Software Test Plan

for

**ONLINE SCHOOL OPERATION**

Version 1.0 approved

Prepared by,

Table of Contents

[Revision History 3](#_bookmark0)

1. [TEST PLAN IDENTIFIER:RS-MTP01.3 4](#_bookmark1)
2. [REFERENCES 4](#_bookmark2)
3. INTRODUCTION

3.1 [Background to the Problem 4](#_bookmark4)

3.2 [Solution to the Problem 4](#_bookmark5)

1. [REQUEIREMNT SPECIFICATION 5](#_bookmark6)
   1. 4.1 [System Features 5](#_bookmark7)
   2. 4.2 [System Quality Attributes 7](#_bookmark8)
   3. 4.3 System Interface……………………………………………………………………………………..8
   4. 4.4 Project Requirement………………………………………………………………………………...10
2. [FEATURES NOT TO BE TESTED 12](#_bookmark9)
3. [TESTIN](#_bookmark10)G APPROACH

6.1 [Testing Levels 12](#_bookmark11)

6.2 [Test Tools 13](#_bookmark12)

6.3 [Meetings 15](#_bookmark13)

1. [TEST](#_bookmark14) CASES/ TEST ITEMS……………………………………………………………….16
2. ITEMS PASS/FAIL CITERIA 36
3. TESTING SCHEDULE 36

10. STAFFING AND TRAINING NEEDS……………………………………………………36

11. RESPONSIBILITIES ……………………………………………………………………...37

12. TESTING SCHEDULE…………………………………………………………………….37

13. PLANNING RISKS AND CONTINGENCIES…………………………………………...39

14. APPROVALS…………………………………………………………………………..........39

1. **TEST PLAN IDENTIFIER**:RS-MTP01.3
2. **REFERENCES**
   * https:/[/www.eskooly.com/](http://www.eskooly.com/)
   * Software Engineering Course Slide
   * Software Quality and Testing Course Slides
   * Software Requirement Specification (SRS) Document
3. **INTRODUCTION**

## 3.1 Background to the Problem

1. Student management is becoming a fundamental requirement in education in the present era, and it automates every daily task carried out in the school, college, and university. This technology allows us to swiftly gather all the data required for management. The primary goal is to develop software that will manage these many modules. What is the root cause of this problem? why is this problem is so important to consider?
2. Eskooly is the company we've chosen. The system can dynamically manage and save student, faculty, and teacher information. The suggested strategy provides the most straightforward method for managing all facets of student and university management. The program enables exploration of all internal activities that students are unaware of taking place at the school, college, or institution.

## 3.2 Solution to the Problem

The activities of students and teachers are provided by our system. With the use of this system, users may handle things like student information, grade data, registration, class schedules, notes, and payments. We are making sure that student data is secure. Our main offerings include

1. Result and Academic Transcript
2. Accounts
3. Online Admission
4. Messaging System
5. Students/Parents Login
6. Student Manager
7. We have a strong desire to offer first-rate service. Since ensuring customer satisfaction is our top priority, we also provide our training, operational management, professional skill, and recruitment divisions special attention.

Since ensuring customer satisfaction is our top priority, we also provide our training, operational management, professional skill, and recruitment divisions special attention.

1. Information on students, instructors, and teachers can be stored and managed by our business, Eskooly. Arena Phone Bd Ltd produced the cloud-based Eskooly application. It features SMS service and mobile apps. The use of Eskooly is free for educational institutions. It allows staff members to efficiently complete their regular duties and keep an eye on a variety of other institutional operations. In addition, parents are updated on their children's growth.
2. **REQUEIREMNT SPECIFICATION**

* 1. **4.1 System Features**

### Functional Requirements

### Login

The user login into the software with their username and password.

If the login successful the main home page in this software will be displayed.

* + - 1. If the user forgets the username and password so they can be reset password using their valid email address and phone number.
      2. If the user input 3 times wrong password, the system will block the user account login for half one hour.

### Priority level: High

**Precondition:** User have valid username and password.

### Cross-references: None

* + 1. **Admin**

### Functional Requirements

Monitoring user information by maintaining accurate database of all ID.

* + - 1. If the user is valid based on his/her information then “activate” card otherwise “reject” and report to the user.
      2. Admin will able to see the payment status. If paid then allow to request otherwise report to the user for payment. After reporting 2 times if user doesn’t

pay then admin can deactivate the user ID.

### Priority level: High

**Precondition:** Admin must have the rights to access the whole system.

**Cross-references:** 1.1, 1.2, 3.1

* + 1. **Accounts**

### Functional Requirements

* + - 1. The payment transaction will be done using user ID.
      2. The payment slip can be gotten from the account department using ID.
      3. Then the payment information will show in the portal.

### Priority level: High

**Precondition:** Payment system must be pay using user ID.

**Cross-references:** 2.1, 2.2, 2.3

* + 1. **Search**

### Functional Requirements

* 1. User can search for what service he/she expecting.
  2. System gives user review of their expecting service.
  3. User can also search for the information (Admission, Cost and Other details).

**Priority level:** Medium

**Precondition:** Must be in logged in condition

**Cross-references:** 5.1, 6.2

## 4.2 System Quality Attributes

There are some quality attributes that are very important to ensure the quality of a software.

**Usability:** Any skilled user should be able to register and login to the system.

**Efficiency:** Each and every functional need must be fulfilled.

**Portability:** This will be capable of running properly on all the devices.

**Maintainability:** If any issue locates or detect in the system then it will be possible to fix it.

**Correctness:** Mentioned all features will be completed according to the preferences of the patients.

**Functionality:** Will display available slot list, their details and it will also show if they are parking or not.

**Accessibility:** It’s a web-based software so it will be accessible from anywhere on the Internet.

**Readability:** It's critical to rely on appointment software to properly and containing accurate scheduling demands. It is so important to check to see if the system is durable enough to sustain any situation. So, regular counting on parking slot.

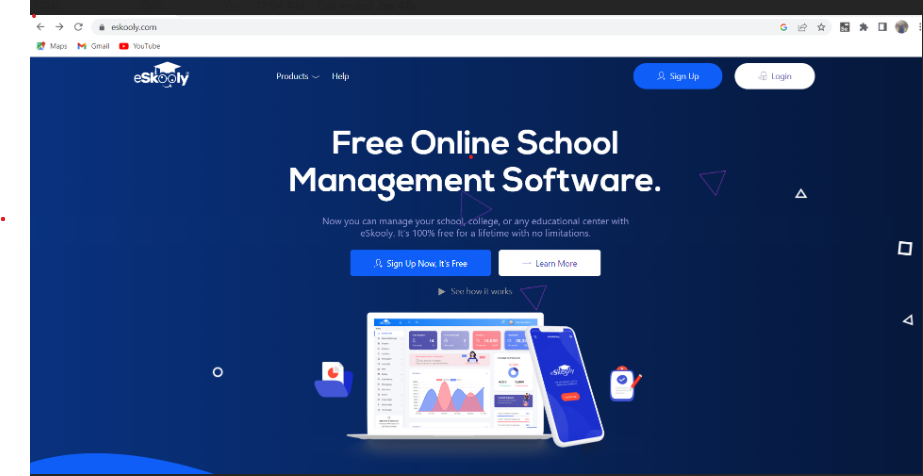
**Reliability:** All features will perform in various working environments or devices.

