A picture containing text, clipart

Description automatically generated

**Formative Assignment 3**

|  |  |
| --- | --- |
| Module Name | Application Implementation (Agile Project Management) |
| Course Name | Bachelor of Technology in Software Engineering |
| Assignment Title | Test Preparation to test the community portal |

|  |  |
| --- | --- |
| Learner Name | Agung Yuda Pratama |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.    Student signature: Date: 22-07-2023 |

## Purpose of this assignment

To demonstrate your capabilities in the following areas:

* Preparing Test Cases and Scripts to test a software application

## Submission Format

The submission is in the form of an individual written report with examples and illustrations. This should be written in a concise, formal academic style using single spacing and font size 12.You are required to make use of headings, paragraphs, and subsections as appropriate, and all work must be supported with research and referenced using the Harvard referencing system. Please also provide a bibliography using the Harvard referencing system.

## Submission file

Submission File Name format: Cohort Code \_Full Name\_Assignment Number

E.g.: AIM-1222A\_AliceTan\_A3.docx

## Scenario

## You have been approached by ‘ABC Jobs Pte Ltd’ as a website developer to develop a community portal for Software Developers. For this module the scope is to Design, Develop, Implement & Document the data-driven website using the framework Spring boot and react.

The Scope of the Project is to design a Community Portal Similar to Linkedin.com. Users will be able to register in the portal using the Registration Page. Users of the portal can search for other users using various parameters such as First Name, Last Name, Company Name, City & Country. Users will be able to view the Public Profile of users after searching them. The portal allow users to login, request for forgotten password and Update their profile information.

The overview of the project is as below

There are 2 types of users in this Community portal. They are

1. Software Programmer
2. Administrator

Software Programmer should be able to perform following functions in the portal

1. Allow the programmers to register in the portal, show a thank you page & send a registration confirmation email.
2. Search & Find Other Programmers after login and view their profile.
3. Provide Login Page
4. Provide password retrieval functionality.
5. Update their Profile after logging in.

**Following Functionality is part of Database Design & will be enhanced as features as required in the Capstone Project, depending on time availability.**

1. Send Messages to Each Other on the Portal
2. Create Threads & Post Replies to a Thread
3. Post Job Opportunities in the Portal

## Administrator should be able to perform following functions in the portal

## Administer user data.

## Send bulk email inviting programmers to register on the community portal

## The portal consist of the following Key pages (For Reference)

## Community Portal Home Page

## Registration Page

## Registration Confirmation Page (with Registration Confirmation Email)

## Login Page

## Reset Password Page

## Reset Password Confirmation Page

## Update Profile Page

## Search Users Page

## List Search Results

## Public Profile Page

## *Pages for enhancement (depending on time availability)*

## *Send Messages*

## *Read Messages*

## *Post thread in Message Board*

## *List threads in Message Board*

## *Read A Thread*

## *List Job Opportunities & Responses*

## *Post Job Opportunities*

1. *Apply for Jobs*

You will be required to:

Create a testing plan document for the community portal project. The document should include the following sections:

1. Test Metrics: Identify quantitative measures used to estimate the progress, quality, productivity and health of the software testing process
2. Test Schedule: Create a schedule that outlines the timeline for different testing activities.
3. Test Cases: Develop a set of test cases that cover different scenarios and functionalities of the community portal.
4. Test Data: Identify the test data required for executing the test cases.
5. Test Execution: Explain **how** the test cases will be executed, including the test environment setup and execution process.

## Evidence to exhibit

* Test Plan with test cases

# **1.Test Metrics**

|  |  |
| --- | --- |
| TEST METRIC | TESTING PHASE |
| Number of test cases | Test Development Phase |
| Number of Test cases Passed | Test Execution Phase |
| Number of Test cases Executed | Test Execution Phase |
| Number of Test cases Failed | Test Execution Phase |
| Number of Test cases under Investigation | Test Development Phase |
| Number of Test cases Blocked | Test Development Phase / Execution Phase |
| Number of Test cases Re- executed | Regression Phase |
| Number of First run Failures | Test Execution Phase |
| Total Executions | Test Reporting Phase |
| Total Passes and Failures | Test Reporting Phase |
| Test case Execution time | Test Reporting Phase |
| Test Execution time | Test Reporting Phase |

