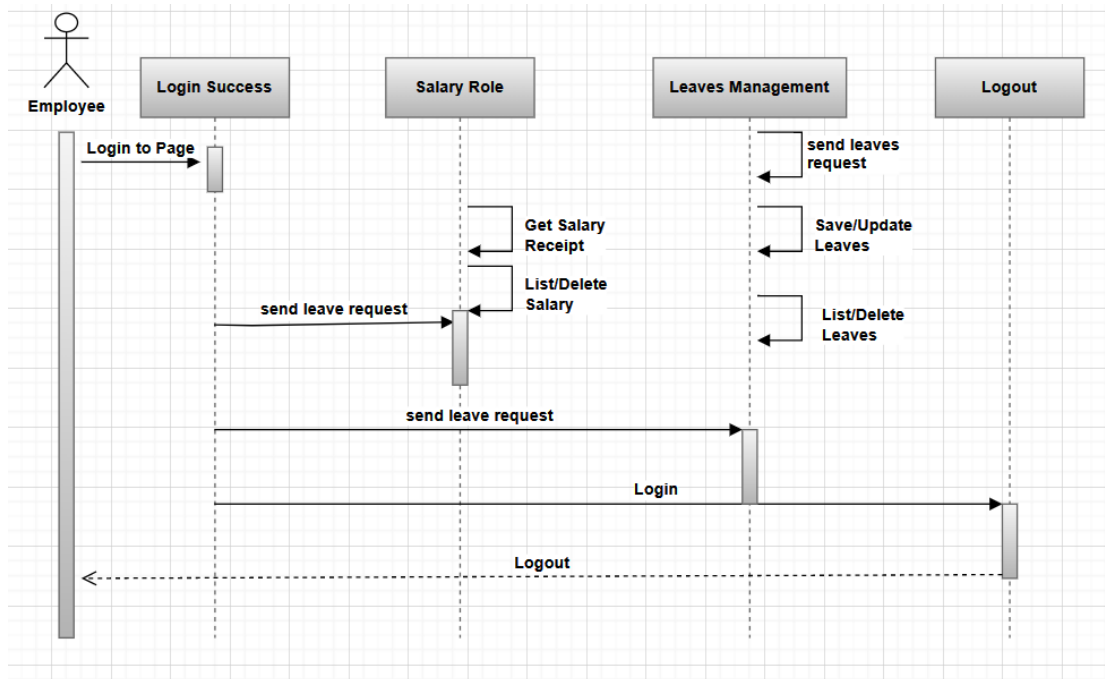
4. Graphic User Interface (GUI) Design

4.1 Prototype GUI of the Welcome Window
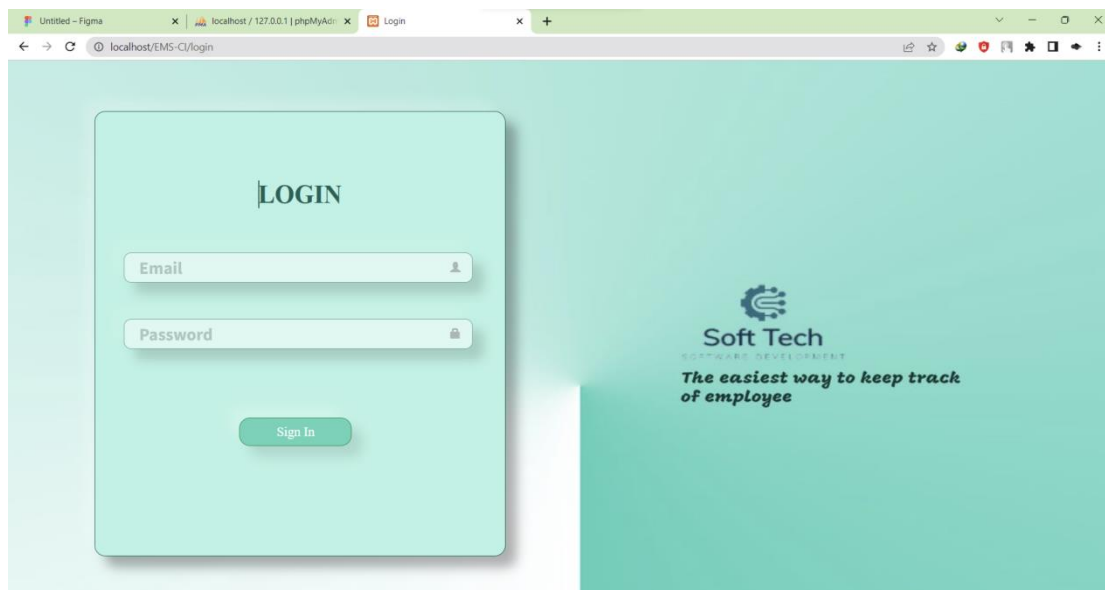
