**Project Report**

|  |  |
| --- | --- |
| **Product Name** | Applied Degree in Software Engineering |
| **Qualification Name (NICF)** | Applied Degree in Software Engineering |
| **Product Name** | Design, develop and test the Community portal website |
| **Module Name (NICF)** | NICF-Application Implementation |

|  |  |  |  |
| --- | --- | --- | --- |
|  | | **Assessor name** | |
|  | |  | |
| **Date issued** | **Completion date** | | **Submitted on** |
|  |  | |  |
|  | |  | |
| **Project title** | Design, develop and test the Community portal website | | |

|  |
| --- |
| **Learner declaration** |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature : Date : 24/07/2023 |

**AIM Project report**

**Content**

1. Project Background
2. Project Objective
3. Project Scope
4. Tools and Platform
5. Task 1
6. Task 2
7. Task 3
8. Task 4
9. Task 5
10. Task 6
11. Task 7
12. **Project Background**

The project aims to develop a community portal for professionals, designed to be similar to Linkedin.com. The portal will provide a platform for individuals to interact, network, and share their professional profiles. The website will be built using the Spring Boot framework for the backend and React for the frontend, combining their strengths to create a robust and user-friendly platform.

**Project Objectives**

* User Registration and Authentication: Allow new users to register on the portal and implement a secure login system to authenticate users.
* User Search Functionality: Develop a user search feature based on various parameters like First Name, Last Name, Company Name, City, and Country.
* Public Profile View: Design public user profiles showcasing professional information, skills, and experiences, accessible to other users after a successful search.
* Password Recovery: Implement a password recovery mechanism for users who forget their passwords.
* Profile Information Update: Enable users to update their profile information, including personal details, experiences, education, and skills.
* Documentation: Provide comprehensive documentation of the development process, including technical specifications and project structure

1. **Project Scope**

2.1 Features to Be Tested

1. Registration
2. Login
3. Forget the Password
4. Update Profile
5. Search Function
6. Other Profile View

2.2 Features Not to Be Tested

## Send Messages

## Read Messages

## Post a thread in Message Board

## List threads in Message Board

## Read A Thread

## List Job Opportunities & Responses

## Post Job Opportunities

1. Apply for Jobs

**Deliverables**

1) Briefly explain different types of testing which you will undertake

2) Test schedule document

3) Phase test plan

4) Test Execution and verify the results

5) Prepare the project report as per pre-defined template

6) Prepare the project presentation as per pre-defined template

7) Prepare a testing document

1. **Tools and Platform**
   1. **Project Libre**
   2. **Microsoft Word**
   3. **Microsoft PowerPoint**
   4. **Chrome**
   5. **Chrome Dev Tools**
2. **Task 1 : : Write a brief requirement specification for the Community Portal.**
3. Functional Requirements

|  |  |  |  |
| --- | --- | --- | --- |
| No | Requirements | Web Page / Process | Field detail |
|  | Users can fill in their data on the registration page. | Registration | Logo, Registration Image, Form Fields, Hyperlinks, Buttons, Copyright. |
|  | Users will get a confirmation message after registration. | Registration Confirm | Button, Message field. |
|  | Users can log in with data already available in the database. | Login | Logo, Login Image, Text Fields, Hyperlinks, Buttons, Copyright. |
|  | Users can use the forgot password feature if they forget it so they can log back in. | Forget Password | Logo, Input Fields, Button, Copyright. |
|  | Users can change or update their data | Update Profile | Biodata Section, Education Section, Experience Section, Button, Name Section, Background Image, Footer, and Copyright. |
|  | Users can search for other users. | Search | Search Field, Search Button, Search Results Page, Profile Preview Section. |
|  | Users can see the search result. | Search Result | List of Profiles, See Profile Button. |
|  | Users can view other user profiles. | Others Profile | Others Profile Preview Section. |
|  | Users can Read and react to a thread. | Homepage | Thread section, My Profile Section, Footer Section |

1. Non-Functional Requirements

|  |  |  |
| --- | --- | --- |
| No | Requirements | Purpose |
| 1 | Performance | Ensure the Community Portal operates efficiently and responds quickly to user actions. |
| 2 | Availability | Ensure the Community Portal is accessible to users with minimal downtime. |
| 3 | Security aspects | Protect user data, prevent unauthorized access, and ensure data privacy. |
| 5 | Database backup | Regularly backup the database to prevent data loss and enable recovery in case of failures. |

1. Technical requirements

Hardware :

Processor: Dual Core Cpu or more.

RAM: 4 GB or more.

HDD: 60 GB or more.

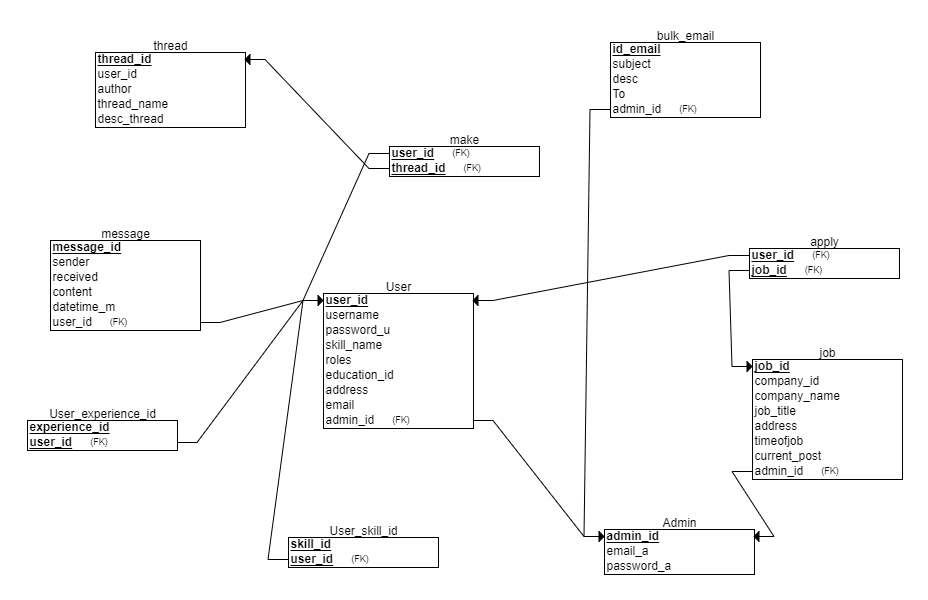
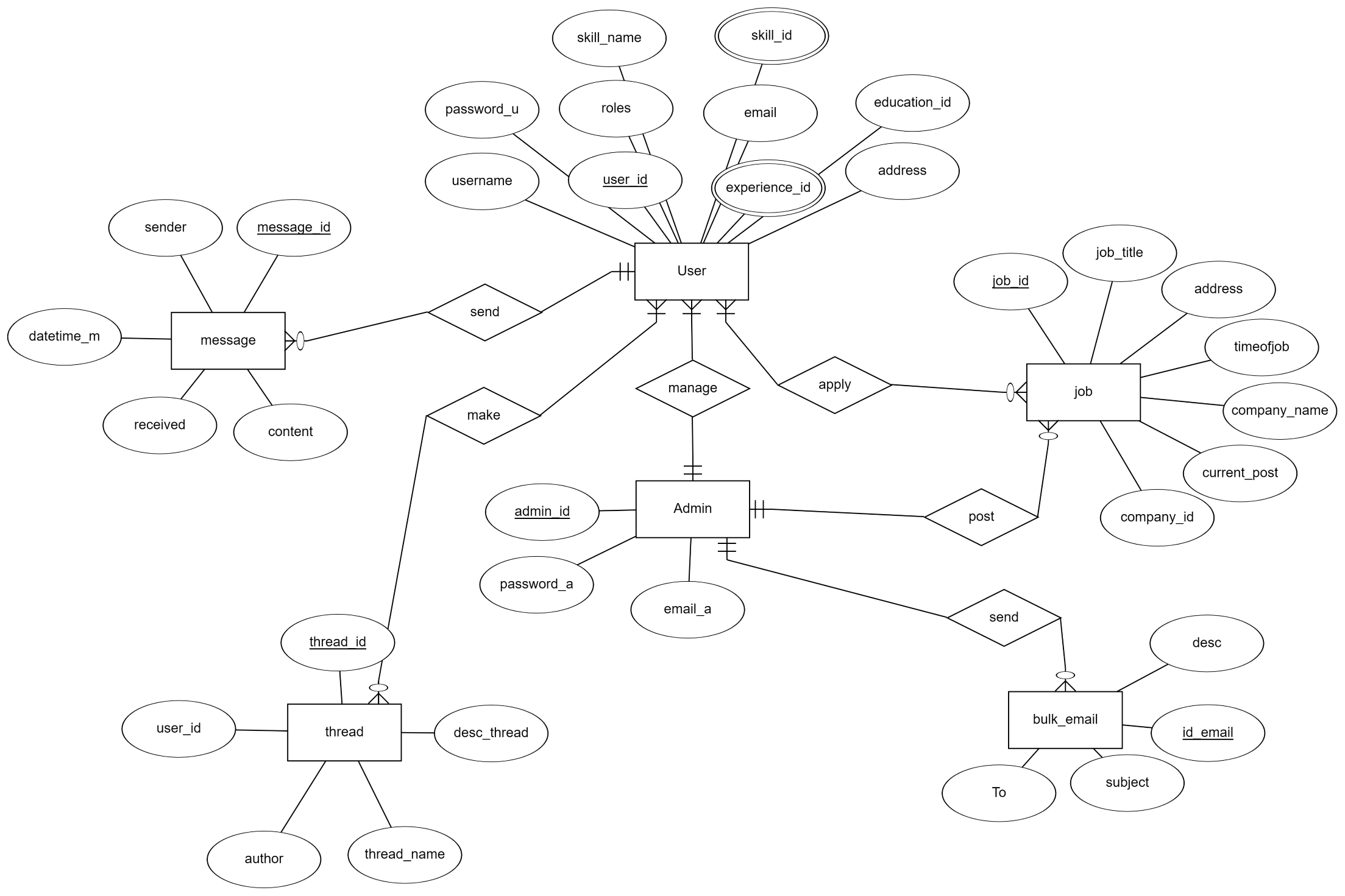
Graphics Card: Intel HD / Radeon Graphics or more.

