A picture containing text, clipart

Description automatically generated

**Formative Assignment 2**

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

|  |  |
| --- | --- |
| Learner Name | Muhammad Kemal |

|  |
| --- |
| Learner declaration |
| I certify that the work submitted for this assignment is my own and research sources are fully acknowledged.  Student signature:  Date: 21-07-2023 |

## Purpose of this assignment

To demonstrate your capabilities in the following areas:

* Preparing Test Plan for 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\_A2.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. Introduction: Provide an overview of the project and its objectives.
2. Test Scope: Define the scope of testing, including the specific functionalities and features to be tested.
3. Test Strategy: Describe the overall approach and strategy for testing the community portal.
4. Test Environment: Specify the hardware, software, and testing tools required for the testing process.
5. \*Test Schedule: Create a schedule that outlines the timeline for different testing activities.
6. \*Test Cases: Develop a set of test cases that cover different scenarios and functionalities of the community portal.
7. \*Test Data: Identify the test data required for executing the test cases.
8. \*Test Execution: Explain how the test cases will be executed, including the test environment setup and execution process.
9. Defect Reporting: Define the process for reporting and tracking defects encountered during testing.
10. Risk Assessment: Identify potential risks and mitigation strategies for the testing process.
11. Resource Allocation: Determine the resources needed for testing, including the number of testers and their roles.
12. Test Deliverables: Specify the expected deliverables from the testing process, such as test reports and documentation.
13. Approval: Include a section for approval and review of the testing plan.

*\* Items e – h (above) / 5 – 8 (in the template below) will be covered in Assignment 3*

## Evidence to exhibit

* Phase Test Plan

# **1. Introduction**

## **A. Project Overview**

ABC Job Portal is a web application designed to bring together software developers from various backgrounds and regions onto a unified platform. This community-driven portal will enable users to register, search for other developers based on different parameters, view public profiles, and interact with each other in a professional setting. The project will be developed using Spring Boot for the backend and React for the frontend, ensuring a seamless and responsive user experience

## **B. Objectives**

The community portal is for Software Developers to interact among themselves. As part of this project we need to plan types of testing, create a phase test plan and schedule them optimally and execute the test scripts. Verify the results after testing.

Validate User Registration and Authentication: To ensure that users can successfully register, login, and recover forgotten passwords securely, without encountering errors or data inconsistencies.

Verify User Profile Management: : To validate that users can update their profiles with accurate information, and the portal securely handles and saves large data inputs without data loss or corruption.

Test User Search Functionality: To verify that the search functionality provides accurate and relevant results based on different search parameters, and displays appropriate messages for various search scenarios.

Ensure Proper Public Profile Viewing: To test that the public profiles of users are displayed correctly, presenting accurate information and preventing unauthorized access to private profile details.

Validate Responsiveness and Compatibility: To ensure that the website's design is responsive across different devices and screen sizes, and that it functions correctly on major web browsers, providing a consistent user experience.

Perform Security Testing: To identify and address potential security vulnerabilities, ensuring that sensitive user data is protected and that the portal is resistant to common security threats.

Test Error Handling and Validation: To verify that the portal handles errors gracefully, displaying appropriate error messages to users and preventing invalid data submissions.

Check Performance and Load Handling: To assess the portal's performance under typical user loads, ensuring responsiveness and stability during peak traffic periods.

Assess Usability and User Experience: To conduct usability testing to evaluate the website's ease of use, intuitive navigation, and overall user experience, collecting user feedback for possible improvements.

# **2. Test Scope**

The testing phase of the Software Developers Community Portal project is crucial to ensure the website functions as intended, meets the specified requirements, and delivers a seamless user experience. The following test scope outlines the various aspects that will be covered during the testing process:

2.1. User Registration and Authentication:

* Verify that users can successfully register on the portal with valid information.
* Test the registration process for handling invalid or duplicate data entries.
* Ensure that the authentication mechanism securely grants access to registered users only.
* Validate the "Forgot Password" feature to reset passwords and regain access to accounts.