4.2 AUC01 Prototype GUI Admin Dashboard
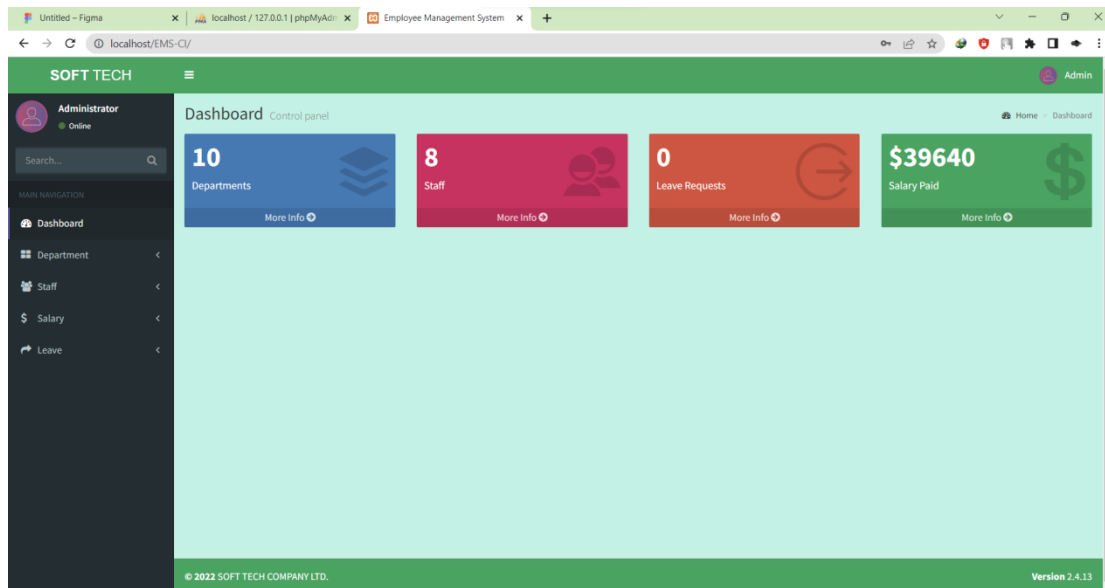
4.3 EUC01 Prototype GUI Employee Dashboard
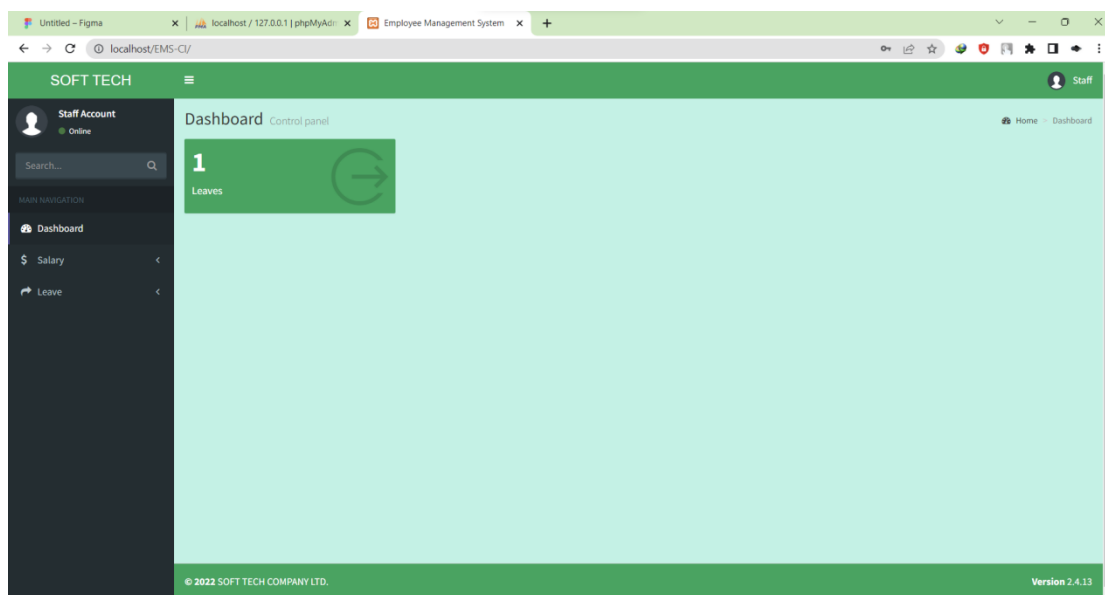
Testing Report