**Flexibility:** Will flexible enough to modify in terms of any needs.

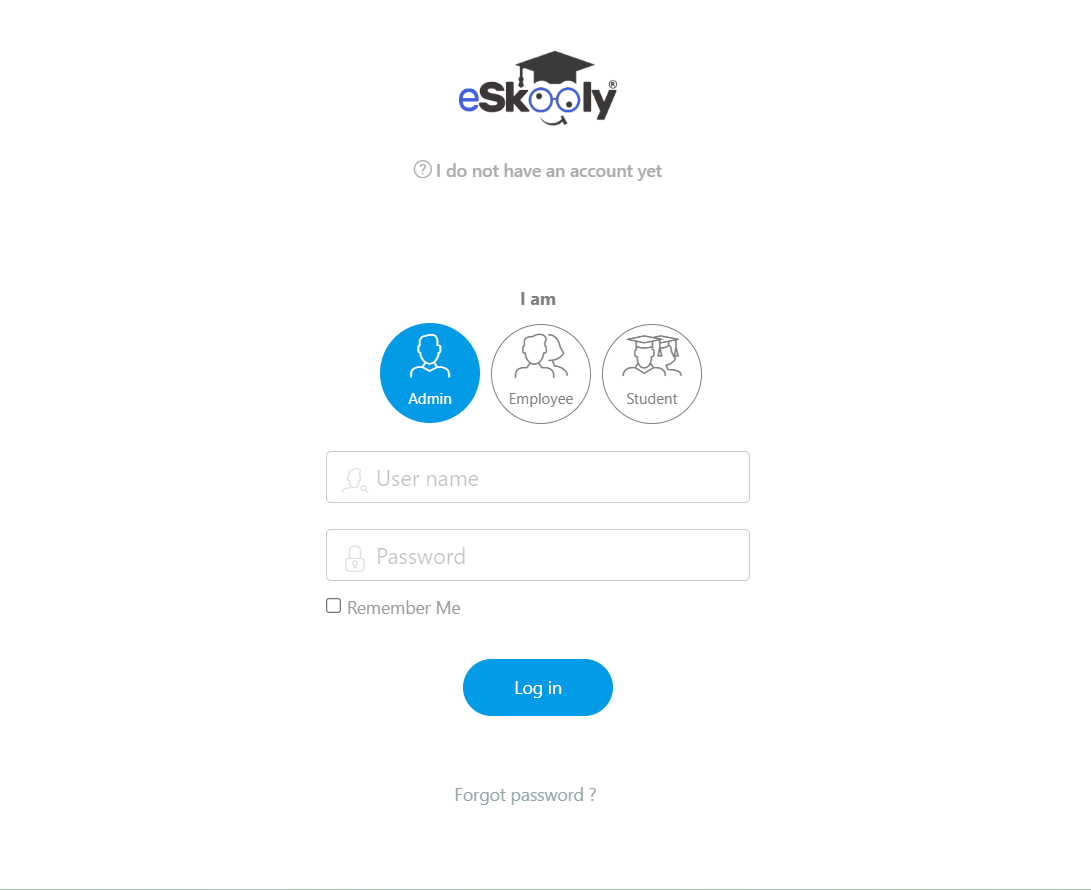
**Integrity/Security:** System integrity or security should be sufficient to prevent unauthorized access to system functions, information loss, and virus infection of software, as well as to protect the privacy of data entered into the system. Actually, Integrity comes with security

**4.3 System Interface**

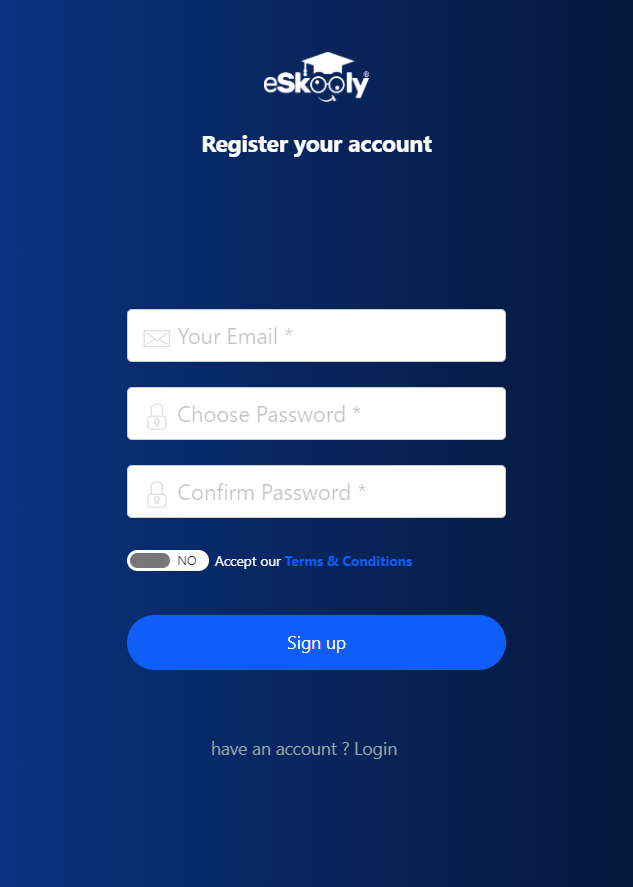
**Home Page:**

****

**Login:**

****

**Sign Up:**

****

**4.4 Project Requirement**

Effort Estimation:

The COCOMO (Constructive Cost Model) regression model may be supported by LOC, or lines of code. It is a procedural cost estimate model for software projects that is occasionally used as a method of accurately estimating the various factors that affect how a project will turn out, including size, effort, cost, time, and quality. The key parameters which indicate the quality of any software products that were made, which are also an outcome of the COCOMO are primarily Effort & Schedule of the project:

Effort: the quantity of work needed to complete a task. Units of measurement are person-months.

Schedule: simply refers to how long it takes to do a task, which is, of course, inversely related to the amount of effort made. It is measured in terms of time-based units like weeks and months.



The type of the developed product is Semi-detached. So, coefficient=2.4 P=1.05, T=0.38, SLOC=35000

* **Effort = PM = Coefficient<Effort Factor>\*(SLOC/1000) ^P**

=2.4\*(35000/1000) ^1.05

=104.83

### Development time = DM = 2.50\*(PM)^T

=2.50\*(104.83) ^0.38

= 14.65 ~ 15

### Required number of people = ST = PM/DM

**=**104.83/14.65

=7.16 ~ 8

Budget Estimation:

Duration in weeks = 15 weeks Week days = 5 days

Working hours = 8 Hours

Per week working hours is = (5\*8) hours

= 40 hours Total Working hours is = (40\*15) hours

= 600hours.