Software :

* Eclipse IDE: Download and install Eclipse IDE for Java EE Developers. Ensure a version that supports Java EE development, including the necessary tools and plugins for Spring MVC development. (Eclipse Version: 4.26.0)
* Spring Framework: Download and include the Spring framework libraries in your project. (Spring framework 5.2.10)
* Apache Tomcat: Install a servlet container like Apache Tomcat on your machine. We need it to deploy and run the Spring MVC application. Ensure that the servlet container is compatible with the version of the Spring framework you're using. (Apache Tomcat 9.0)
* Maven or Gradle: A build automation tool like Maven or Gradle can simplify the project setup and manage dependencies effectively.
* Java Development Kit (JDK): Install the latest version of JDK on your system. Eclipse requires a compatible JDK to run. (JDK 18.0.2)

Database

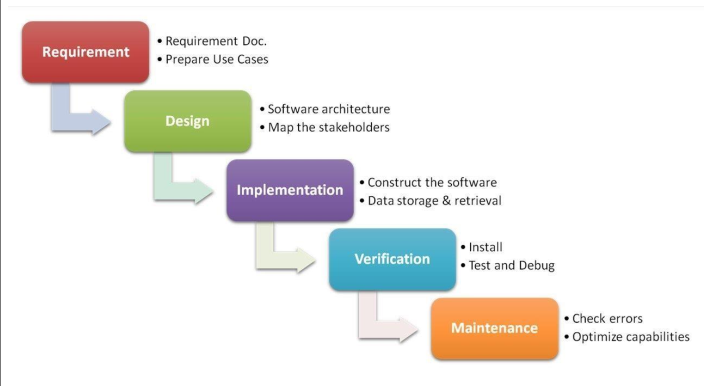
* MySQL WorkBench 8.0. (MySQL Server 8.0.31)



**Task Statement: Justify the chosen SDLC methodology with pros and cons**

**Waterfall Model**:

The Waterfall model is a sequential software development process in which projects steadily flow down through different stages, similar to a cascade waterfall. It is considered as one of the original software development life cycle (SDLC) approaches. The model is divided into separate phases, with the output of each stage serving as the input to the next. It strictly follows a top-down approach, starting with requirements collection, design, implementation, testing, and maintenance. One of its key features is that requests for one phase are completely resolved before the next phase can begin, resulting in a well-documented and structured process. However, the waterfall model may require more flexibility for projects with changing or uncertain requirements, as changes at a later stage can be difficult and expensive to implement.



Waterfall Model Strength

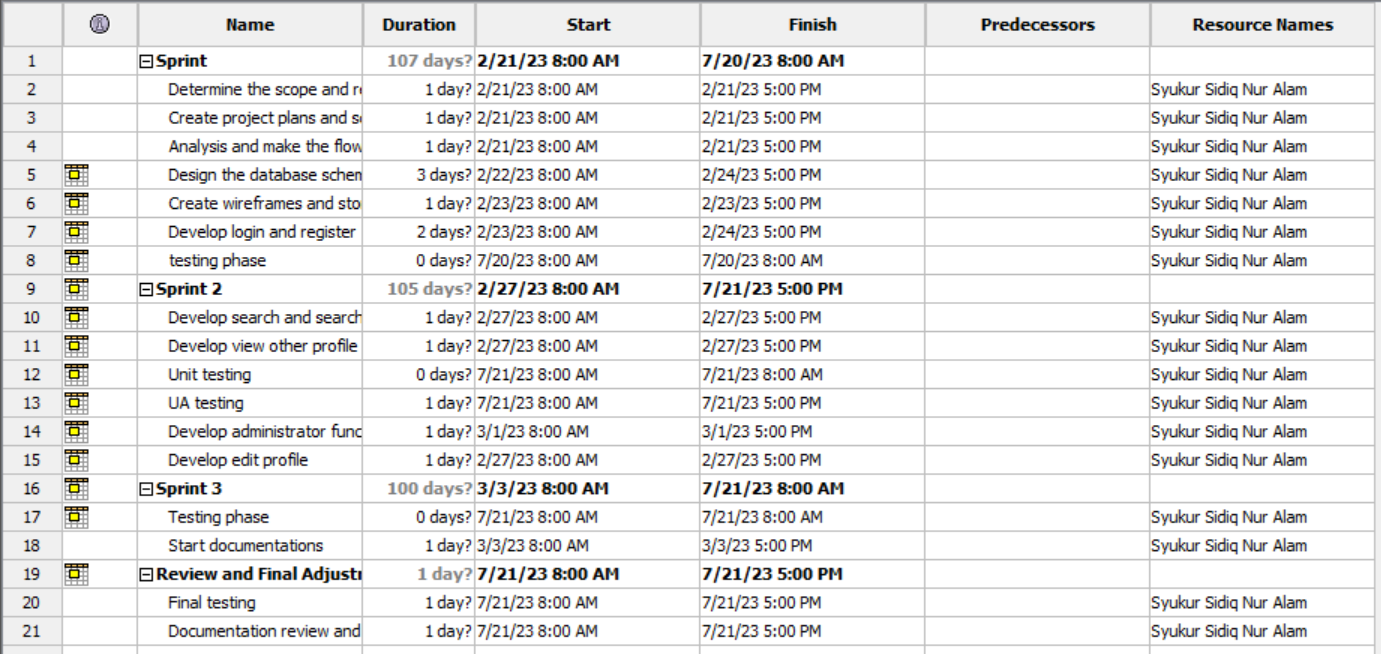
* Easy to understand and use: The simple, linear nature of the waterfall model makes it easy for team members and stakeholders to understand and apply, making it accessible to those with limited experience in software development methodologies.
* Provide structure for inexperienced staff: With well-defined stages and a sequential development process, the waterfall model provides a structured framework to guide and support inexperienced team members throughout the development process.
* Milestones are well understood: The different phases of the model create clear milestones, allowing the team a comprehensive understanding of project progress and ensuring consistent results are achieved.
* Establishing requirements stability: The waterfall model emphasizes gathering complete requirements early in the project, promoting stability within the project, and reducing the likelihood of significant changes later in the development process.
* Suitable for management control (plan, staff, track): Due to its structured nature and clear milestones, the waterfall model facilitates effective management control, allowing for better planning, staffing, and monitoring of project activities. .
* Performs well when quality is more important than cost or schedule: The waterfall model emphasizes testing and validation at every stage ensuring a focus on quality results, making it suitable for projects where ensuring a high level of quality is paramount, even if this could lead to longer development times or higher costs. .

Waterfall Model Weakness:

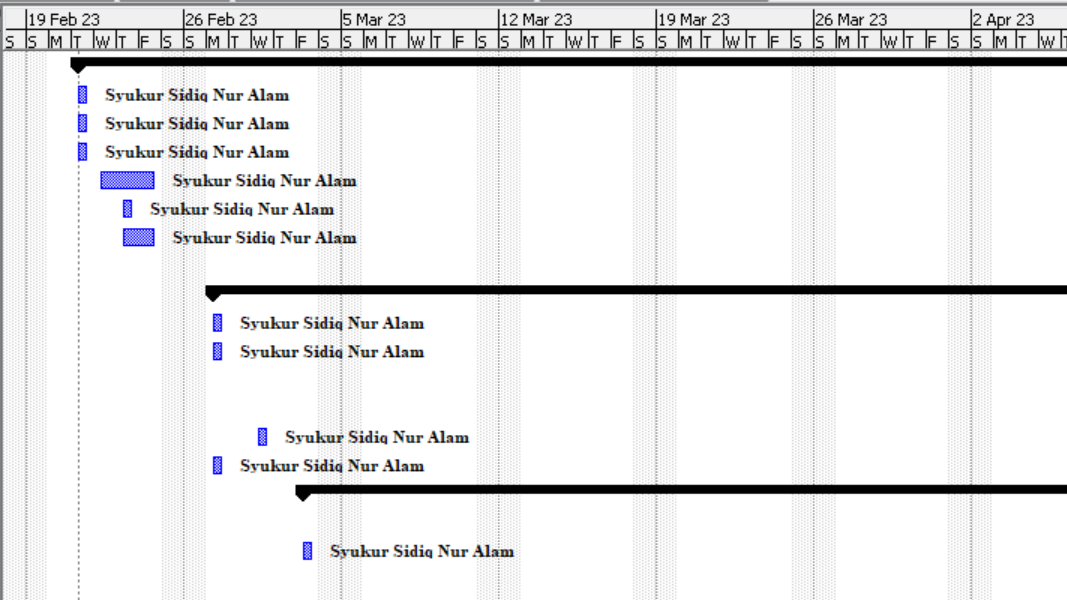
* All requirements must be known in advance: The waterfall model assumes that all requirements can be collected and defined from the beginning of the project. This can be a significant drawback as it can be difficult to accurately predict and capture all requirements, leading to potential changes or additions later in the development process.
* Submissions for each phase are considered frozen: Once this stage is completed and its submissions made, the submissions are considered final and frozen. This lack of flexibility can become a problem if changes are needed later, as previous steps may need to be reviewed and can result in additional costs and delays.
* Can give the wrong impression of progress: The linear nature of the waterfall model can create the illusion of progress as deliverables are created at each stage. However, significant progress can only be made later, which can lead to a misjudgment of the overall progress of the project.
* Does not reflect the problem-solving nature of software development – iteration phase: Software development often involves an iterative and incremental approach to solving complex problems and ever-changing needs. The rigid sequential nature of the waterfall model may not be suitable for the iterative nature of the software development process, which can lead to inefficiencies and missed improvement opportunities.
* Integration is done at the end: Integrating components and modules is often delayed until the end of development in the waterfall model. This can pose challenges in identifying and resolving integration issues, as they may not emerge until later stages of the project. .
* Few opportunities for customers to review the system: The Waterfall model often provides customers with limited opportunities to review and provide feedback on products as they are developed through the final stages. This lack of initial customer involvement can lead to a mismatch between the final product and customer expectations.

**Project Plan**

Gantt Table



Gantt Chart



1. **Task 2: Briefly explain different types of testing and how they will help to create a better system**

**Task Statement: Explain the benefits of 4 testing methods under functional and non-functional category**

1. **Functional Testing**
   1. **UAT Test (User Acceptance Testing):**

User Acceptance Testing (UAT) is a functional testing method. This involves the end user testing the software to verify if it meets their requirements and expectations. UAT ensures systems are aligned with business requirements and ready for production deployment.

* 1. Compatibility Testing (Cross-Browser Testing):

Compatibility testing, specifically Cross-Browser Testing, is a functional testing method. It ensures the software functions as intended across various web browsers and versions. This test helps verify that application functionality remains consistent regardless of the browser the end user is using.

1. **Non-Functional Testing**
2. **Performance Testing:**

A non-functional testing technique is performance testing. It assesses system performance under various circumstances, including load, stress, and scalability. The software must support the anticipated user load and provide a responsive and reliable user experience.

1. **Portability Testing (Multiple Devices / Platforms):**

Portability testing is a non-functional testing methodology determining the ease or difficulty of moving a software component or application from one environment to another. The test results obtained from Portability Testing help determine how easily a software component from one environment can be used in another environment. The term ‘environment’ refers to moving from one operating system to another operating system, one browser to another browser or from one database version to another database version.