1 Introduction

1.1 Objective

The main purpose of this project is to detect the possible uncovered errors and analysis the values.

1.2 References

1. Dr. Zhu Ming, "Software Engineering Course", Shandong University of Technology, 2nd September, 2021.

1.3 Testing Plan (Black box testing and White box testing)

1.3.1 Structural testing (Black box testing)

Basis path coverage

1. Index function
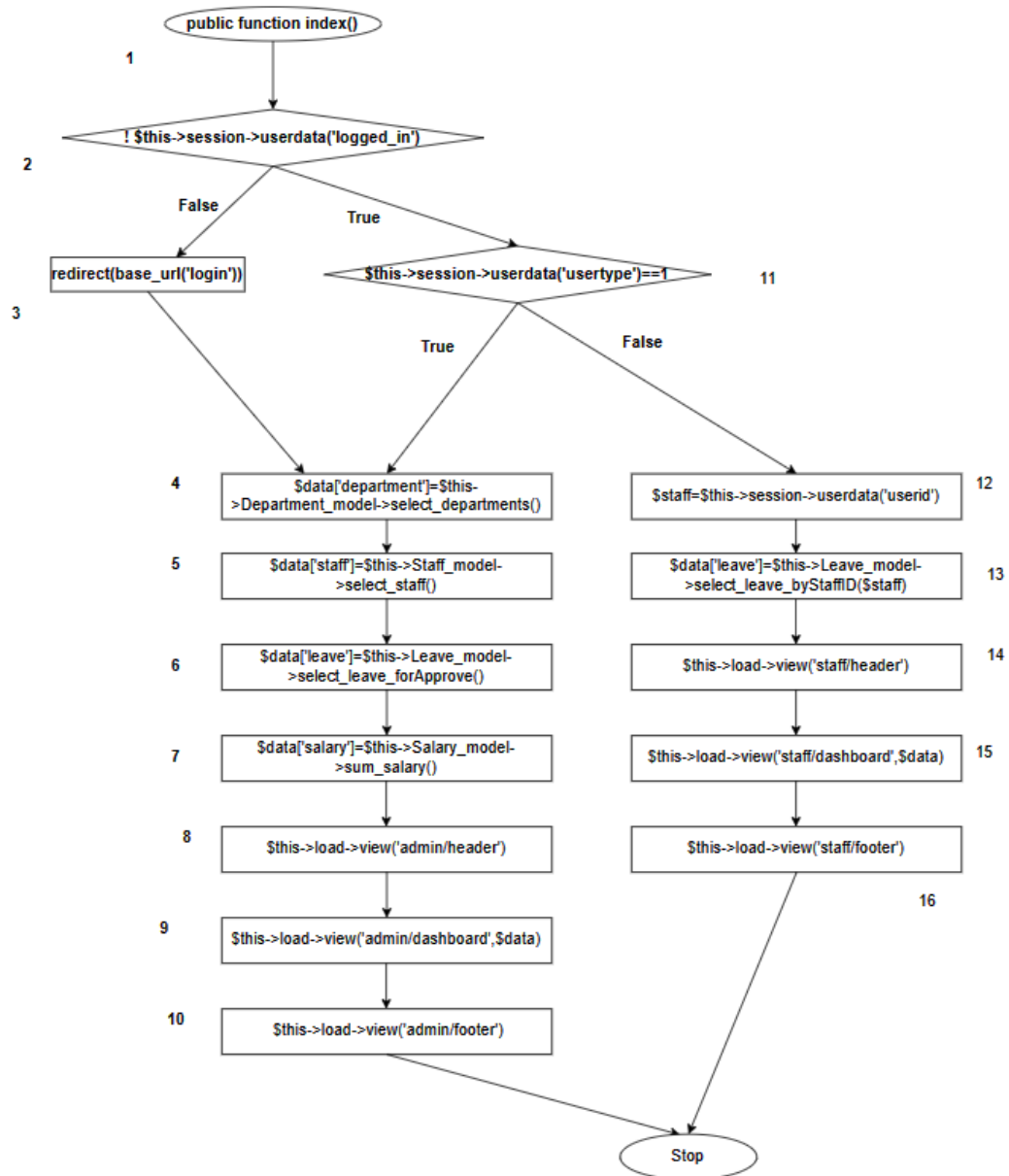
Minimum number of test cases = e − n + 2 = 18 - 17 + 2 = 3

Test cases: {1,2,3,4,5,6,7,8,9,10,17}, {1,2,11,4,5,6,7,8,9,10,17}, {1,2,11,12,13,14,15,16,17}

2. Login Function
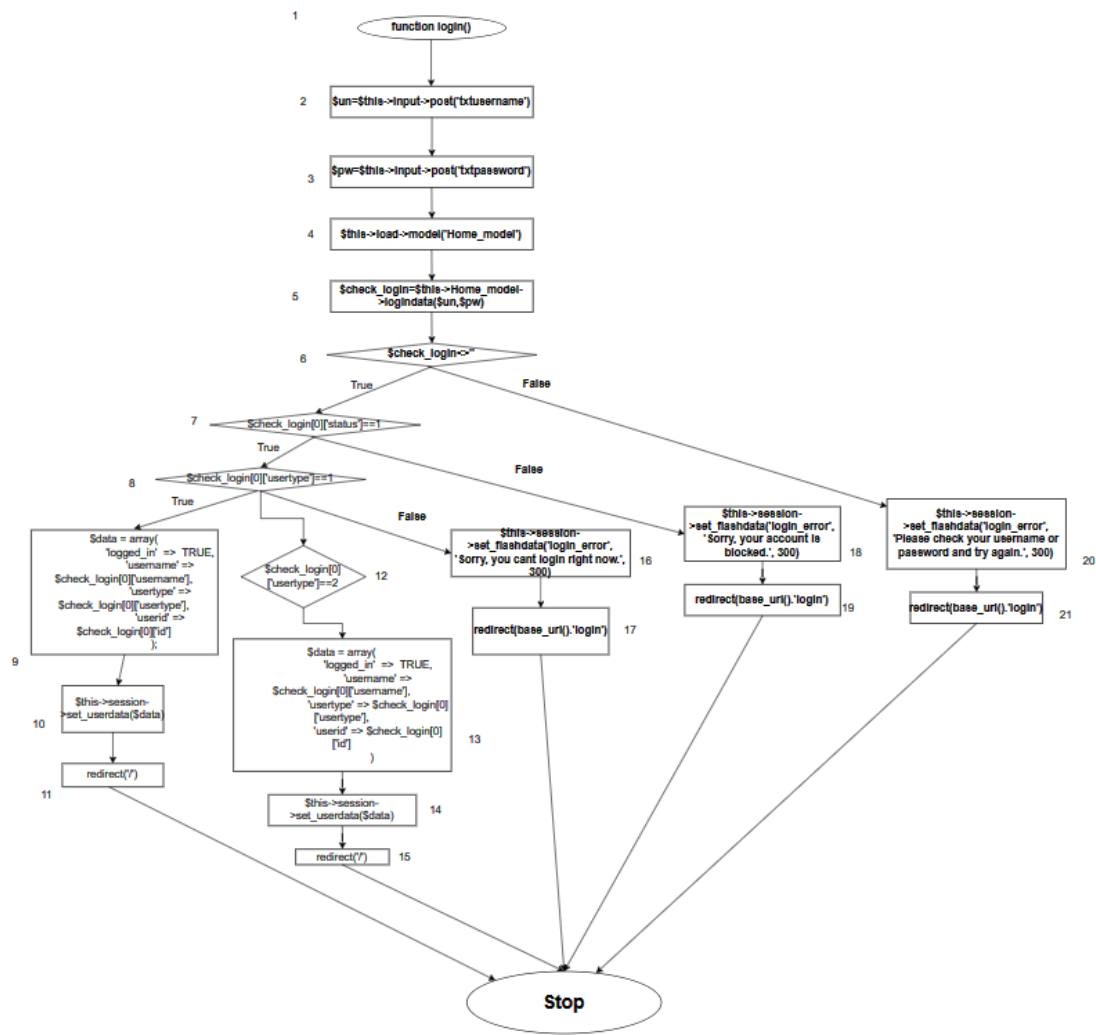
Minimum number of test cases = e − n + 2 = 25 - 22 + 2 = 5

Test cases: {1,2,3,4,5,6,7,8,9,10,11,22}, {1,2,3,4,5,6,7,8,12,13,14,15,22}, {1,2,3,4,5,6,7,8,16,17,22}, {1,2,3,4,5,6,7,18,19,22}, {1,2,3,4,5,6,20,21,22}