* + 1. **FEATURES NOT TO BE TESTED**

The following is a list of the areas that will not be specifically addressed. All testing in these

* + - Areas will be indirect as a result of other testing efforts. For example:
    - How to perform our application under low network that will not be tested.
    - External functionality over the program is not supported by this system. As a result,
    - Application to server testing should be avoided.
    - Browser compatibility
    1. **TESTING APPROACH**
       1. 6.1 Testing Levels

**UNIT Test**: The fundamental level of testing is the unit test. After creating each feature,a developer will verify that it functions as intended inside the component design. Because they are the ones who understand how these specific features functioned and because they are familiar with internal logic, program structure, etc., developers will perform this testing. In our application, the developer will conduct UNIT testing, and the development team leader will provide their approval. Before unit testing is accepted and handed off to

the tester, the programmer must show the team leader proof of it (test case list, sample output, data printouts, defect information). The test person will also receive access to all

unit test data.

**Integration Test**: The second level of testing is integration testing. the present level. The modules or features will be linked together one at a time. The leader of our development team will oversee this testing and determine whether or not the data transmission between these modules is accurate. At this level, we'll use strategies like the sandwich strategy, big bang approach, bottom-up integration, and top-down integration.

**System Test**: Our quality assurance team will carry out this testing level once the unit test and integration test have been completed. Our quality team will check the complete system against the customer's specification once our full program has been developed. These testing methods are known as black boxes. Various testing methods will have been used at this testing level. In addition to performing functional testing, our testing team also performs nonfunctional testing such as volume, load, and performance testing.

**Acceptance Test:** We will move on to the acceptance testing level once our entire application has been completed and the first three levels have been tested. At this stage,

the test team leader will work with the end user to evaluate our product. In essence, we will check the software's usability and functionality. We will ensure that our system satisfies all user requirements after this testing phase.

## 6.2 Test Tools

The ‘Selenium IDE’ – testing tool has been used to build test cased and test our website.

Selenium IDE has been used because it supports automated testing. We will perform manual testing initially, and then we will perform automation testing. The most crucial tool for automation testing is Selenium. Selenium is an open-source, free framework for assessing web applications across a variety of platforms and browsers. Scripts for Selenium tests can be written in a variety of programming languages, including Java, C#, Python, and others. However, we'll be automating our testing using Java. Below are a few screenshots of selenium.