1. **Task3 Propose relevant tests suitable for the Project Scenario**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1.  Functional Testing - UAT test cases | | |  |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Expected Result** |
| 1 | TS001 | Identify the user can log in to the portal with registered credentials. | Functional Testing | Upon attempt of authentication, users will be authenticated to the portal and given authorization of member or user as default. |
| 2 | TS002 | Identify whether users can register or not. | Functional Testing | Upon registration, they can use the registered credentials to authenticate to the User to System. |
| 3 | TS003 | Identify whether the user can use forgot password feature. | Functional Testing | The Reset Password Page should allow the user to enter a new password and confirm the password change. |
| 4 | TS004 | test function users can edit their profile information | Functional Testing | The database should correctly store the modified data. |
| 5 | TS005 | Find People by using the search function. | Functional Testing | The profile displayed must match the search parameters entered (eg, First Name, Last Name, and City). |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 2.  Compatibility - Cross-browser testing/responsiveness | | | |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Expected Result** |
| 6 | TS006 | Identify if the Result portal works as intended on Microsoft Edge. | Cross-browser Testing | The look and feel are working as intended on Microsoft Edge. |
| 7 | TS007 | Identify if the result portal works as intended on Mozilla Firefox. | Cross-browser Testing | The look and feel are working as intended on Mozilla Firefox. |
| 8 | TS008 | Identify if the Result portal works as intended on Chrome. | Cross-browser Testing | The look and feel are working as intended on Google Chrome. |

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 3. Performance Efficiency - Load Testing/Stress Testing | | | |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Expected Result** |
| 9 | TS09 | Identify if the register page load time is within 10 – 15 | Performance Testing | The register page will load in less than 15 seconds |
| 10 | TS010 | Identify if the login page load time is within 10 – 15 | Performance Testing | The login page will load in less than 15 seconds |
| 11 | TS011 | Identify if the index load time is within 10 – 15 | Performance Testing | The index page will load in less than 15 seconds |

leTS

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 4.  Portability - Multi platforms (different platforms, different devices) | | | | |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Expected Result** |
| 12 | TS012 | Identify whether the portal functionality works as intended using various platforms and Mobile. | Portability Testing | The portal should have a responsive look and feel, which scales down to a mobile view. |
| 13 | TS013 | Identify whether the portal functionality works as intended using various platforms and Tablets. | Portability Testing | The portal should have a responsive look and feel, which scales down to a Tablet view. |
| 14 | TS014 | Identify whether the portal functionality works as intended using various platforms and Dekstop. | Portability Testing | The portal should have a responsive look and feel with a Dekstop view. |

1. **Task 4: Optimal Scheduling**

**Task Statement: Identify the appropriate method to compute the testing schedule duration and prepare the test schedule**

|  |  |  |  |
| --- | --- | --- | --- |
| **Web page** | **Weighting** | **Functional Points (FP)** | **Total** |
| Registration | 3 | 5 | 15 |
| Login | 1 | 3 | 3 |
| Reset Password | 1 | 1 | 1 |
| Profile | 3 | 3 | 9 |
| Update Profile | 3 | 4 | 12 |
| Search Users | 3 | 3 | 9 |
| See Profile Other Users | 3 | 4 | 12 |
|  |  | Total functional points | 61 |
|  |  | Estimated person hour per FP | 1 |
|  |  | Total hours needed | 62 |
|  |  | Total days needed | 8 |

Test Schedule :

|  |  |  |
| --- | --- | --- |
| Test Schedule ID : | TSCH01 |  |
| Product ID / Name : | ABC Jobs |  |
| Product Version or Build : | 0.0.1 |  |
| Present Owner : | Syukur Sidiq Nur Alam | Software Engineer |
| Created On : | 21 July 2023 |  |
| Review On : | 24 July 2023 |  |
| Review By : | Syukur Sidiq Nur Alam | QA Manager |
| Review Comments : | Great |  |
| Current Version : | 1.0.0 |  |
| Change Details : | Send Email Feature |  |
| Current Status : | Complete |  |
| Signing Off Authority : | Syukur Sidiq Nur Alam | Software Engineer |

1. **Task 5:** **Phase Test Plan**

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| No | Task | Start | End | Person Days | Resource |
| 1 | Prepare Test Plan | 17-7-2023 | 17-7-2023 | 1 | Project Manager |
| 2 | Review Test Plan | 17-7-2023 | 17-7-2023 | 1 | Quality Assurance |
| 3 | Prepare Test Scripts | 18-7-2023 | 18-7-2023 | 1 | Software Engineer |
| 4 | Review Test Scripts | 18-7-2023 | 18-7-2023 | 1 | Quality Assurance |
| 5 | Set up Test Environment | 19-7-2023 | 19-7-2023 | 1 | Software Engineer |
| 6 | Maintain Test Environment | 19-7-2023 | 19-7-2023 | 1 | Software Engineer |
| 7 | Establish Test Procedures | 20-7-2023 | 20-7-2023 | 1 | Quality Assurance |
| 8 | Promote Software from Development to Test Environment | 20-7-2023 | 20-7-2023 | 1 | Software Engineer |
| 9 | Execute Testing/Record Faults | 21-7-2023 | 21-7-2023 | 1 | Tester |
| 10 | Re-execute Tests as Needed | 21-7-2023 | 21-7-2023 | 1 | Tester |
| 11 | Provide Support/Resolve Faults | 22-7-2023 | 22-7-2023 | 1 | Tester |
| 12 | Promote Software from Test to Development Environment | 23-7-2023 | 23-7-2023 | 1 | Software Engineer |
| 13 | Certify Application for Release to Beta Test Group | 24-7-2023 | 24-7-2023 | 1 | Project Manager |

1. **Task 6 : UNIT and UAT Test Scripts**

**Task Statement : Prepare the test scripts with use case, scenario, test description, pre-conditions, test data, and expected output**

Test Scenario :

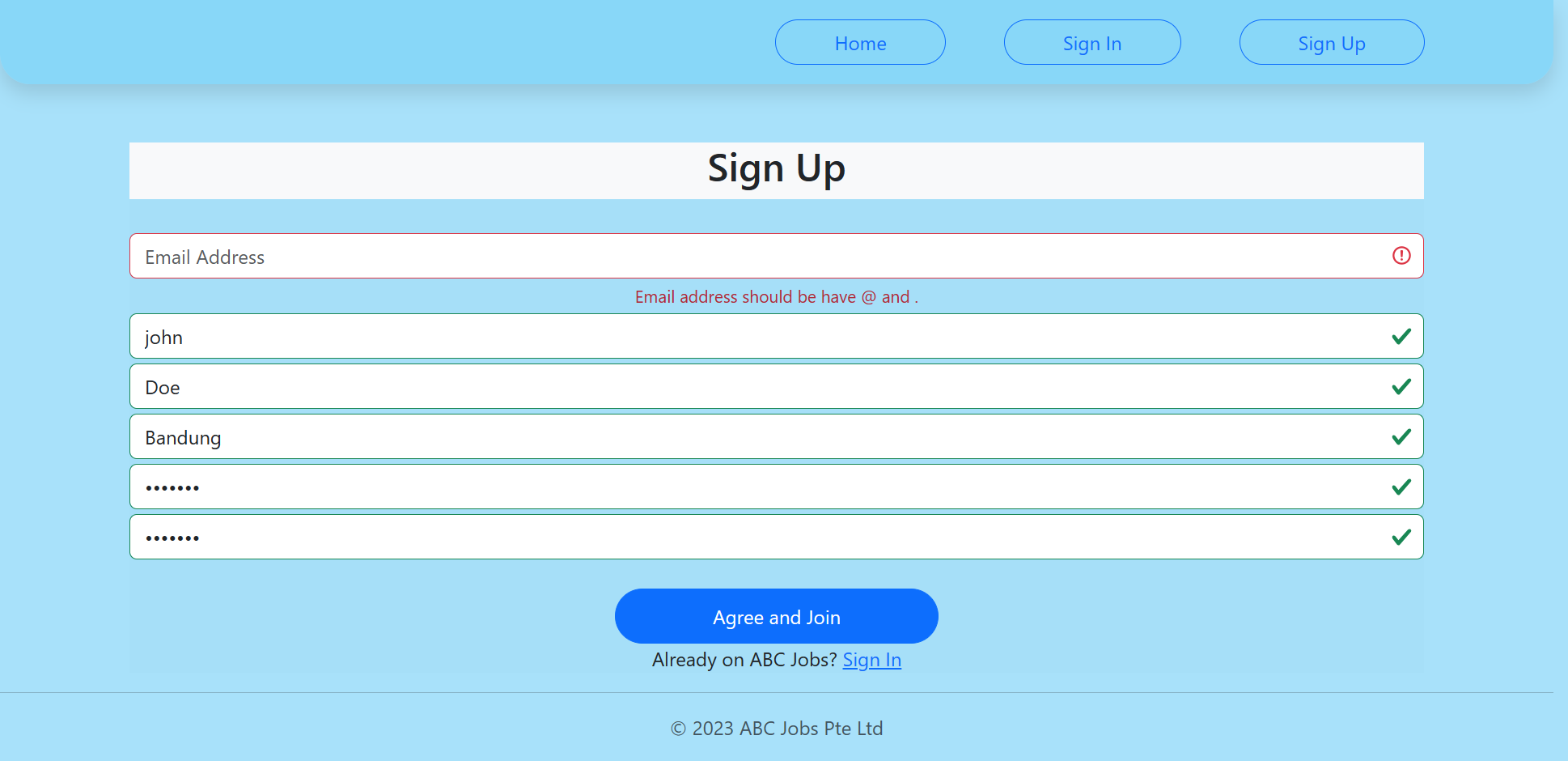
|  |  |  |  |
| --- | --- | --- | --- |
| Requirements | Test Scenario ID | Test Scenario | Number of test cases |
| Functional Testing | TS001-Unit testing | Ensure that the user enters the correct data on the registration page | 3 |
| TS002-UAT testing | Check that a navigation menu is provided on each page and is consistent. | 3 |