Defect Rate: Total No of test 14

|  |  |
| --- | --- |
| Num. of Test cases Passed | 14 |
| Num. of Test Coverage | 14 |
| Num. of Defects corrected | 14 |
| Num. of Test cases Blocked | 0 |
| Num. of Rework | 0 |
| Num. of Test Effectiveness | 100% |

# **Test Schedule**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Sno | Test Description | Optimal scheduling Time (hours) | Start Date | End Date |
|  | UAT Test | 0.5 | 22-7-2023 | 22-7-2023 |
|  | Compatibility Test | 0.5 | 22-7-2023 | 22-7-2023 |
|  | Performance Test | 0.5 | 22-7-2023 | 22-7-2023 |
|  | Portability Test | 0.5 | 22-7-2023 | 22-7-2023 |

# **Test Cases**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 1.  Functional Testing - UAT test cases | | |  |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Expected Result** |
| 1 | TS001 | Identify whether 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 the user can successfully 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 successfully 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 | Make sure users can accurately edit their profile information. | Functional Testing | The database should correctly store the modified data. |
| 5 | TS005 | Find different programmers 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 Google Chrome. | 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. | 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. | 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** | **Technique** | **Expected Result** |
| 9 | TS009 | Identify if the index load time is within 10 – 15 | 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 | 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 | 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** | **Technique** | **Expected Result** |
| 12 | TS012 | 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. |
| 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 Mobile. | Portability Testing | The portal should have a responsive look and feel, which scales down to a mobile view. |

# **Test Data**

## **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: [korekapi@gmail.com](mailto:korekapi@gmail.com)  Pass: obatnyamuk | 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: [korekapi@gmail.com](mailto:korekapi@gmail.com)  Pass: obatnyamuk | 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: [korekapi@gmail.com](mailto:korekapi@gmail.com)  Pass: garpit123 | 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:  61236809612 | 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. | Pratama | 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 |

# **Test Execution**

**Test Date:** 22 July 2023

**Test Environment:** Chrome, Edge, and Firefox browsers. Mobile Simulator Extension.

**Test Scope:** ABC Jobs Portal

**Test Objective:** This test aims to verify the functionality, compatibility, and performance of the ABC Jobs Portal across different browsers (Chrome et al.).

**Test Cases to Execute:**

1. TS001 - Identify whether the user can log in to the portal with registered credentials.
2. TS002 - Identify whether the user can successfully register or not.
3. TS003 - Identify whether the user can successfully use the forgot password feature.
4. TS004 - Make sure users can accurately edit their profile information.
5. TS005 - Find different programmers by using the search function.
6. TS006 - Identify if the Result portal works as intended on Google Chrome.
7. TS007 - Identify if the Result portal works as intended on Microsoft Edge.
8. TS008 - Identify if the Result portal works as intended on Mozilla Firefox.
9. TS009 - Identify if the index load time is within 10 – 15 seconds.
10. TS010 - Identify if the login page load time is within 10 – 15 seconds.
11. TS011 - Identify if the register page load time is within 10 – 15 seconds.
12. TS012 - Identify whether the portal functionality works as intended using various platforms and Desktop.
13. TS013 - Identify whether the portal functionality works as intended using various platforms and Tablets.
14. TS014 - Identify whether the portal functionality works as intended using various platforms and Mobile.

**Test Execution Steps:**

1. Set up the test environment with the latest Chrome, Edge, and Firefox browser version on the testing machine.
2. Ensure the ABC Jobs Portal is deployed and accessible at the designated URL.
3. Create necessary test accounts or ensure the accounts with predefined test data are available for testing.
4. Execute each test case in the specified order, recording the actual results and any observations during the test execution.
5. For cross-browser testing (TS006, TS007, TS008), open the ABC Jobs Portal URL in each designated browser and verify the look, feel, functionality, and responsiveness.
6. For performance testing (TS009, TS010, TS011), measure the page load times using appropriate browser developer tools.
7. Record any defects or issues encountered during the test execution and report them to the development team.
8. Verify that the expected results match the actual results for each test case.
9. Document the test execution results, including any deviations, and prepare a test execution report.

**Test Execution Outcome:**

The test execution is successful if all test cases produce the expected results and meet the defined criteria for pass/fail. Any identified defects should be reported, and the development team should address them accordingly.

**Test Completion:**

Check the test execution report after completion to ensure all the test goals have been accomplished. If necessary, give the development team comments for any necessary corrections or enhancements.