1.3.2 Decomposition tree

1.3.3 Functional testing

1. Boundary Value testing for login function

We took two input variable for login functionality

So, the number of test cases is: 4*2+1 = 9

Case	Input1	Input2	Expected output
TC_Login001	Valid Email	Valid Password	Login Successfully
TC_Login002	Invalid Email	Invalid Password	Login Failed
TC_Login003	Valid Email	Invalid Password	Login Failed
TC_Login004	Invalid Email	Valid Password	Login Failed
TC_Login005	Empty Email	Empty Password	Login Failed
TC_Login006	Valid Email	Empty Password	Login Failed
TC_Login007	Empty Email	Valid Password	Login Failed
TC_Login008	Invalid Email	Empty Password	Login Failed
TC_Login009	Empty Email	Invalid Password	Login Failed

2 Testing Results

We used php code for our project. For Unit testing we used PHPUnit platform(terminal based).

2.1 Test case for login method (Email)

```
4    use PHPUnit\Framework\TestCase;

5

6    require_once 'd:\Book\CST\seventh semester\darwin-cms-Lesson-13.UnitTesting\darwin-cms-Lesson-13.UnitTesting\src/interface
7    require_once 'd:\Book\CST\seventh semester\darwin-cms-Lesson-13.UnitTesting\darwin-cms-Lesson-13.UnitTesting\src/Validatio
8    require_once 'd:\Book\CST\seventh semester\darwin-cms-Lesson-13.UnitTesting\darwin-cms-Lesson-13.UnitTesting\src/validatio

9

10   final class ValidationTest extends TestCase

11   {

12       public function testValidationEmail(): void

13       {

14           $validationClass = new Validation();

15           $validationClass->addRule(new ValidateEmail());

16

17           $this->assertFalse($validationClass->validate('aty123'));

18           $this->assertTrue($validationClass->validate('test@email.com'));

19       }

20   }
```

ValidationTest ● testValidationEmail

Test Results   Terminal - ...ng/darwin-cms-Lesson-13.UnitTest...   ×

```
ERRORS!
Tests: 2, Assertions: 1, Errors: 1.
TAYBUL ISLAM TUSER@LAPTOP-HJCS2F47 /cygdrive/d/Book/CST/seventh semester/darwin-cms-Lesson-13.UnitTesting/darwin-cms-Lesson-13.UnitTesting
$ ./vendor/bin/phpunit tests
PHPUnit 9.5.27 by Sebastian Bergmann and contributors.

..                                                              2 / 2 (100%)

Time: 00:00.012, Memory: 4.00 MB

OK (2 tests, 3 assertions)

TAYBUL ISLAM TUSER@LAPTOP-HJCS2F47 /cygdrive/d/Book/CST/seventh semester/darwin-cms-Lesson-13.UnitTesting/darwin-cms-Lesson-13.UnitTesting
$ ▮
```

ID	Description	Input	Expected Output	Output	Result
201	Check the email for (valid email)	test@gmail.com	Login Successfully	Login Successfully	Successful
202	Check the email for (Invalid email)	Aty123	Login failed	Login failed	Successful

2.2 Test for login method (Password)

```
 9
10   final class ValidationTest extends TestCase
11   {
12       public function testValidationPassword(): void
13       {
14           $validationClass = new Validation();
15           $validationClass->addRule(new ValidatePassword());
16
17           $this->assertFalse($validationClass->validate('aty123'));
             $this->assertFalse($validationClass->validate('tuser'));
19       }
20   }
```

ValidationTest > testValidationPassword >

Test Results   Terminal - ...ng/darwin-cms-Lesson-13.UnitTest...   ×

```
TAYBUL ISLAM TUSER@LAPTOP-HJCS2F47 /cygdrive/d/Book/CST/seventh semester/darwin-cms-Lesson-13.UnitTesting/darwin-cms-Lesson-13.UnitTesting
$ ./vendor/bin/phpunit tests
PHPUnit 9.5.27 by Sebastian Bergmann and contributors.

..                                                          2 / 2 (100%)

Time: 00:00.012, Memory: 4.00 MB

OK (2 tests, 3 assertions)

TAYBUL ISLAM TUSER@LAPTOP-HJCS2F47 /cygdrive/d/Book/CST/seventh semester/darwin-cms-Lesson-13.UnitTesting/darwin-cms-Lesson-13.UnitTesting
$
```

ID	Description	Input	Expected Output	Output	Result
201	Check the password for (valid passwordl)	aty123	Login Successfully	Login Successfully	Successful
202	Check the password for (Invalid password)	tuser	Login failed	Login failed	Successful