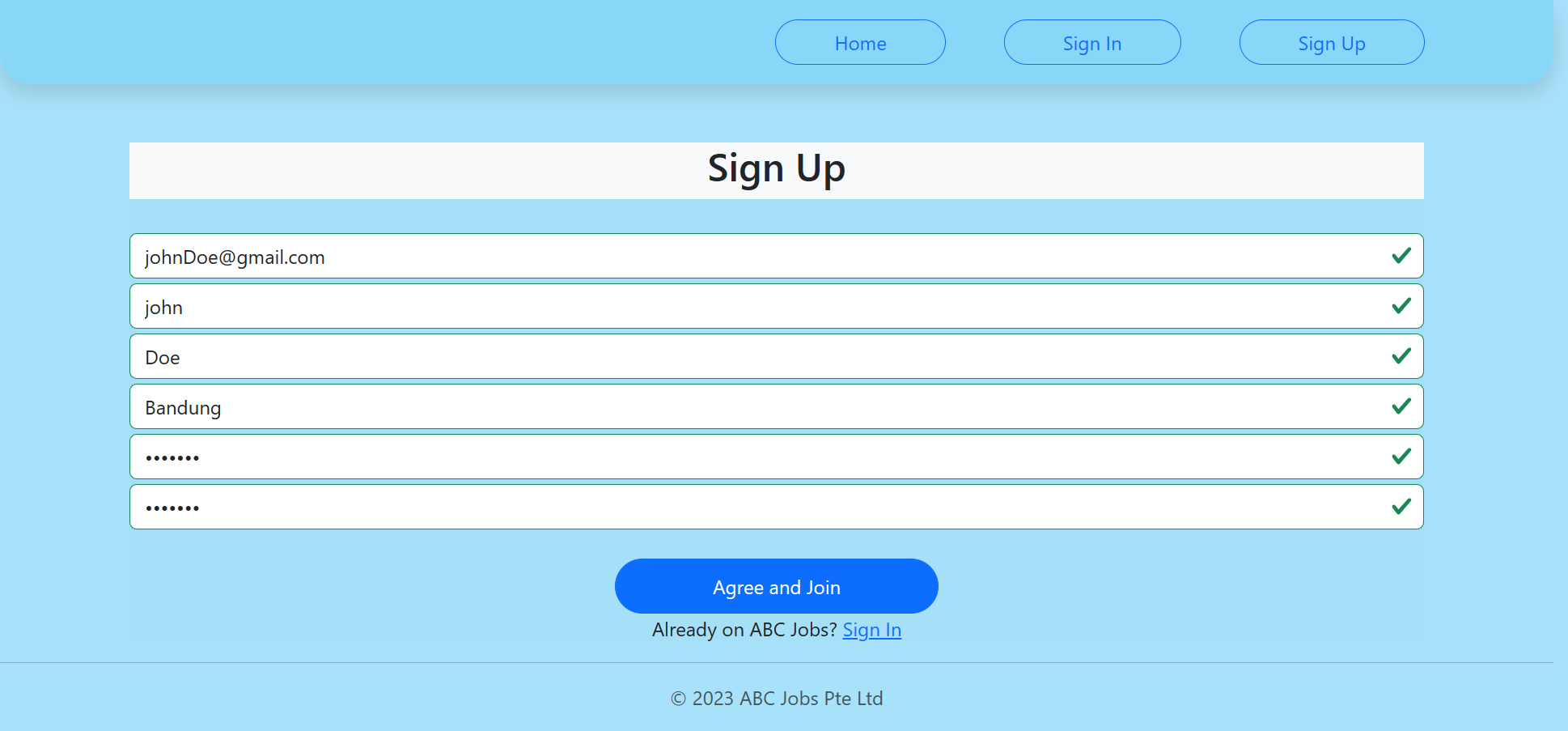
Scenario 1 : Unit Testing

|  |  |
| --- | --- |
| Test Scenario | Ensure that the user enters the correct data on the registration page |
| TS001 |
| Test Cases | Unit Testing |
| TC001 | Check that all the mandatory fields are entered. |
| TC002 | Check that the email-address validation is working correctly. |
| TC003 | Check post submission loading of thank you page. |

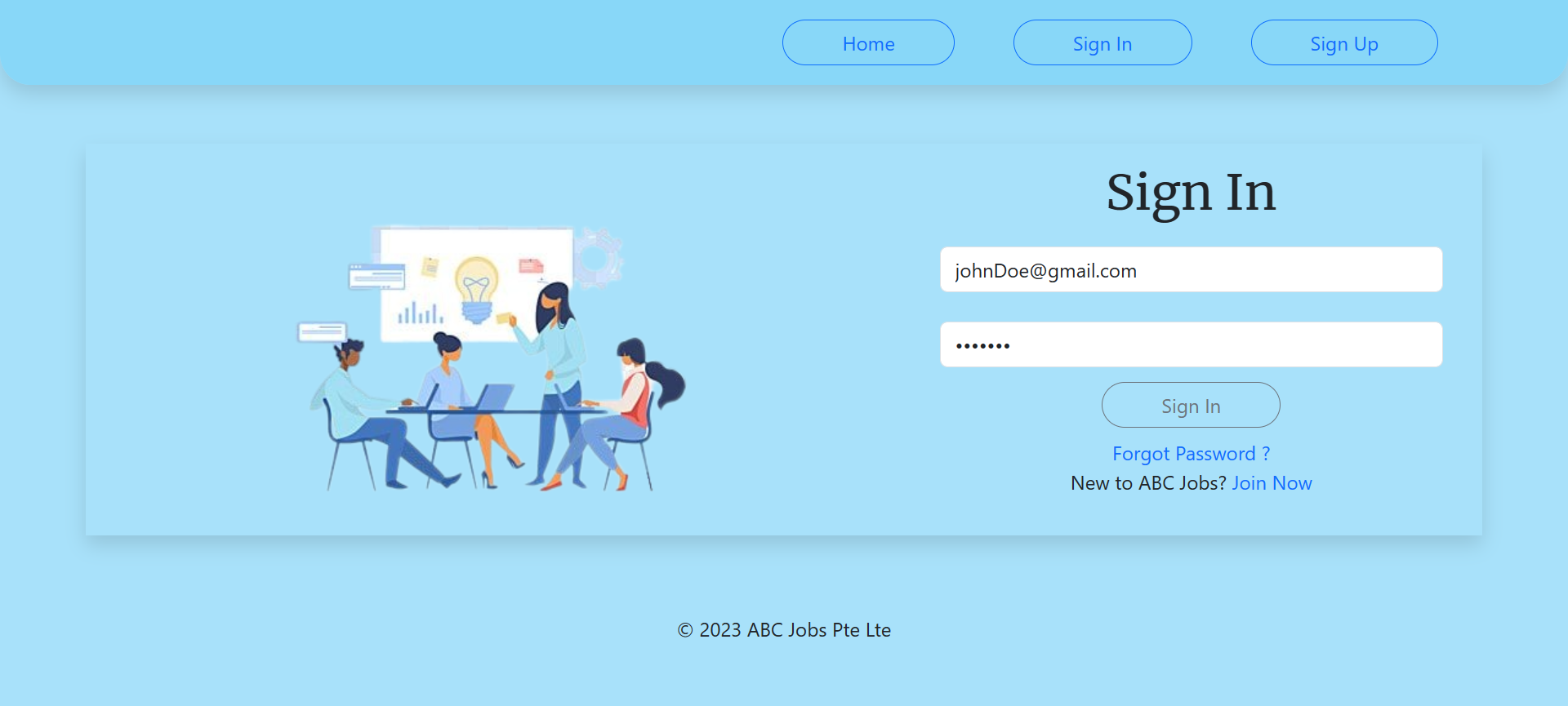
|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Scenario ID | Test Case ID | Test Case | Preconditions | Test Step | Test Data | Expected Results | Actual Results | Pass / Fail |
| TS001 | TC001 | Test Case 1 | The user should have entered the website's URL and accessed the abcjobs.com homepage. | 1. Type www.abcjobs.com. 2. Click “Sign up” on the website home page linked to the registration page. 3. Enter all the fields. 4. Click submit. | First Name = john  Last Name = Doe  City = Bandung  Email = [JohnDoe@gmail.com](mailto:JohnDoe@gmail.com)  Password = bjir123 | The user should land on to login page. | As expected | Pass |
| First Name = john  Last Name = Doe  City = Bandung  Email =  Password = bjir123 | It will display the error message “Email address should be have @ and . ” | As expected | Pass |
| TS001 | TC001 | Test case 2 | The user should have entered the website's URL and accessed the abcjobs.com homepage. | 1. Enter the URL 2. Click on the Sign-In Button 3. Enter all details 4. Enter submit button | Email = [JognDoe@gmail.com](mailto:JognDoe@gmail.com)  Password = bjir123 | Continue to the homepage user | As expected | Pass |
| TS001 | TC002 | Test case 3 | The user should have entered the website's URL and accessed the abcjobs.com homepage. | 1. Type www.abcjobs.com.  2. Click the Sign In Button.  3. Enter all the fields.  4. Click submit. | Email = [JohnDoe.gmail](mailto:demara@gmail.com)  Password = bjir123 | It will display the error message “Enter valid email” | As expected | Pass |
| TS001 | TC003 | Test case 4 | The user should have the URL of the website and open the homepage of the abcjobs.com | 1. Type www.abcjobs.com.  2. Click “Sign up” on the website home page linked to the registration page.  3. Enter all the fields.  4. Click submit. | First Name = John  Last Name = Doe  City = Bandung  Email = [JohnDoe@gmail.com](mailto:JohnDoe@gmail.com)  Password = bjir123 | User should land to thank you page | As expected | Pass |