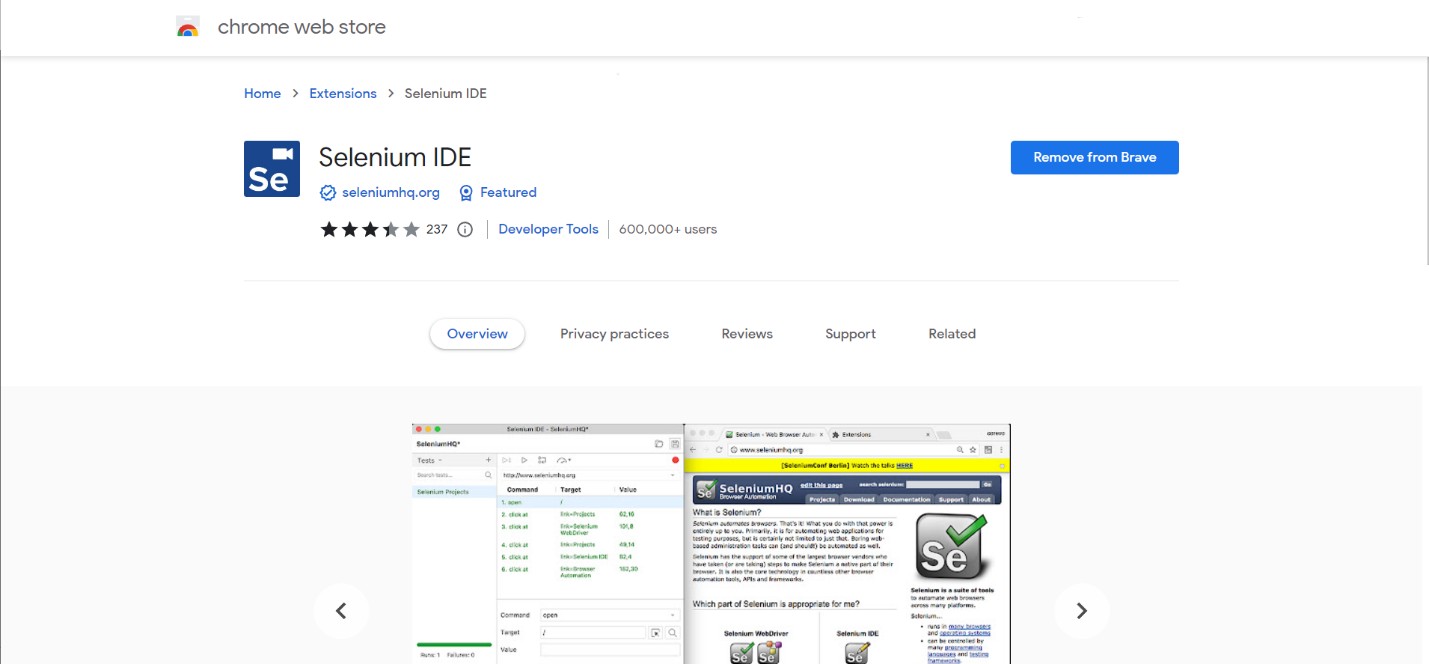


Fig-1

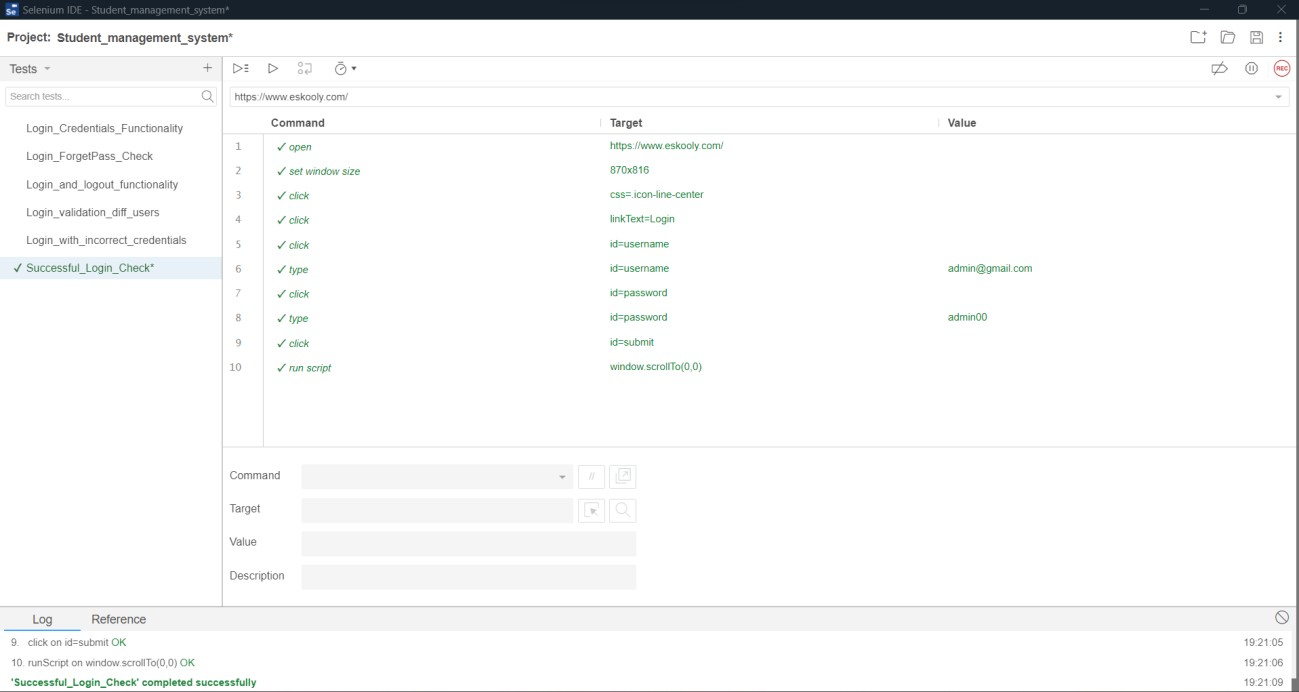


Fig-2

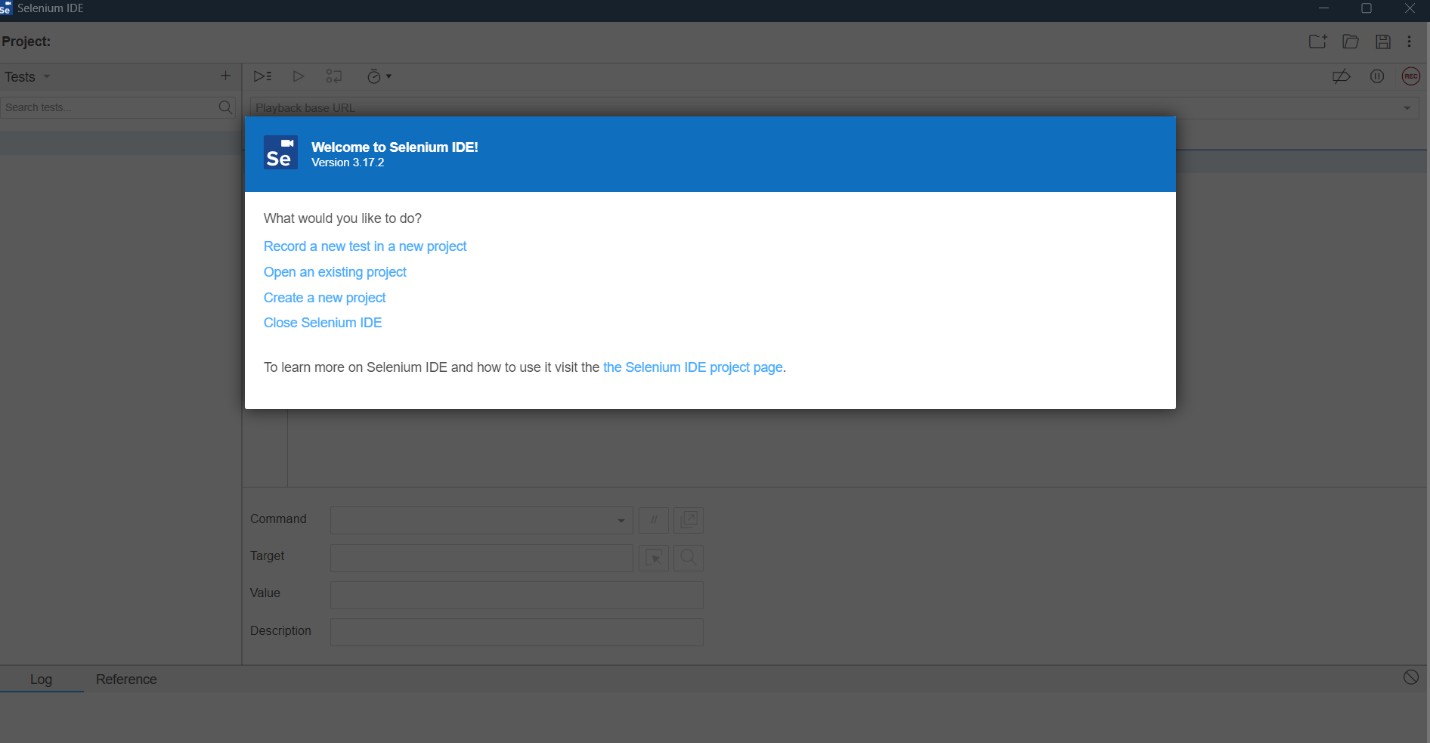


Fig-3

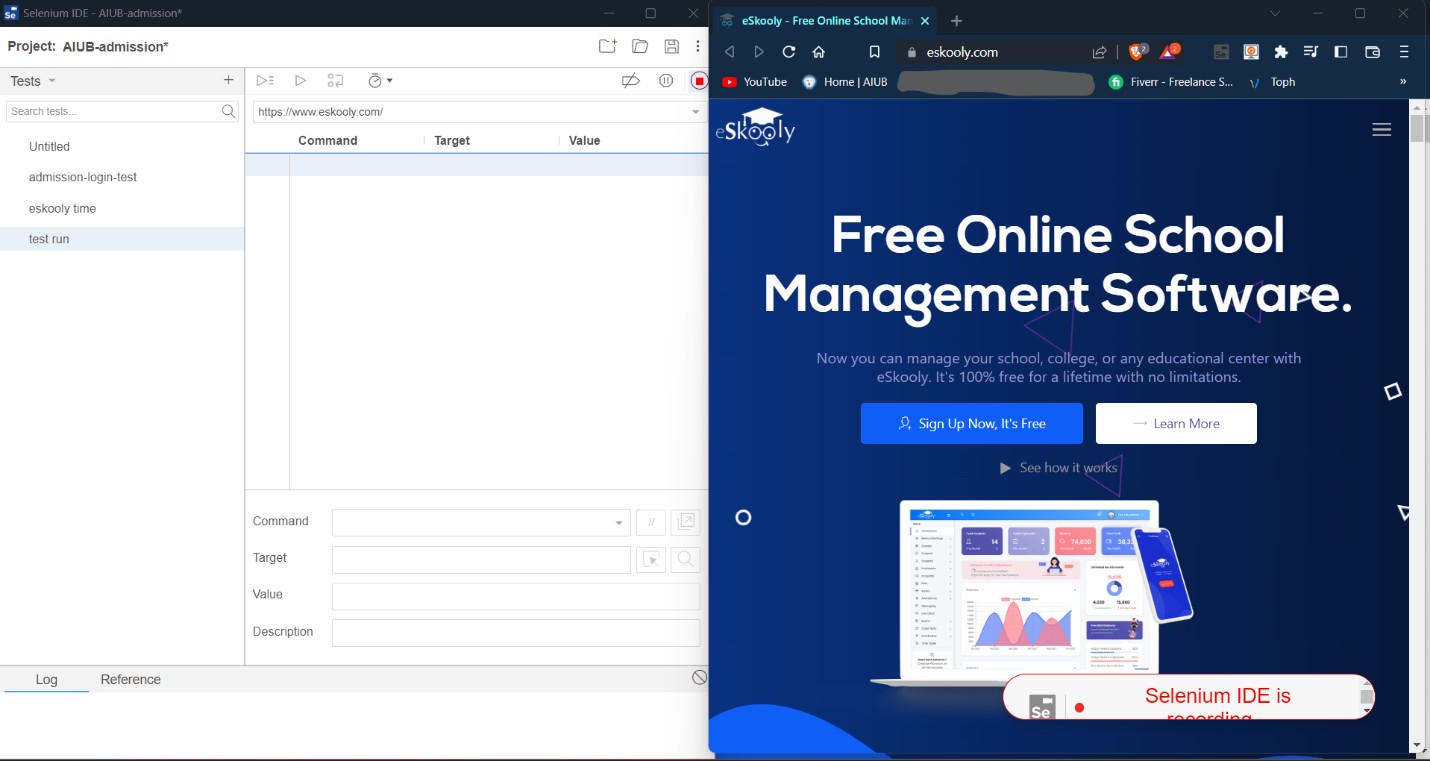


Fig-4

## 6.3 Meetings

In order to discuss about the project, we must go through group meetings. We used Microsoft teams to arrange our virtual meetings. We also organized some in person meeting at university. Before the group meeting we all prepared each of our stand alone plan for the project. In the meeting we had decided to follow the best plan. We equally shared our work in this project. The meetings were often organized to keep updating our work