2.3 Selenium Testing
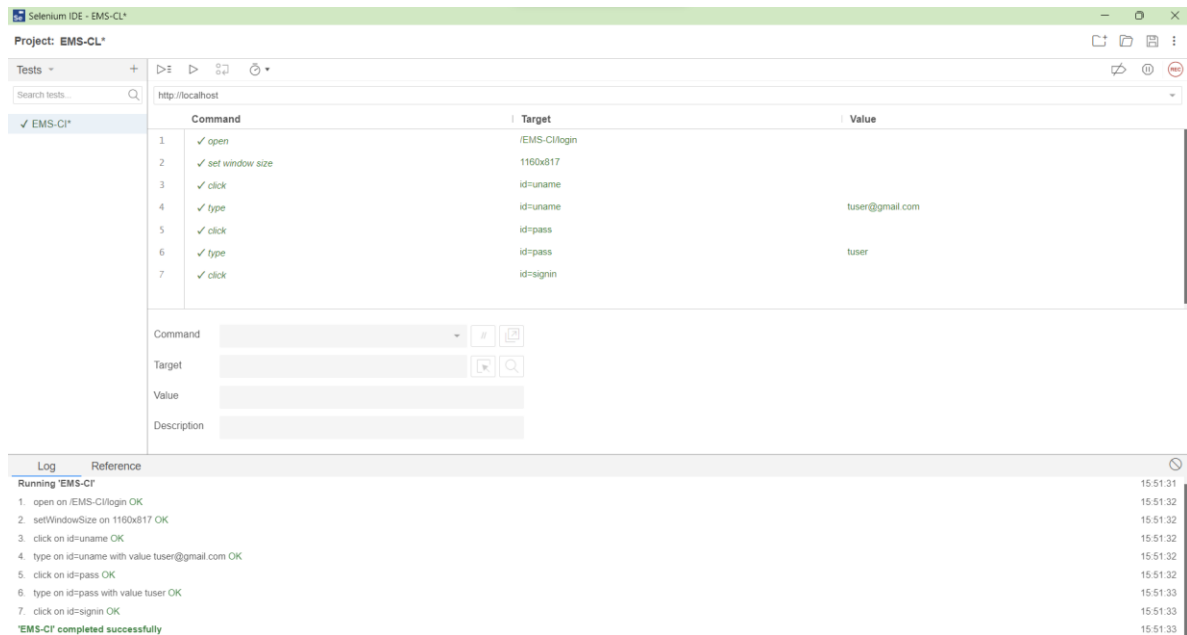
3 Results Analysis and Suggestions.
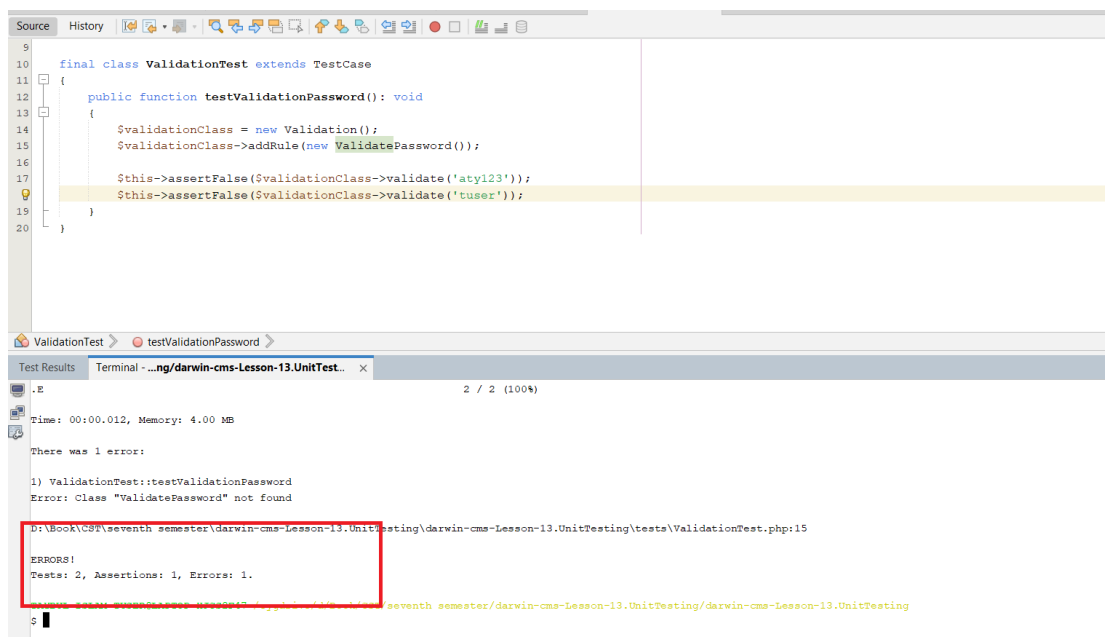
At first we got error while testing password functionality, because we used password hash method. We couldn't encrypted the password for testing, then we used normal password type and test again. Finally, we got successful testing report.