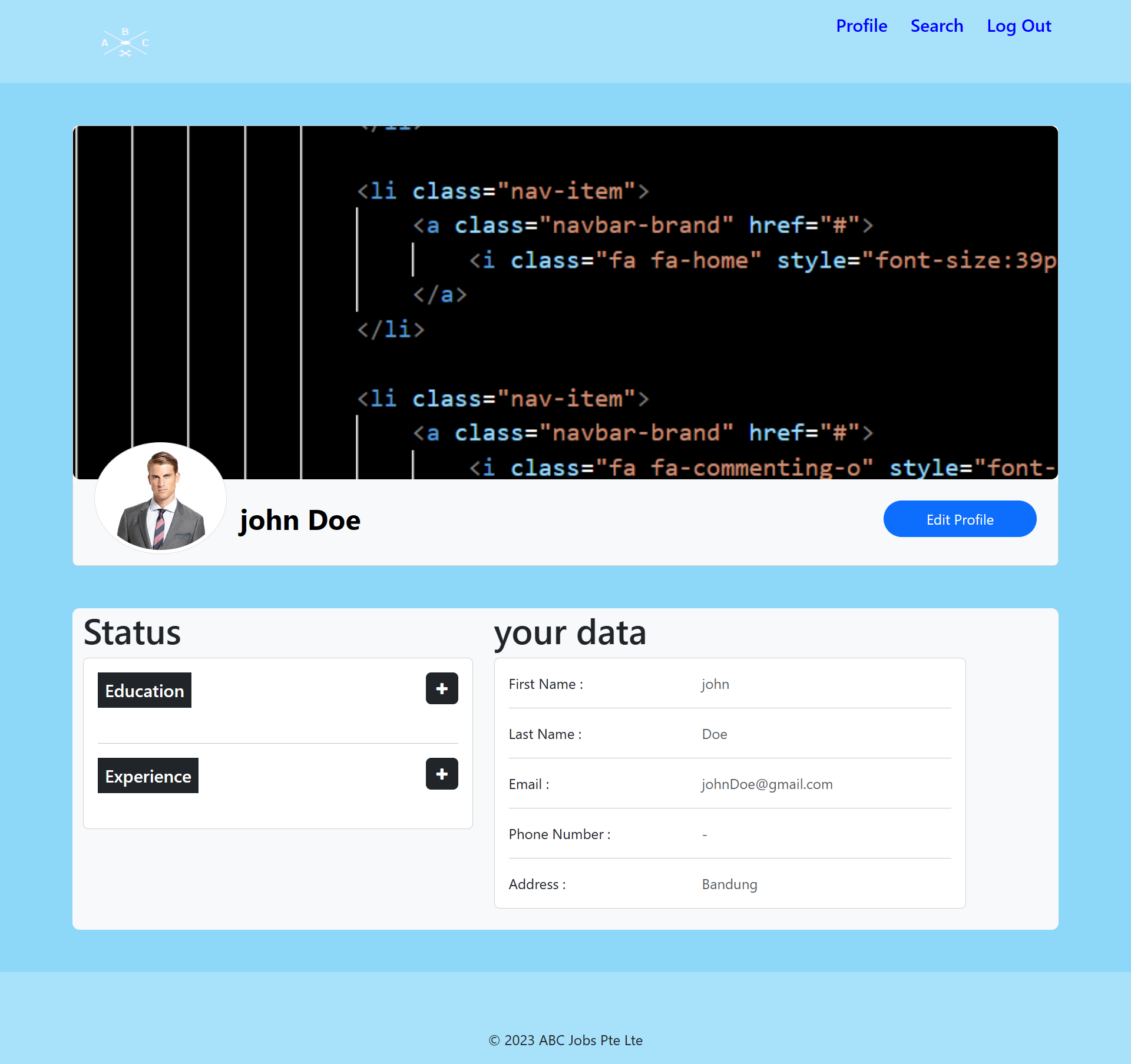
**TC001**

****

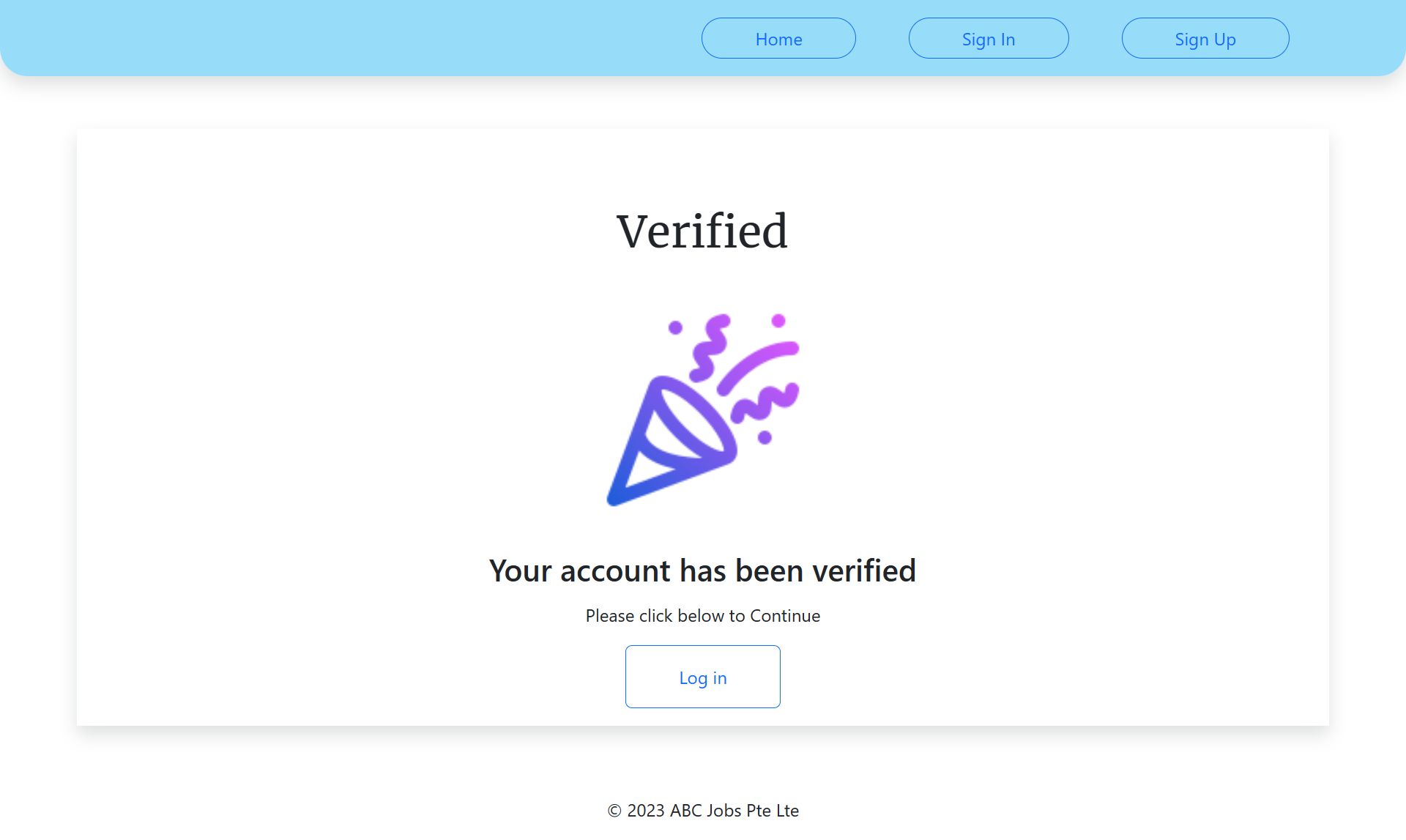
****

**TC002**

****

****

**TC003**

****

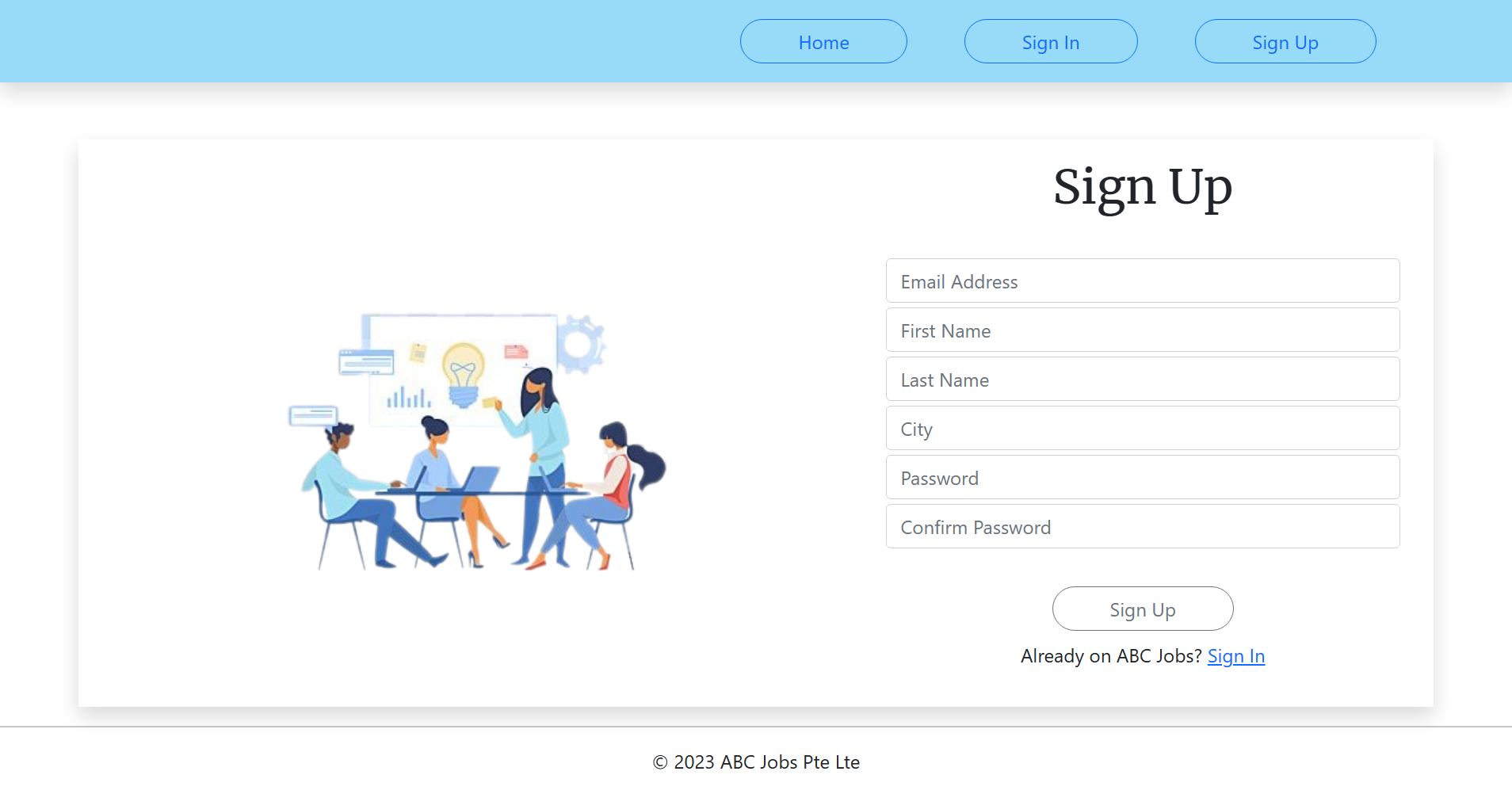
Scenario 2 : UAT Testing

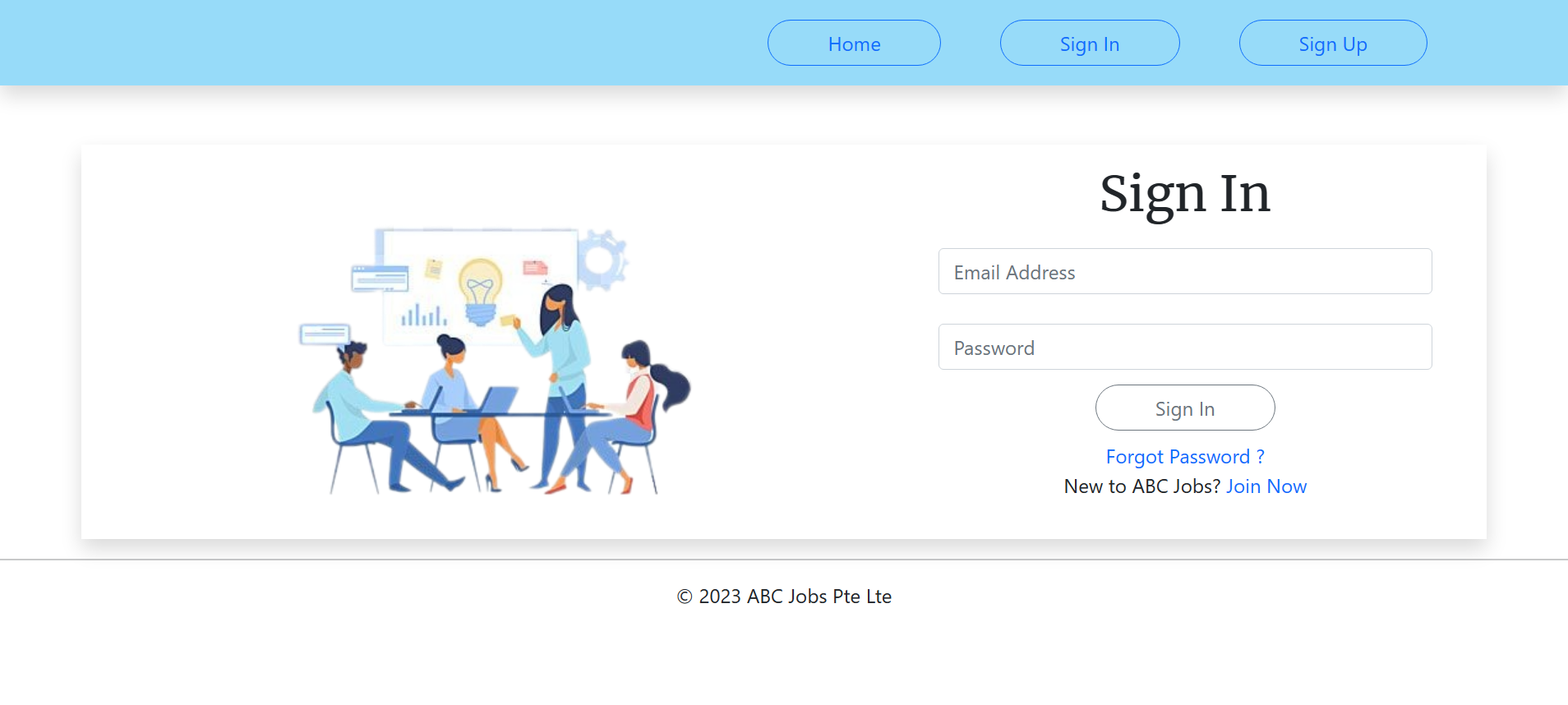
|  |  |
| --- | --- |
| Test Scenario | Check that a navigation menu is provided on each page and is consistent. |
| TS002 |
| Test Cases | UAT Testing |
| UAT01 | Check that each hyperlink menu is loading the correct page |
| UAT02 | Sign in to the community portal with the account that has been created. |
| UAT03 | Reset the password |

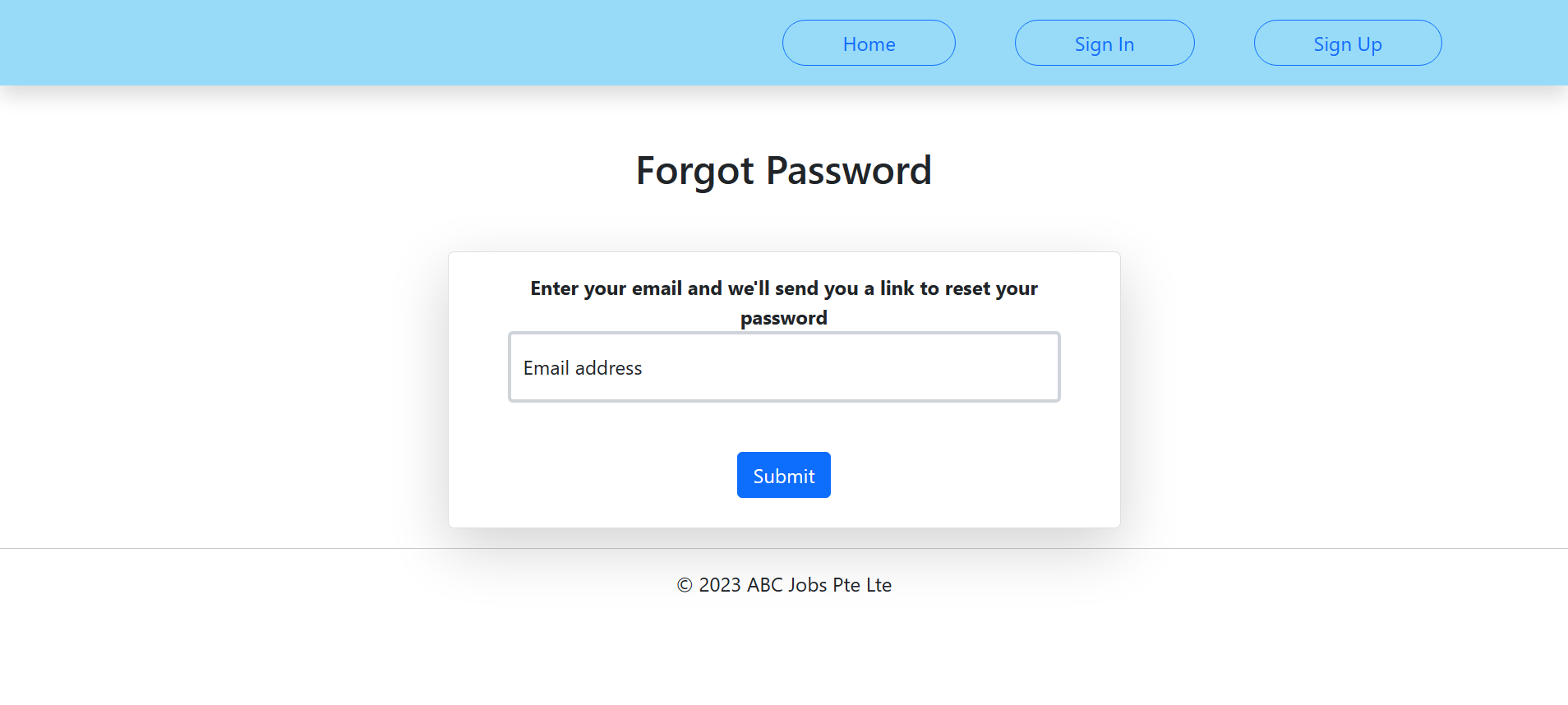
Test cases:

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
| Scenario ID | Test Case ID | Test Case | Preconditions | Test Step | Test Data | Expected Results | Actual Results | Pass / Fail |
| TS002 | UATC01 | Check that each button and hyperlink menu is loading the correct page | The website is configured to Browser by Apache Tomcat Server. | 1. Go to Google Chrome. 2. Type www.abcjobs.com and enter. | Click on the sign up button.  Click on the sign in button.  Click on the forgot password link. | Loaded Registration page  Loaded Login page  Loaded Forgot Password page | As expected | Pass |
| TS002 | UATC02 | Log-in to community portal with the account that has been created | The website is configured to Browser by Apache Tomcat Server. | 1. Go to Google Chrome 2. Type www.abcjobs.com and enter. 3. User will land on Home Page 4. Click sign-in button | registered username and password | The user will be redirected to user profile page | As expected | Pass |
| TS002 | UATC03 | Reset the password | The website is configured to Browser by Apache Tomcat Server. | 1. Go to Google Chrome 2. Type www.abcjobs.com and enter. 3. User will landing on Home Page 4. Click Log-in button 5. Click Forget Password button | Registered email address | User will be redirect to Forget Password Confirmation page | As expected | Pass |