2.2. User Profile Management:

* Test the functionality to update user profiles and verify that changes are saved correctly.
* Validate the restriction of unauthorized access to other users' profile update pages.
* Ensure the portal handles profile updates containing special characters and large data inputs gracefully.

2.3. User Search Functionality:

* Test the search feature using various parameters (First Name, Last Name, Company Name, City, Country).
* Verify that search results are accurate and relevant to the search criteria.
* Ensure the search functionality displays appropriate messages for no search results or errors.

2.4. Public Profile Viewing:

* Test the public profile display to ensure that user information is presented correctly.
* Validate that unauthorized users cannot access private profile information.
* Check that all public profile elements (skills, employment history, etc.) are correctly displayed.

2.5. Responsiveness and Compatibility:

* Test the website's responsiveness across different devices and screen sizes (desktop, tablet, mobile).
* Validate compatibility with major web browsers (Chrome, Firefox, Safari, Edge, etc.).
* Check for consistent behaviour and layout across different platforms.

2.6. Security Testing:

* Perform security testing to identify and address potential vulnerabilities.
* Ensure that sensitive user data, such as passwords, is stored securely using encryption techniques.
* Validate protection against common security threats like SQL injection, cross-site scripting (XSS), etc.

2.7. Error Handling and Validation:

* Test error handling scenarios, such as incorrect form submissions and invalid inputs.
* Validate that appropriate error messages are displayed to users when required.
* Verify that form validation prevents users from submitting incomplete or erroneous data.

2.8. Performance and Load Testing:

* Test the portal's performance under typical user loads to ensure responsiveness and stability.
* Validate that the website performs efficiently even during peak traffic periods.

2.9. Usability and User Experience:

* Conduct usability testing to assess the website's ease of use and intuitive navigation.
* Collect user feedback to identify potential improvements for a better user experience.

# **3. Test Strategy**

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 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. |

# **4. Test Environment**

* 1. Hardware
     1. Processor:
     2. RAM: 4 GB or more.
     3. HDD: 60 GB or more.
  2. Software
     1. OS: Windows 10
     2. Spring boot
     3. VSCODE
     4. React
     5. MySQL Workbench 8.0
  3. Testing Tools

Chrome Developer Tools

**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 | 19-7-2023 | 19-7-2023 | 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 | 22-7-2023 | 22-7-2023 | 1 | Tester |
| 10 | Re-execute Tests as Needed | 22-7-2023 | 22-7-2023 | 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. \*Test Schedule
2. \*Test Cases
3. \*Test Data
4. \*Test Execution
   1. Test Environment Setup
   2. Test Execution Process

# **Defect Reporting**

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Deflect Id | Test Id | Test Scenario | Severity | Priority | Detected By | Date Detected | Desc. Of Detect |
| D0001 | TS0003 | User Forgot Password | High | High | UAT Tester | 21-07-2023 | The user cannot create a new password. |
| D0002 | TS0004 | Update Profile User | High | High | UAT Tester | 21-07-2023 | The data cannot add or update. |
| D0003 | TS0005 | Search Other Profile | Medium | Medium | UAT Tester | 21-07-2023 | The other profile is not showing. |