Input and output Screen of the project

You can follow some step to run our project

1. You have to install "XAMPP" and "MySQL" on your computer.

2. You need to start "Apache and MySQL" from XAMPP.

3. You need to put our project file on "C:\xampp\htdocs" folder.

4. You need to create a database name as "employeeigniter".

5. import the Database named "employeeigniter.sql" on employeeigniter database. You can find the   mysql database on "DATABASE FILE" folder on our project
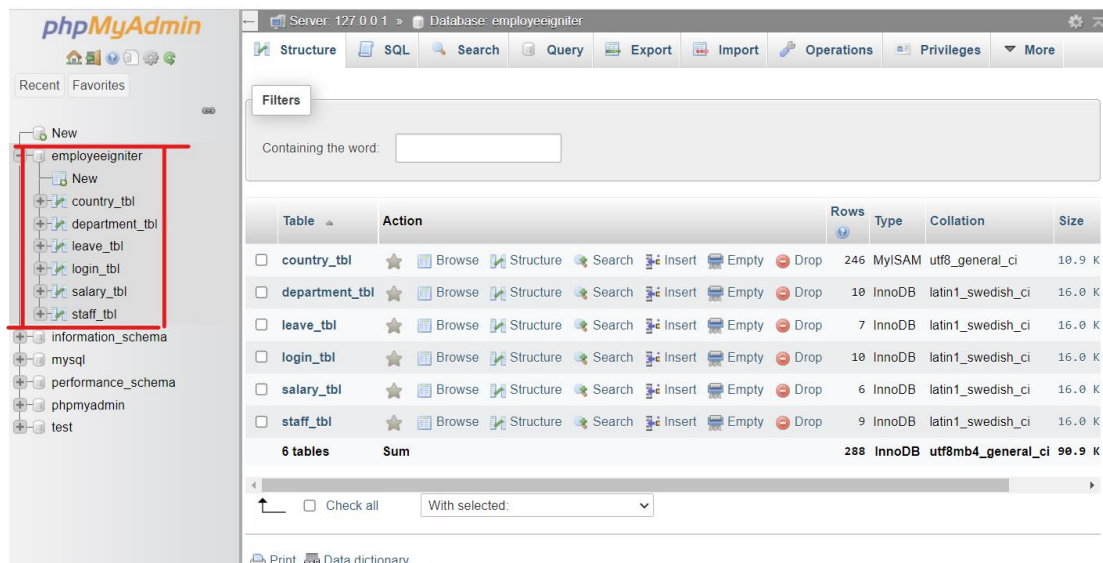
6. Then Enter the URL: http://localhost/EMS-CI/ on a browser.
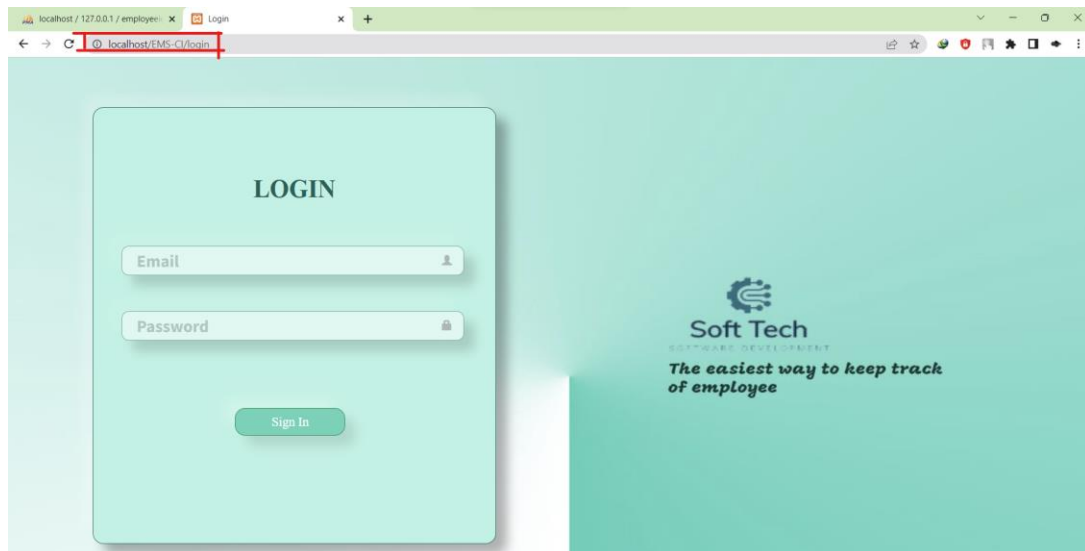
7. You need Login now by clicking on Sign In button.