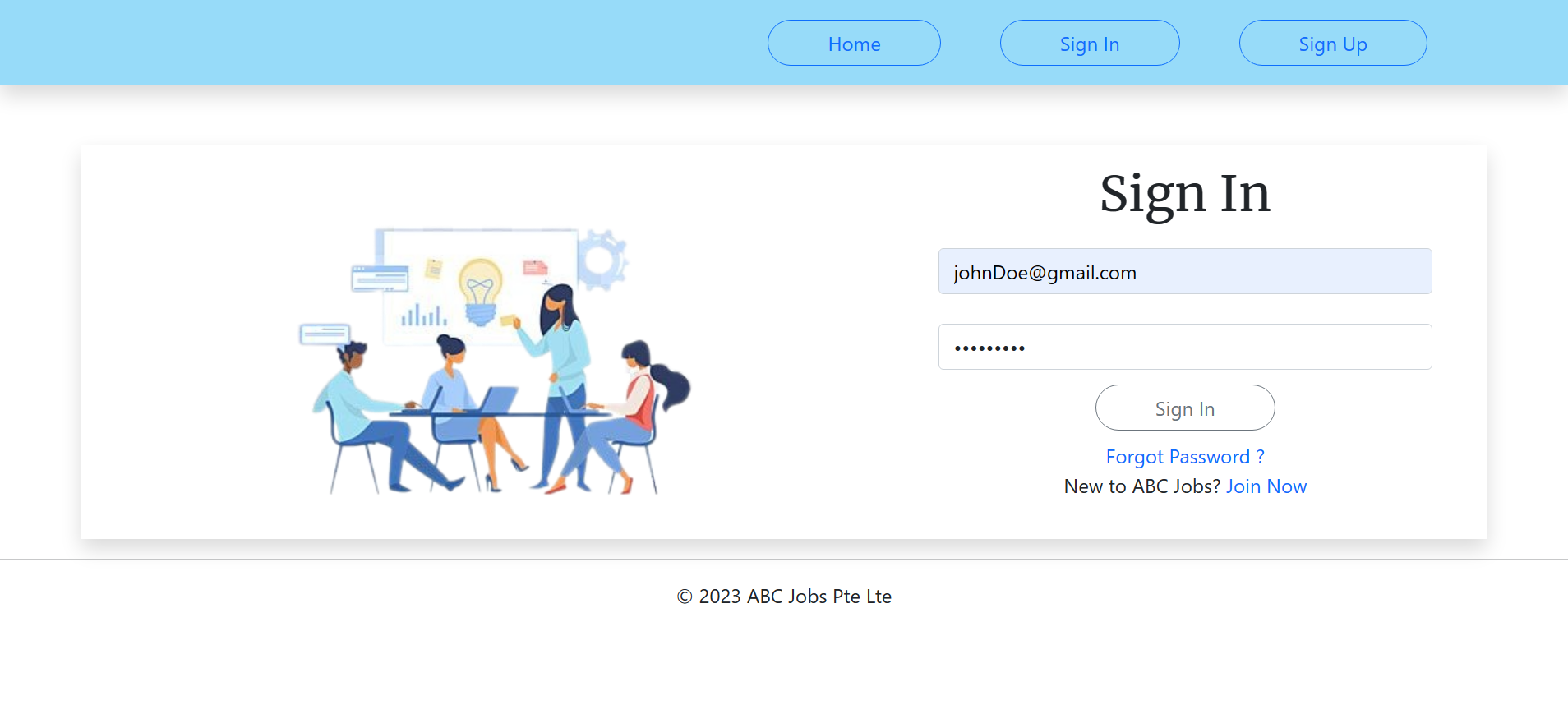
**UAT001**

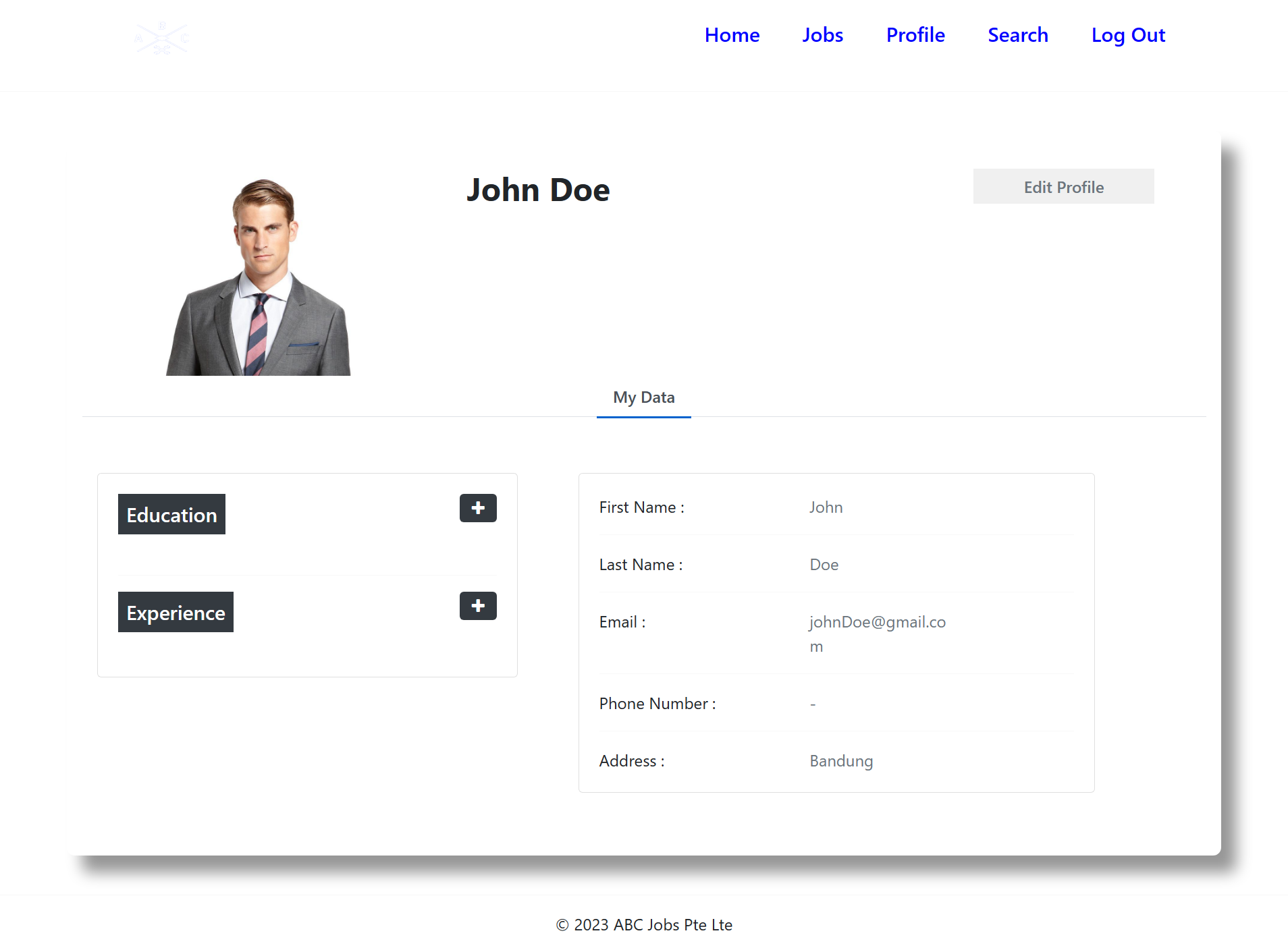
****

****

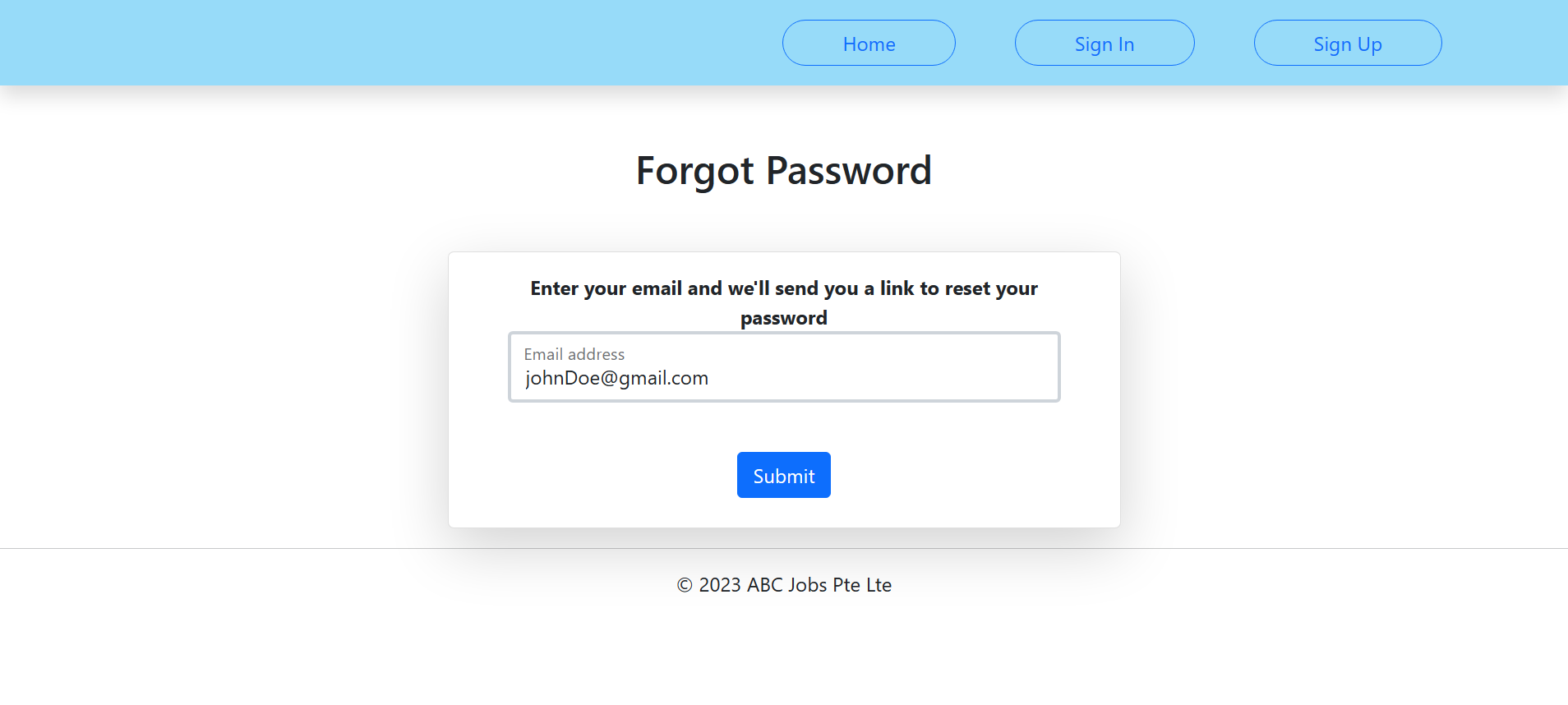
****

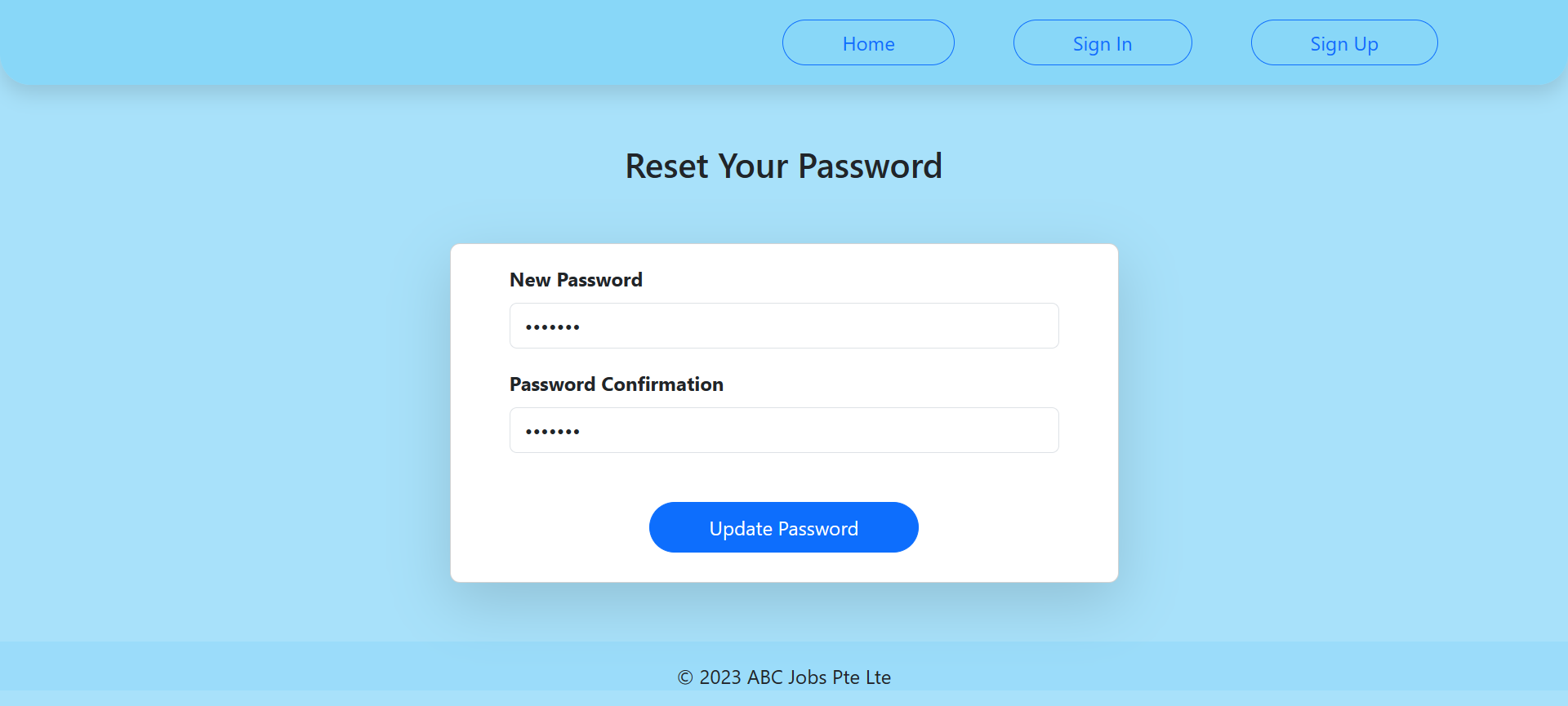
**UAT002**

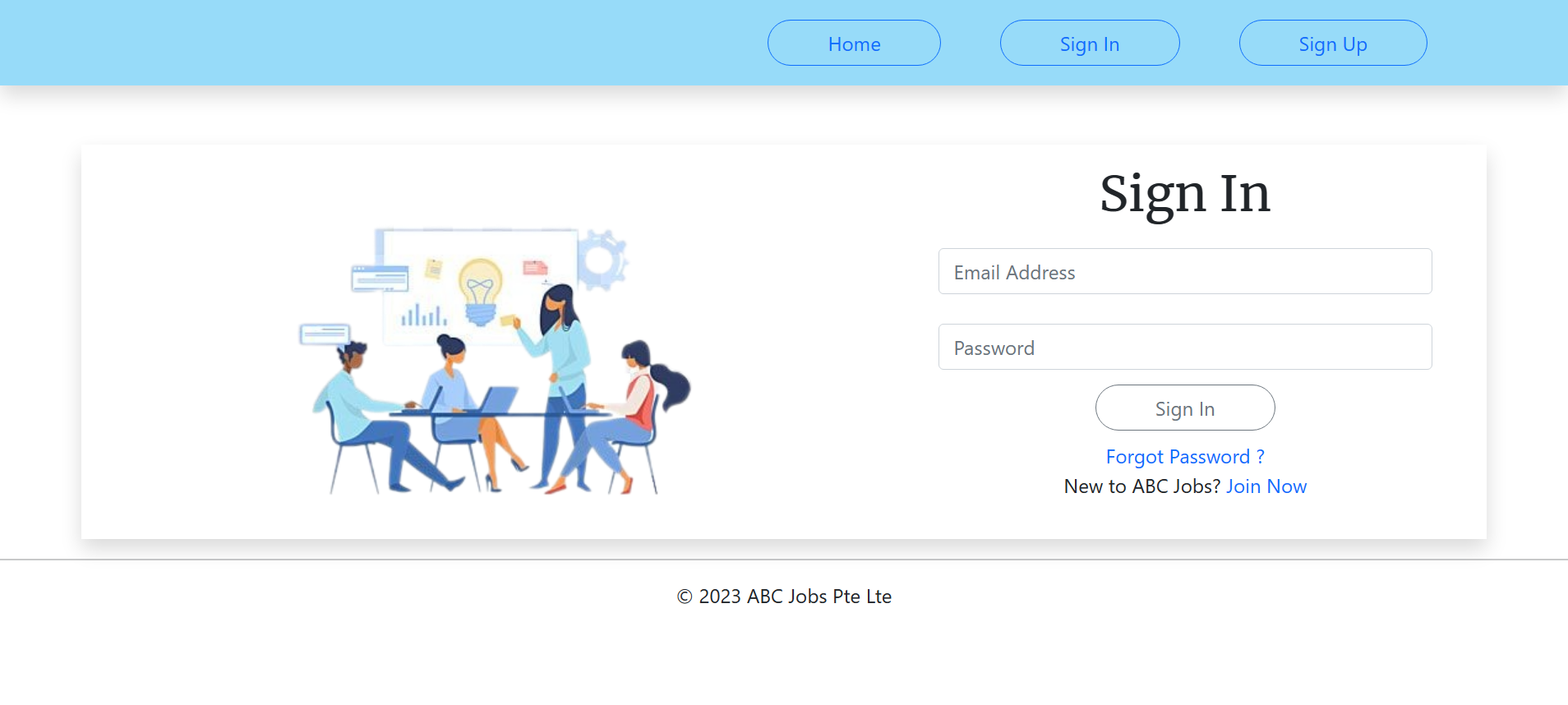
****

****

**UAT003**

****

****

****

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.  Functional Testing - UAT test cases | | |  |  |  |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Pro** | **Risk** | **Expected Result** |
| 1 | TS001 | Determines whether the user can login to the portal with the saved credentials. | Functional Testing | High | login without details. | Upon attempt of authentication, users will be authenticated to the portal and given authorization of member or user as default. |
| 2 | TS002 | Determine if the user can successfully register | Functional Testing | Medium | The user will not be able to use the website. | Upon registration, they can use the registered credentials to authenticate to the User to System. |
| 3 | TS003 | Determines whether the user can successfully use the Forgot Password feature. | Functional Testing | High | The password will forget forever, and the user cannot log in. | The Reset Password Page should allow the user to enter a new password and confirm the password change. |
| 4 | TS004 | Make sure users can edit their profile information correctly. | Functional Testing | Medium | The modified data cannot store in the database, and the user’s profile cannot update. | The database should correctly store the modified data. |
| 5 | TS005 | Find people by using the search function. | Functional Testing | Low | People will not show. | The profile displayed must match the search parameters entered (eg, First Name, Last Name, and City). |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | 2.  Compatibility - Cross-browser testing/responsiveness | | | |  |
| Test Id | **Test Scenario** | | **Test Objective** | | **Pro** | **Risk** | **Technique** | **Expected Result** |
| 6 | TS006 | | Identify if the Result portal works as intended on Google Chrome. | | High | Users may get annoyed with the different look and feel of the portal and lose credibility on the portal. | Cross-browser Testing | The look and feel are working as intended on Google Chrome. |
| 7 | TS007 | | Identify if the Result portal works as intended on Microsoft Edge. | | High | Users may get annoyed with the different look and feel of the portal and lose credibility on the portal. | Cross-browser Testing | The look and feel are working as intended on Microsoft Edge. |
| 8 | TS008 | | Identify if the result portal works as intended on Mozilla Firefox. | | High | Users may get annoyed with the different look and feel of the portal and lose credibility on the portal. | Cross-browser Testing | The look and feel are working as intended on Mozilla Firefox. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | 3. Performance Efficiency - Load Testing/Stress Testing | | | |  |
| Test Id | **Test Scenario** | | **Test Objective** | | **Pro** | **Risk** | **Technique** | **Expected Result** |
| 9 | TS009 | | Identify if the index load time is within 10 – 15 | | High | Users will lose interest and not want to use the application. | Performance Testing | The index page will load in less than 15 seconds. |
| 10 | TS010 | | Identify if the login page load time is within 10 – 15 | | High | Users will lose interest and not want to use the application. | Performance Testing | The login page will load in less than 15 seconds. |
| 11 | TS011 | | Identify if the register page load time is within 10 – 15 | | High | Users will lose interest and not want to use the application. | Performance Testing | The register page will load in less than 15 seconds. |

|  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- |
|  | |  | | 4.  Portability - Multi platforms (different platforms, different devices) | | | | |
| Test Id | **Test Scenario** | | **Test Objective** | | **Pro** | **Risk** | **Technique** | **Expected Result** |
| 12 | TS012 | | Identify whether the portal functionality works as intended using various platforms and Dekstop. | | High | Users cannot fully utilize the function on the portal page in a Desktop view. | Portability Testing | The portal should have a responsive look and feel with a Dekstop view. |
| 13 | TS013 | | Identify whether the portal functionality works as intended using various platforms and Tablets. | | High | Users cannot fully utilize the function on the portal page in a Desktop view. | Portability Testing | The portal should have a responsive look and feel, which scales down to a Tablet view. |