* + 1. **TEST CASES/TEST ITEMS**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nafisa Sultana | | |
| Test Case ID: FR\_1 | | | Test Designed date: 23/04/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nafisa Sultana | | |
| Module Name: Registration Session | | | Test Execution date: 24/04/2023 | | |
| Test Title: Successful Registration Check | | |  | | |
| Description: Test website’s Registration Page and storing username and password to the database. | | |  | | |
| Precondition (If any): Must land on sign up page from website homepage successfully. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Click on signup 3. Enter E-mail 4. Enter password 5. Enter Confirm password 6. Accept terms and condition button 7. Click signup button | E-mail: [nafisasultana@gmail.com](mailto:nafisasultana@gmail.com)  Password: 258963  Confirm Password: 258963 | User should register successfully to the website | | As expected, | Pass |
| Post Condition: User is registered with database and can be log in later. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nafisa Sultana | | |
| Test Case ID: FR\_2 | | | Test Designed date: 23/04/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nafisa Sultana | | |
| Module Name: Registration input field validation | | | Test Execution date: 23/04/2023 | | |
| Test Title: Registration Field Validation Check | | |  | | |
| Description: Test website’s Registration Field | | |  | | |
| Precondition (If any): Must land on sign up page. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website’s signup page 2. Skip E-mail’s field 3. Enter password 4. Enter confirm password 5. Accept terms and condition button 6. Click signup button 7. Clear the confirm password input field 8. Enter Confirm password 9. Enter E-mail 10. Didn’t click terms and condition button 11. Click signup button | E-mail:  [aiub@gmail.com](mailto:aiub@gmail.com)  Password:  12121212  Confirm Password: 12121212 | User should  not register successfully to the website and must show a message | | Result not as expected, | Fail |
| Post Condition: User isn’t registered with database and can’t be log in later. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nafisa Sultana | | |
| Test Case ID: FR\_3 | | | Test Designed date: 24/04/2023 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Nafisa Sultana | | |
| Module Name: Registration input field functionality 1.0 | | | Test Execution date: 24/04/2023 | | |
| Test Title: Registration’s E-mail Pattern Validation Checking | | |  | | |
| Description: Testing username field functionality as an e-mail’s pattern validation | | |  | | |
| Precondition (If any): Must land on sign up page. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website’s signup page 2. Enter E-mail 3. Enter password 4. Enter confirm password 5. Accept terms and condition button 6. Click signup button | E-mail: sqtgmail  Password: Jakia123  Confirm Password: Jakia123 | User should not register successfully to the website | | Result not as expected, | Fail |
| Post Condition: User can’t be registered. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nafisa Sultana | | |
| Test Case ID: FR\_4 | | | Test Designed date: 25/04/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nafisa Sultana | | |
| Module Name: Registration input field functionality 1.1 | | | Test Execution date: 25/04/2023 | | |
| Test Title: signup password/confirm password input field validation | | |  | | |
| Description: Testing password/confirm password field in signup page | | |  | | |
| Precondition (If any): Must land on sign up page. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website’s signup page 2. Enter E-mail 3. Enter password 4. Enter confirm password 5. Accept terms and condition button 6. Click signup button | E-mail: [Prova@gmail.com](mailto:Prova@gmail.com)  Password: 25698  Confirm Password: 25478 | User should not register successfully to the website | | Result not as expected, | Fail |
| Post Condition: User isn’t registered with database and can’t be log in later. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nabila Chowdhury Joya | | |
| Test Case ID: FR\_5 | | | Test Designed date: 24-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nabila Chowdhury Joya | | |
| Module Name: Login Session Admin | | | Test Execution date: 24-04-2023 | | |
| Test Title: Successful Login Check | | |  | | |
| Description: Test website login page for user- Admin | | |  | | |
| Precondition (If any): User must have valid username and password | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Go to login page 3. Enter admin username 4. Enter admin password 5. Click submit | Username: [admin@gmail.com](mailto:admin@gmail.com)  Password: admin00 | User should login into the application | | As expected, | Pass |
| Post Condition: User is validated with database and successfully login to account. The account session details are logged in the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nabila Chowdhury Joya | | |
| Test Case ID: FR\_6 | | | Test Designed date: 23-03-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nabila Chowdhury Joya | | |
| Module Name: Login Validation 1.0 | | | Test Execution date: 23-03-2023 | | |
| Test Title: Login with Incorrect Credentials | | |  | | |
| Description: Checking for log into admin account with incorrect username and password | | |  | | |
| Precondition (If any): Must be on the Login page. Username and password must be registered in the database before checking | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Go to login page 3. Enter incorrect admin username 4. Enter incorrect admin password 5. Click submit | Username: [nabila234@yahoo.com](mailto:nabila234@yahoo.com)  Password: 789456123 | User shouldn’t be log in to different accounts | | As expected, | Pass |
| Post Condition: Must be shown Incorrect username and password after login | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nabila Chowdhury Joya | | |
| Test Case ID: FR\_7 | | | Test Designed date: 24-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nabila Chowdhury Joya | | |
| Module Name: Login Validation 1.1 | | | Test Execution date: 24-04-2023 | | |
| Test Title: Login Validation Different Users | | |  | | |
| Description: Checking for log into different users with same username and password | | |  | | |
| Precondition (If any): Username and password must be registered in the database before checking | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Go to login page 3. Enter admin username into employee username 4. Enter admin password into employee password 5. Click submit | Username: [admin@gmail.com](mailto:admin@gmail.com)  Password: admin00 | User shouldn’t be log in to employee accounts | | As expected, | Pass |
| Post Condition: User must have different username and password for log into different accounts. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nabila Chowdhury Joya | | |
| Test Case ID: FR\_8 | | | Test Designed date: 25-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nabila Chowdhury Joya | | |
| Module Name: Login and Logout Session | | | Test Execution date: 25-04-2023 | | |
| Test Title: Login and Logout Functionality | | |  | | |
| Description: Checking the login and logout functionality working successfully | | |  | | |
| Precondition (If any): Must be on the Login page. Username and password must be correct. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Go to the website 2. Go to login page 3. Enter correct username 4. Enter correct password 5. Click submit 6. Click logout | Username: [Teacher@yahoo.com](mailto:Teacher@yahoo.com)  Password: teacher1 | User should be able login and logout successfully | | As expected, | Pass |
| Post Condition: Same as precondition | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nuzhat Tanzina Prova | | |
| Test Case ID: FR\_9 | | | Test Designed date: 26-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nuzhat Tanzina Prova | | |
| Module Name: Employee addition feature | | | Test Execution date: 27-04-2023 | | |
| Test Title: Add employee with valid input | | |  | | |
| Description: Checking the addition of a new employee by providing the valid information in both required and optional section of the employee form | | |  | | |
| Precondition (If any): Must be landed on the dashboard page. Username must be registered to database. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click Add new button 5. Enter employee name 6. Enter mobile no 7. Enter joining date 8. Enter employee-type 9. Enter monthly salary 10. Enter father name 11. Enter CNIC 12. Enter education 13. Enter gender 14. Enter religion 15. Enter blood group 16. Enter experience 17. Enter email 18. Enter date of birth 19. Enter address 20. Click on submit | Name: Prova  Mob no: 01825984369  Joining-date:2021-11-11  Employee-type: Teacher  Monthly-salary:10000 Father-name: Adnan  CNIC:3123  Education: AIUB Gender: Male Religion: Islam  Blood group: A+ | User must be registered as an employee | | As expected, | Pass |
|  | Experience: Teaching |  | |  |  |
|  | [Email:Prova@gmail.com](mailto:sajid@gmail.com) |  | |  |  |
|  | Date of birth:2000-12-1 |  | |  |  |
|  | Address:Uttara Bangladesh |  | |  |  |
| Post Condition: The registered user got a new role as an employee in the website. The data must be stored into the database | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nuzhat Tanzina Prova | | |
| Test Case ID: FR\_10 | | | Test Designed date: 27-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nuzhat Tanzina Prova | | |
| Module Name: Employee form validation 1.0 | | | Test Execution date: 27-04-2023 | | |
| Test Title: Add employee with invalid input | | |  | | |
| Description: Checking the addition of a new employee by providing the invalid information in required section of the employee form | | |  | | |
| Precondition (If any): Must be landed on the employee form page during the addition of the new employee. Username must be registered to database. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click Add new button 5. Enter employee name 6. Enter mobile no 7. Enter joining date 8. Enter employee-type 9. Enter monthly salary 10. Click on submit | Name: 1234  Mob no: 01644412678  Joining-date:2020-12-1 Employee-type: other Monthly-salary: -9000  Father-name: Null | Employee should not able to be registered. | | Not as expected, | Fail |