8. We have two type of login account "Admin" & "Employee"

9. For Admin User: Email: tuser@gmail.com

Password: tuser

Email: rokaya@gmail.com

Password: rokaya

For Employee User: Email: tushar@gmail.com

Password: tushar

10. After login you can enter the dashboard

Admin Dashboard:
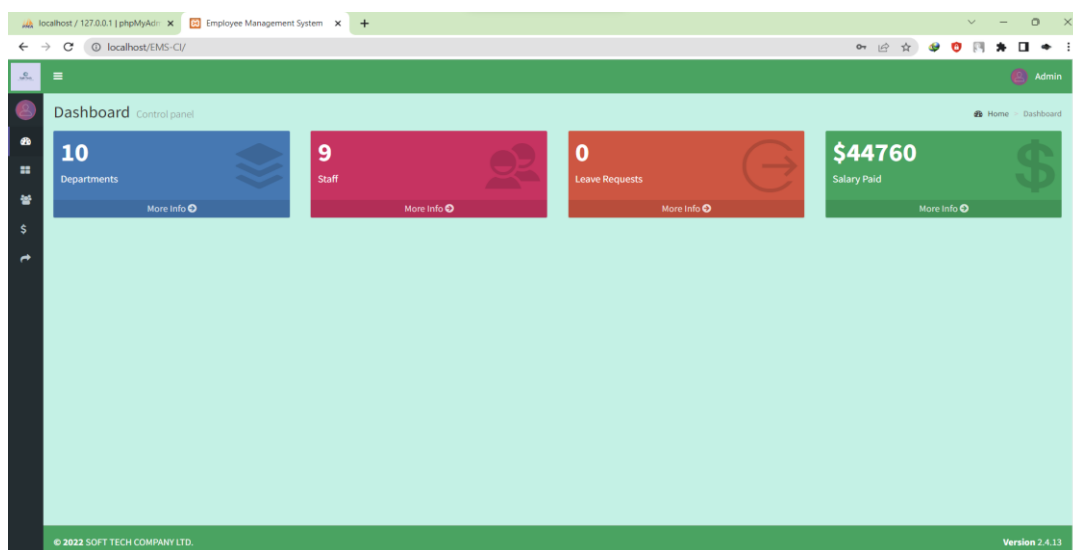
Employee Dashboard:
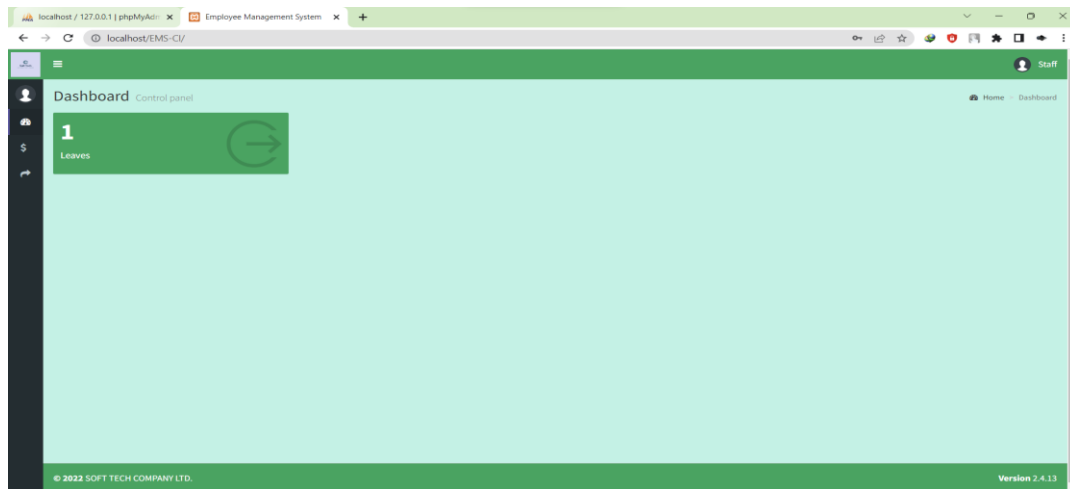
11. Admin Dashboard has Department, Staff, Salary, Leave entities. You can find these entities on the sidebar.
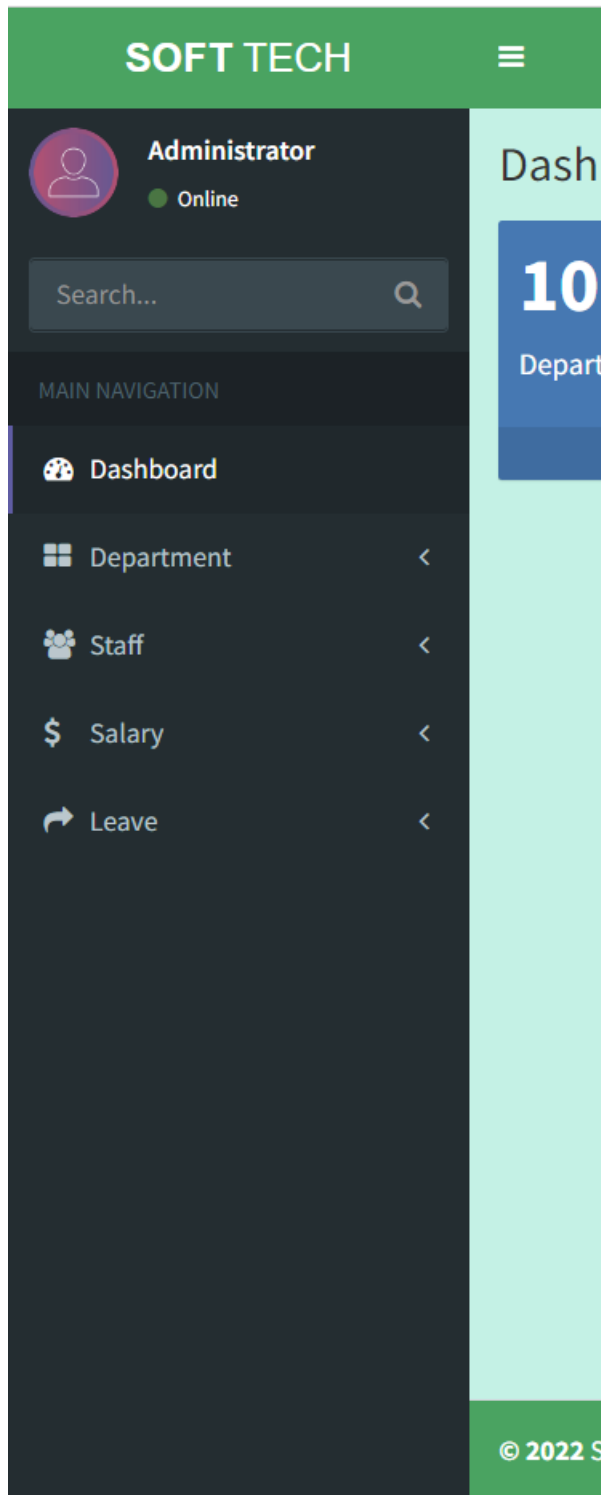
12. In the Department section you can add a new Department and Manage Department.
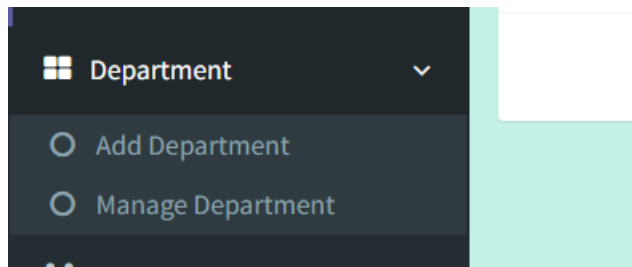
12.1. By Clicking the Add Department you can add a new Department.

After Clicking the submit button you show a Success alert.

12.2. By clicking the Manage Department you modify the pre added Department.(Edit & Delete)

13.1. By clicking the Staff you can find two sub Menu (Add Staff & Manage Staff).
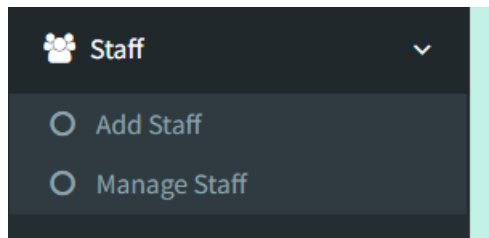
13.2. By clicking Add Staff you can add new employee.

After clicking the submit button you can show a success alert or failed alert.