| 14 | TS014 | | Identify whether the portal functionality works as intended using various platforms and Mobile. | | High | Users cannot fully utilize the function in the portal page in a Desktop view. | Portability Testing | The portal should have a responsive look and feel, which scales down to a mobile view. |

**Test Cases :**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Scenario ID | Test Case ID | Test Case | Test Step | Test Data | Expected Results | Actual Results | Pass / Fail |
| TS001 |  | Identify whether the user can log in to the portal with registered credentials. | On the homepage, click sign-in bottom on the navbar. Input the email and password. | Email: [yangbener@gmail.com](mailto:yangbener@gmail.com)  Pass: nantinangiss | Upon attempt of authentication, users will be authenticated to the portal and given authorization of member or user as default. | As expected | Pass |
| TS002 |  | Identify whether the user can successfully register or not. | On the homepage, click sign-up button. Fulfill the form. | Email:  [yangbener@gmail.com](mailto:%20yangbener@gmail.com)  Pass: nantinangiss | Upon registration, they can use the registered credentials to authenticate to the User to System. | As expected | Pass |
| TS003 |  | Identify whether the user can successfully use forgot password feature. | On the homepage, click sign-in bottom on the navbar. Click Forgot Password Link. Enter your email, and enter a new password | Email:  [yangbener@gmail.com](mailto:%20yangbener@gmail.com)  Pass: kannangis | The Reset Password Page should allow the user to enter a new password and confirm the password change. | As expected | Pass |
| TS004 |  | Make sure users can accurately edit their profile information. | After login, go to your profile, and click the edit profile button. Also, you can add education and experience | Phone number:  626263394756 | The database should correctly store the modified data. | As expected | Pass |
| TS005 |  | Find different programmers by using the search function. | After Login, click the button search on the navbar. You can search by first name, last name, and city. | JohnDoe | The profile displayed must match the search parameters entered (eg, First Name, Last Name, and City). | As expected | Pass |
| TS006 |  | Identify if the Result portal works as intended on Google Chrome. | Open Google Chrome and type  http://localhost:8080/ABCPortal | Google Chrome App | The look and feel are working as intended on Google Chrome. | As expected | Pass |
| TS007 |  | Identify if the Result portal works as intended on Microsoft Edge. | Open Microsoft Edge and type  http://localhost:8080/ABCPortal | Microsoft Edge App | The look and feel are working as intended on Microsoft Edge. | As expected | Pass |
| TS008 |  | Identify if the result portal works as intended on Mozilla Firefox. | Open Mozilla Firefox and type  http://localhost:8080/ABCPortal | Mozilla Firefox App | The look and feel are working as intended on Mozilla Firefox. | As expected | Pass |
| TS009 |  | Identify if the index load time is within 10 – 15 | type  <http://localhost:8080/ABCPortal> on your browser and see the page will load less in 15 seconds | Browser | The index page will load in less than 15 seconds | As expected | Pass |
| TS010 |  | Identify if the login page load time is within 10 – 15 | Click the Sign In button, and you will go to Login Page and see the page will load in less than 15 seconds. | Browser | The login page will load in less than 15 seconds | As expected | Pass |
| TS011 |  | Identify if the register page load time is within 10 – 15 | Click the Sign-Up button, and you will go to Login Page and see the page will load in less than 15 seconds. | Browser | The register page will load in less than 15 seconds | As expected | Pass |
| TS012 |  | Identify whether the portal functionality works as intended using various platforms and Dekstop. | type  <http://localhost:8080/ABCPortal> on your browser and see if the page will appear correctly. | Browser | The portal should have a responsive look and feel with a Dekstop view. | As expected | Pass |
| TS013 |  | Identify whether the portal functionality works as intended using various platforms and Tablets. | type  <http://localhost:8080/ABCPortal> on your browser and see if the page will appear correctly. | Browser | The portal should have a responsive look and feel, which scales down to a Tablet view. | As expected | Pass |
| TS014 |  | Identify whether the portal functionality works as intended using various platforms and Mobile. | type  <http://localhost:8080/ABCPortal> on your browser and see if the page will appear correctly. | Browser | The portal should have a responsive look and feel, which scales down to a mobile view. | As expected | Pass |