|  | CNIC: Null |  | |  |  |
|  | Education: Null |  | |  |  |
|  | Gender: Null |  | |  |  |
|  | Religion: Null |  | |  |  |
|  | Blood group: Null |  | |  |  |
|  | Experience: Null |  | |  |  |
|  | Email: Null |  | |  |  |
|  | Date of birth: Null |  | |  |  |
|  | Address: Null |  | |  |  |
| Post Condition: The registered user will not able to add as an employee. A warning must be showed for the invalid name and the invalid salary | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nuzhat Tanzina Prova | | |
| Test Case ID: FR\_11 | | | Test Designed date: 28-04-22 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nuzhat Tanzina Prova | | |
| Module Name: Employee form validation 1.1 | | | Test Execution date: 28-04-22 | | |
| Test Title: Add employee with invalid mobile no | | |  | | |
| Description: Checking the addition of a new employee by providing the invalid mobile no in the required section of the employee form | | |  | | |
| Precondition (If any): Must be landed on the employee form page during the addition of the new employee. Username must be registered to database. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click Add new button 5. Enter employee name 6. Enter mobile no 7. Enter joining date 8. Enter employee-type 9. Enter monthly salary 10. Click on submit | Name: Invalid phone, String number  Mob-no: Null, String\_) (\*&^&^^%  Joining-date: 1999-12-1,  2010-12-1  Employee-type: Store manager, Other  Monthly-salary: 3000,2000  Father-name: Null CNIC: Null Education: Null Gender: Null Religion: Null Blood group: Null Experience: Null Email: Null  Date of birth: Null  Address: Null | Employee should not able to be registered. | | Not as expected, | Fail |
| Post Condition: The registered user will not able to add as an employee and not add the information to the database. A warning need to be showed for the null input in the required section and the invalid string input as the mobile number. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Nuzhat Tanzina Prova | | |
| Test Case ID: FR\_12 | | | Test Designed date: 28-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nuzhat Tanzina Prova | | |
| Module Name: Employee form validation 1.2 | | | Test Execution date: 28-04-2023 | | |
| Test Title: Add employee without other required input | | |  | | |
| Description: Checking the addition of a new employee without providing the inputs in the required section of the employee form | | |  | | |
| Precondition (If any): Must be landed on the employee form page during the addition of the new employee. Username must be registered to database. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click Add new button 5. Land on the employee form section 6. Enter mobile number 7. Enter employee name, employee type and salary keeping joining date empty 8. Enter joining date, employee type and salary keeping employee name empty 9. Enter employee name, joining date, and salary keeping employee type empty 10. Enter employee name, joining date, employee type keeping salary empty 11. Click on submit | Name: test\_req, Null Mob-no: 01824500111  Joining-date: Null, 2019-  12-1  Employee type: Other, Null  Monthly-salary: 1000,Null  Father-name: Null CNIC: Null Education: Null Gender: Null Religion: Null Blood group: Null Experience: Null Email: Null  Date of birth: Null Address: Null | Employee should not able to be registered. | | As expected, | Pass |
| Post Condition: The blank textboxes in required information must show the warning and new employee is not registered or not added to the database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Jakia Noor | | |
| Test Case ID: FR\_13 | | | Test Designed date: 29-04-2023 | | |
| Test Priority (Low, Medium, High): Medium | | | Test Executed by: Jakia Noor | | |
| Module Name: Employee form validation 1.3 | | | Test Execution date: 29-04-2023 | | |
| Test Title: Add employee with invalid optional email | | |  | | |
| Description: Checking the addition of a new employee providing the invalid email in the optional section of the employee form. | | |  | | |
| Precondition (If any): Must be landed on the employee form page during the addition of the new employee and must provide the valid information in the required section. Username must be registered to database. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click Add new button 5. Enter employee name 6. Enter mobile no 7. Enter joining date 8. Enter employee-type 9. Enter monthly salary 10. Enter email 11. Click on submit | Name:Invalid mail Mob-no:01644412678  Joining-date:2022-12- 16  Employee type: Other Monthly-salary: 1000 Father-name: Null CNIC: Null Education: Null Gender: Null Religion: Null  Blood group: Null Experience: Null Email: asif\_gmail.com Date of birth: Null  Address: Null | Employee cannot able to be registered. | | As expected, | Pass |
| Post Condition: The invalid email in the textbox showed warning and the user is not registered as an employee when the submit button is clicked. The information must not store into the database as well. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Project Name: Student Management System | | | Test Designed by: Jakia Noor | | |
| Test Case ID: FR\_21 | | | Test Designed date: 29-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Jakia Noor | | |
| Module Name: Employee edit and delete feature | | | Test Execution date: 29-04-2023 | | |
| Test Title: Employee edit and delete check | | |  | | |
| Description: Checking the deletion of an existing employee and editing the internal data of an employee | | |  | | |
| Precondition (If any): User must be an existing employee and the data must be in the database for the deletion and the edit operation. | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1. Land to the dashboard section 2. Go to Employees in menu bar 3. Click on All Employees 4. Click the edit button 5. Enter employee name 6. Click on update 7. Enter email address 8. Click on update 9. Click the bin/delete button to delete an existing employee. | Name: Rahman Mob-no: Null Joining-date: Null  Employee type: Null Monthly-salary: Null Father-name: Null CNIC: Null Education: Null Gender: Null Religion: Null  Blood group: Null  Experience: Null  Email: [rahman@gmail.com](mailto:rahman@gmail.com)  Date of birth: Null Address: Null | Employee data should be updated after edit and deleted after the deletion | | As expected, | Pass |
| Post Condition: The data in the database must be updated after the edit. If the employee is deleted from the all employee section then the data of the employee must be removed. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID: EMP\_1 | | | | Test Designed date: 30-04-2023 | |
| Test Priority (Low, Medium, High): HIGH | | | | Test Executed by: Jakia Noor | |
| Module Name: Available ALL Employee | | | | Test Execution date: 30-04-2023 | |
| Test Title: Searching all available employee by admin account | | | |  | |
| Description: Test Eskooly admin searching all employee page | | | |  | |
| Precondition (If any): Admin must have log in by username and password | | | | | |
| Test Steps | Test Data | Expected Results | Actual Results | | Status (Pass/Fail) |
| 1. Go to website 2. Click Employees 3. Click All   Employee   1. Search in the search box | Search By:  Nafisa/Joya/Prova/Jakia | Admin could search all the employees | As expected, | | Pass |
| Post Condition: The employees must be shown after searching. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID: CLC\_1 | | | Test Designed date: 30-04-2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Jakia Noor | | |
| Module Name: Class create | | | Test Execution date: 30-04-2023 | | |
| Test Title: Creating a class by admin account | | |  | | |
| Description: Test the Create classes page | | |  | | |
| Precondition (If any): Admin must have log in by username and password | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1.Go to Website 2.Click Classes 3.Click New class  4. Type class name, tuition fees,  Assign class teacher | Name of the Class: Class 9/Class 10  Monthly Tuition Fees:1500  Select Class Teacher: Nafisa/Joya/Prova/Jakia | Admin can create class | | As expected | Pass |
| Post Condition: Admin successfully added the class and the information saved in database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID: AS\_1 | | | Test Designed date:01/05/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nafisa Sultana, Nabila Chowdhury Joya | | |
| Module Name: Assign Subject | | | Test Execution date: 01/05/2023 | | |
| Test Title: Assigning subject by admin account | | |  | | |
| Description: Test the Subject page | | |  | | |
| Precondition (If any): Admin must have log in by username and password and Class must have been created | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1.Go to Website 2.Click Subjects 3.Click Assign Subjects  4. Select Class, Name of the subject and marks. | Select Class: Class 9/Class 10  Name of Subject: Physics/Math  Marks: 100 | Admin submit successfully | | As expected, | Pass |
| Post Condition: Admin successfully added the subject and the information saved in database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID: ADS\_1 | | | Test Designed date:01/05/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Nuzhat Tanzina Prova | | |
| Module Name: Add Student | | | Test Execution date: 01/05/2023 | | |
| Test Title: Adding student by admin account | | |  | | |
| Description: Test the add student page | | |  | | |
| Precondition (If any): Admin must have log in by username and password and Class and Subject must have been created | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1.Go to Website 2.Click Students 3.Click add new  4.Fill up all the required textbox | Name of Student: Any Name.  Registration Number: not registered number  Select Class: Any available class  Admission Date: Admission date | Admin added student successfully | | As expected, | Pass |
| Post Condition: Admin successfully added student and the information saved in database. | | | | | |