13.3. By clicking the Manage Staff you can modify the pre created staff (edit & delete)

14. In Salary section you can add salary & Manage salary (edit & delete)

By Clicking the Manage Salary you can edit salary and delete salary

15. on the Leave section you can find Manage Staff's Leave request and Leave History.
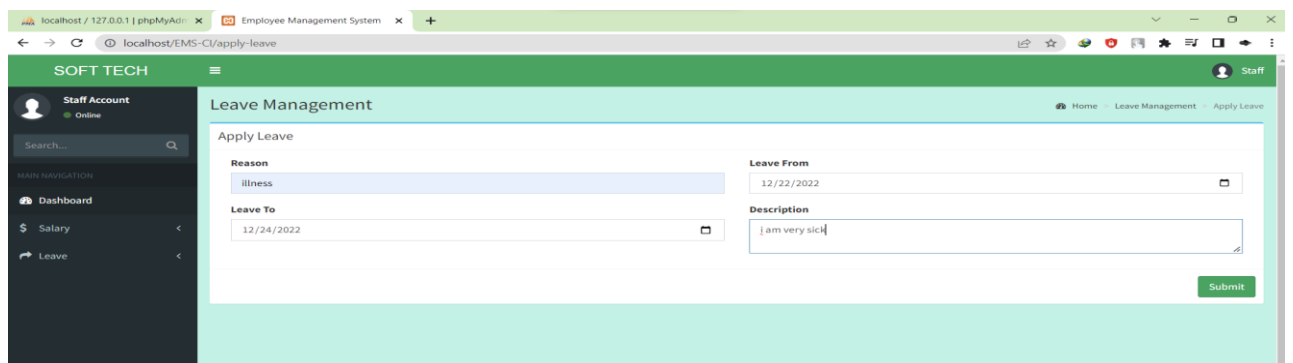

16. Employee User< you can Enter the Employee Dashboard.
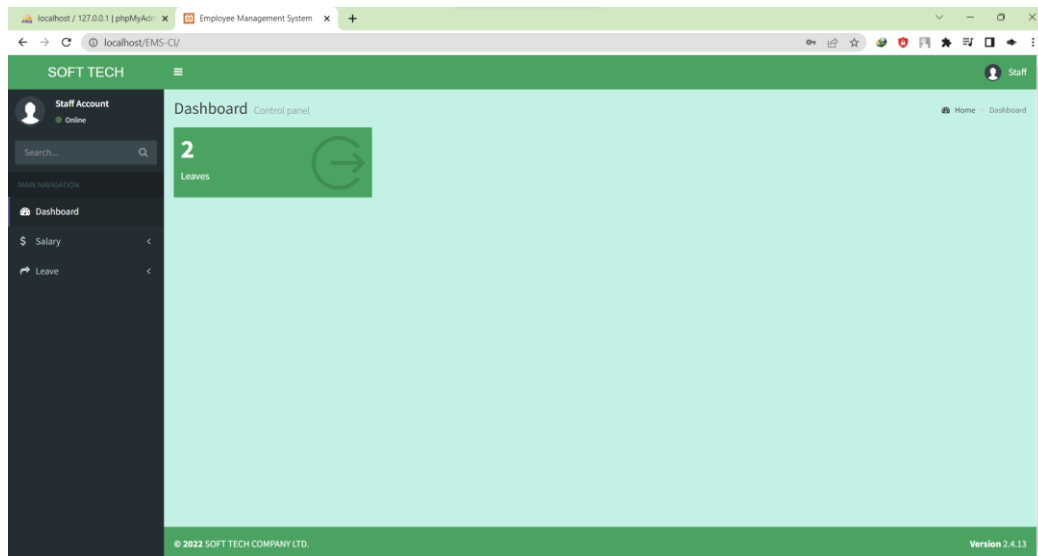
17. In the Salary section you can view you Salary.
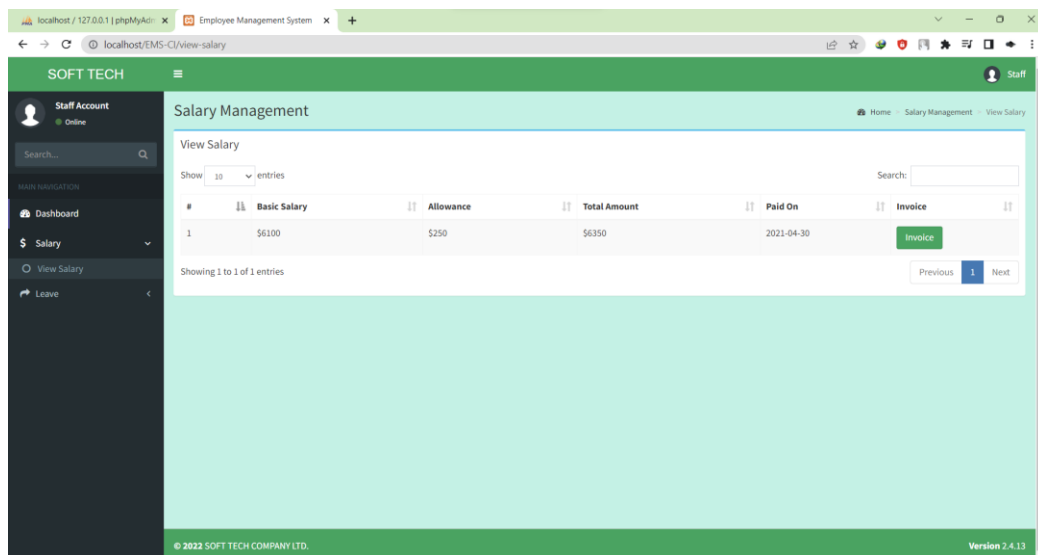
18. In the leave section employee can Apply for leave and find the leave history.

SOFT TECH ≡

Staff Account
● Online

Search... 🔍

MAIN NAVIGATION

🎛 Dashboard

$ Salary ‹

↪ Leave ⌄

○ Apply Leave
○ My Leave History

Leave Management

🏠 Home › Leave Management › View Leave

✔ **Success!**
New Leave Applied Succesfully

×

View Leave

Show 10 entries

Search:

#	Reason	From	To	Status	Description	Applied On
1	illness	22-12-2022	24-12-2022	Pending	i am very sick	21-12-2022
2	illness	15-12-2022	17-12-2022	Approved	Hello, I am ill today	13-12-2022
3	Casual Leave	28-05-2021	29-05-2021	Approved	been working for full hours since last month, got to free my mind for few days	27-05-2021

Showing 1 to 3 of 3 entries

Previous 1 Next