# **Risk Assessment**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 1.  Functional Testing - UAT test cases | | |  |  |  |  |
| Test Id | **Test Scenario** | **Test Objective** | **Technique** | **Pro** | **Risk** | **Expected Result** |
| 1 | TS001 | Determine if the user has registered credentials and may access the portal. | Functional Testing | High | May login without details. | Users who make an effort at authenticating will be admitted to the site and given the default authorisation of member or user. |
| 2 | TS002 | Determine whether a user may register successfully or not. | Functional Testing | Medium | The user will not be able to use the website. | After registering, individuals can log in to the system with the registered credentials. |
| 3 | TS003 | Determine if the user is able to use the forgot password option. | Functional Testing | High | The password will forget forever, and the user cannot log in. | The user should be able to input a new password and confirm the password change on the Reset Password page. |
| 4 | TS004 | Make sure users can accurately edit their profile information. | 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 different programmers by using the search function. | Functional Testing | Low | Other programmers 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 | | Determine whether Google Chrome properly renders the Result portal. | | 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 | | Determine whether Microsoft Edge's Result portal functions as anticipated. | | 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 | | Determine whether Mozilla Firefox satisfies the result portal's requirements. | | 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 9 – 14 | | High | Users will lose interest and not want to use the application. | Performance Testing | The index page will load in less than 14 seconds. |
| 10 | TS010 | | Identify if the login page load time is within 9-11 | | High | Users will lose interest and not want to use the application. | Performance Testing | The login page will load in less than 11 seconds. |
| 11 | TS011 | | Identify if the register page load time is within 11 – 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 | | Using different platforms and Desktop, determine whether the portal's functionality functions as anticipated. | | High | In a Desktop view, users are unable to fully utilize the function on the portal page. | Portability Testing | The portal ought to offer a Desktop view and a responsive appearance and feel. |
| 13 | TS013 | | Test the functioning of the portal on multiple platforms and tablets to see if it functions as planned. | | 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 | | Check the operation of the portal on multiple platforms and mobile devices to see if it functions as planned. | | 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. |

# **Resource Allocation**

|  |  |
| --- | --- |
| Task | Resource Name |
| Sprint 1 |  |
| Determine the scope and requirements. | Muhammad Kemal |
| Create project plans and schedules. | Muhammad Kemal |
| Analysis and make The Flowchart. | Muhammad Kemal |
| Design the database schema. | Muhammad Kemal |
| Create wireframes and Storyboard. | Muhammad Kemal |
| Develop Register & Login. | Muhammad Kemal |
| Develop Edit Profile Users. | Muhammad Kemal |
| Testing Phase. | Muhammad Kemal |
| Sprint 2 |  |
| Develop Search & Search Result. | Muhammad Kemal |
| Implement View Others' Profile. | Muhammad Kemal |
| UAT Testing. | Muhammad Kemal |
| UT Testing. | Muhammad Kemal |
| Develop administrator functions. | Muhammad Kemal |
| Sprint 3 |  |
| Testing Phase. | Muhammad Kemal |
| Start Documentation. | Muhammad Kemal |
| Review and Final Adjustments |  |
| Final testing and bug fixes. | Muhammad Kemal |
| Documentation review and finalization. | Muhammad Kemal |

# **Test Deliverables**

|  |  |
| --- | --- |
| Test | Expectation |
| Functional Testing - UAT test cases | User Acceptance Testing (UAT) will be used throughout the functional testing phase to make sure that the community portal's features and functioning adhere to predetermined standards. It was anticipated that all UAT test cases would be run, and that any faults found would be addressed and fixed. |
| Cross-browser compatibility testing | To ensure that the community portal works consistently and accurately across different web browsers including Chrome, Firefox, and Edge, cross-browser testing will be done. In all supported browsers, the Portal is anticipated to function without any significant problems. |
| Performance Efficiency - Load Testing | To evaluate how well the Portal performs under varied user loads, load testing will be done. The Portal aims to support the anticipated number of concurrent users without noticeably degrading performance or crashing. |
| Portability - Multi platforms | The community portal's compatibility with various platforms, including desktop computers, laptops, tablets, and mobile devices, will be tested during the portability testing phase. The Portal ought to function properly across a range of gadgets and screen sizes and be responsive. |

# **Approval**

|  |  |  |
| --- | --- | --- |
| Deliverable | Approver | Signed Off Date |
| 1. Test Scope | Arvinder Kaur | 21-07-2023 |
| 2. Test Strategy | Arvinder Kaur | 21-07-2023 |
| 3. Test environment | Arvinder Kaur | 21-07-2023 |
| 4. Test Plan Phase | Arvinder Kaur | 21-07-2023 |
| 5. Defect Reporting | Arvinder Kaur | 21-07-2023 |
| 6. Risk Assessment | Arvinder Kaur | 21-07-2023 |
| 7. Resource Allocation | Arvinder Kaur | 21-07-2023 |