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Test Case ID: CR\_1 | | | Test Designed date: 01/05/2023 | | |
| Test Priority (Low, Medium, High): High | | | Test Executed by: Jakia Noor | | |
| Module Name: Student | | | Test Execution date: 01/05/2023 | | |
| Test Title: Verifying student registration number. | | |  | | |
| Description: Test two page of one student to check registration number is same | | |  | | |
| Precondition (If any): Admin must have log in by username and password and Student must have been created | | | | | |
| Test Steps | Test Data | Expected Results | | Actual Results | Status (Pass/Fail) |
| 1.Go to Website 2.Click Students 3.Click All Students   1. Check registration number of a student 2. Go to that student report page 3. Check registration No 7.Match both pages   Registration no | Reg No: 12202127 | Both pages has to be the same registration number of a student | | As expected, | Pass |
|  | | | | | |

* + 1. **ITEM PASS/FAIL CRITERIA**
* Unit test done in each and every module or feature
* All the modules added one by one and integration test done after every module integrated
* 100% integration test passed
* No major defects are outstanding
* Not more than 15 minor defects are outstanding
* Code coverage tools indicates all code covered
* Ensuring all critical Test Cases are passed
* Identifying and fixing all the high-priority defects
  + 1. **TESTING DELIVERABLES**
* **Acceptance test plan:** The user acceptability tests all turned out to be successful. The user interface was easy to use but still efficient.
* **System/Integration test plan:** System integration was completed satisfactorily in every way. The database was therefore correctly functioning and all of its features were responsive.
* **Unit test plans/turnover documentation:** The unit testing was done successfully.
* **Screen prototypes:** Several prototypes were made and
* **Report mock-ups:** There were no mock-ups created for the report we are currently reviewing because it is the project report.
* **Defect/Incident reports and summaries:** Tt functions well in any circumstances but the requirements are not fully understandable. The process is iterative and based on trial.
* **Test logs and turnover reports:** All of the tests were completed correctly, and none revealed any major problems.
  + 1. **STAFFING AND TRAINING NEEDS**

The following staffing is expected for this test plan:

* + Project Manager – 1
  + Test Manager – 1
  + Developer – 2

If the Project Manager is not familiar with software testing, project manager should become

Familiar every testing processes. The Test Manager and Test Analyst must be familiar with how to use the test software. Many software systems come with extensive on-line help, but the test personnel may also need detailed knowledge of the testing criteria standards to fully perform their duties. Developers need more effort to develop software.

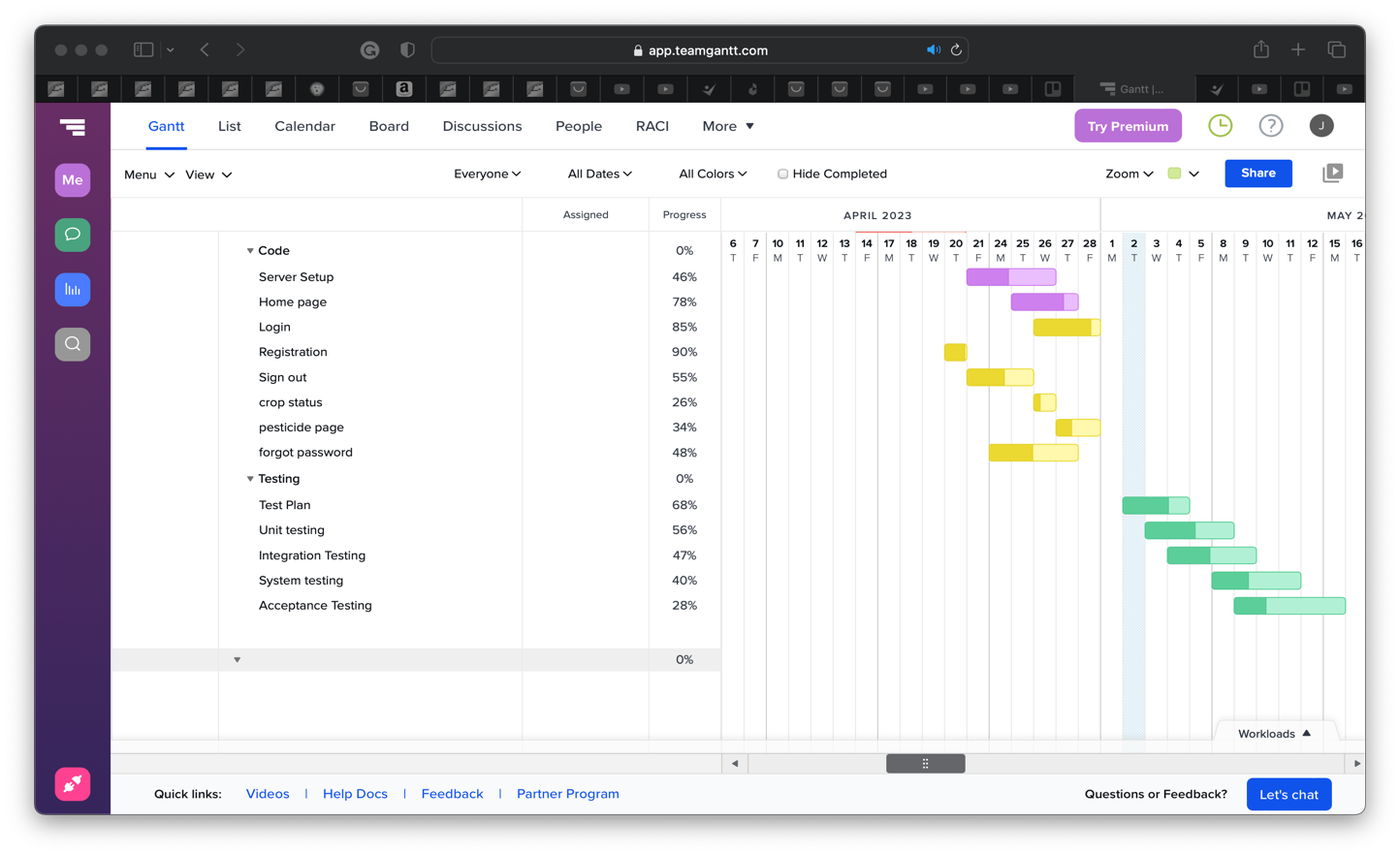
* + 1. **RESPONSIBILITIES**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
|  | **Joya** | **Nafisa** | **Jakia** | **Prova** |
| Acceptance test documentation and execution | **×** |  |  | **×** |
| System/Integration test documentation and execution |  | **×** | **×** |  |
| Unit test documentation and execution | **×** | **×** |  |  |
| System design reviews | **×** |  | **×** |  |
| Detail design reviews |  |  | **×** | **×** |
| Test Procedures and Rules |  | **×** |  | **×** |
| Screen and report prototype reviews | **×** |  | **×** |  |
| Change control and regression testing |  | **×** |  | **×** |

* + 1. **TESTING SCHEDULE**

The project schedule includes time for the upcoming testing activities. The precise hours and dates for each task are listed in the project plan timetable. The project schedule and strategy also detail the personnel requirements for each step. The project manager will coordinate the personnel required for each task, the test team, the development team, management, and the client in collaboration with the leaders of the development and test team

### GANTT CHART



* + 1. **PLANNING RISKS AND CONTINGENCIES**

|  |  |  |  |
| --- | --- | --- | --- |
| Risk | Probability | Impact | Mitigation |
| Error in function | Medium | Medium | Test the web app frequently and  maintain daily  backup. |
| Give invalid input | High | High | Tell user to use right data type in each input field |
| Loss of encrypted data(password) | Medium | High | Maintain security check and backup |

* + 1. **APROVALS**

|  |  |
| --- | --- |
| Project Sponsor | Pass |
| Development Management | pass |
| EDI Project Manager | Pass |
| Rs Test Manager | Pass |
| RS Development Team Manager | Pass |
| Reassigned sales | Pass |
| Order Entry EDI Team Manager | Pass |