***Enhanced Job and Candidate Application***

***Test Plan***

computech logo



Version: 1.0

03/24/2015

**Prepared by:**

Simul Kadakia

Wesley Trescott

Gagandeep Singh

Contents

[1. Introduction 1](#_Toc415152079)

[1.1 Purpose 1](#_Toc415152080)

[1.2 Scope 1](#_Toc415152081)

[1.3 Definitions, Acronyms and Abbreviations 1](#_Toc415152082)

[1.4 References 1](#_Toc415152083)

[2. Functional Testing 2](#_Toc415152084)

[2.1 Test Approach 2](#_Toc415152085)

[2.2 Test Pass / Fail Criteria 2](#_Toc415152086)

[2.3 Test Entry / Exit Criteria 2](#_Toc415152087)

[2.4 Test Deliverables: 2](#_Toc415152088)

[2.5 Test Suspension / Resumption Criteria 25](#_Toc415152089)

[2.6 Test Environmental Needs 26](#_Toc415152090)

[3. Nonfunctional Testing 26](#_Toc415152091)

[3.1 Test Risks / Issues 26](#_Toc415152092)

[3.2 Items To Be Tested 26](#_Toc415152093)

[3.3 Test Approach 26](#_Toc415152094)

[3.4 Test Pass / Fail Criteria 26](#_Toc415152095)

[3.5 Test Entry / Exit Criteria 27](#_Toc415152096)

[3.6 Test Deliverables 27](#_Toc415152097)

[3.7 Test Suspension / Resumption Criteria 28](#_Toc415152098)

[3.8 Test Environmental Needs 28](#_Toc415152099)

[4. Compatibility Testing 28](#_Toc415152100)

[4.1 Test Risks / Issues 28](#_Toc415152101)

[4.2 Items To Be Tested 28](#_Toc415152102)

[4.3 Test Approach 29](#_Toc415152103)

[4.4 Test Pass / Fail Criteria 30](#_Toc415152104)

[4.5 Test Entry / Exit Criteria 30](#_Toc415152105)

[4.6 Test Deliverables 30](#_Toc415152106)

[4.7 Test Suspension / Resumption Criteria 36](#_Toc415152107)

[4.8 Test Environmental Needs 36](#_Toc415152108)

[5. System Testing 36](#_Toc415152109)

[5.1 Test Risks / Issues 37](#_Toc415152110)

[5.2 Items To Be Tested 37](#_Toc415152111)

[5.3 Test Approach 37](#_Toc415152112)

[5.4 Test Pass / Fail Criteria 37](#_Toc415152113)

[5.5 Test Entry / Exit Criteria 37](#_Toc415152114)

[5.6 Test Suspension / Resumption Criteria 38](#_Toc415152115)

[5.7 Test Deliverables 38](#_Toc415152116)

[5.8 Test Environmental Needs 40](#_Toc415152117)

[6. Integration Testing 40](#_Toc415152118)

[6.1 Test Risks / Issues 40](#_Toc415152119)

[6.2 Items To Be Tested 40](#_Toc415152120)

[6.3 Test Approach 41](#_Toc415152121)

[6.4 Test Pass / Fail Criteria 41](#_Toc415152122)

[6.5 Test Entry / Exit Criteria 41](#_Toc415152123)

[6.6 Test Suspension / Resumption Criteria 41](#_Toc415152124)

[6.7 Test Environmental Needs 42](#_Toc415152125)

[7. Performance and Load Testing 42](#_Toc415152126)

[7.1 Test Risks / Issues 42](#_Toc415152127)

[7.2 Items To Be Tested 42](#_Toc415152128)

[7.3 Test Approach 42](#_Toc415152129)

[7.4 Test Pass / Fail Criteria 42](#_Toc415152130)

[7.5 Test Entry / Exit Criteria 43](#_Toc415152131)

[7.6 Test Deliverables 43](#_Toc415152132)

[7.7 Test Suspension / Resumption Criteria 43](#_Toc415152133)

[7.8 Test Environmental Needs 44](#_Toc415152134)

[8. Security Testing 44](#_Toc415152135)

[8.1 Test Risks / Issues 44](#_Toc415152136)

[8.2 Items To Be Tested 44](#_Toc415152137)

[8.3 Test Approach 44](#_Toc415152138)

[8.4 Test Pass / Fail Criteria 45](#_Toc415152139)

[8.5 Test Entry / Exit Criteria 45](#_Toc415152140)

[8.6 Test Deliverables 45](#_Toc415152141)

[8.7 Test Suspension / Resumption Criteria 48](#_Toc415152142)

[8.8 Test Environmental Needs 48](#_Toc415152143)

[9. User Acceptance Testing 48](#_Toc415152144)

[9.1 Test Risks / Issues 48](#_Toc415152145)

[9.2 Items To Be Tested 49](#_Toc415152146)

[9.3 Test Approach 49](#_Toc415152147)

[9.4 Test Pass / Fail Criteria 49](#_Toc415152148)

[9.5 Test Entry / Exit Criteria 49](#_Toc415152149)

[9.6 Test Deliverables 49](#_Toc415152150)

[9.7 Test Suspension / Resumption Criteria 53](#_Toc415152151)

[9.8 Test Environmental Needs 53](#_Toc415152152)

[10. Unit Testing 54](#_Toc415152153)

[10.1 Test Risks / Issues 54](#_Toc415152154)

[10.2 Items To Be Tested 54](#_Toc415152155)

[10.3 Test Approach 54](#_Toc415152156)

[10.4 Test Pass / Fail Criteria 55](#_Toc415152157)

[10.5 Test Entry / Exit Criteria 55](#_Toc415152158)

[10.6 Test Deliverables 55](#_Toc415152159)

[10.7 Test Suspension / Resumption Criteria 66](#_Toc415152160)

[10.8 Test Environmental Needs 66](#_Toc415152161)

# Introduction

This section gives a scope description and overview of everything included in this document. Also, the purpose for this document is described and a list of abbreviations and definitions is provided.

## Purpose

The purpose of this Software Test Plan document is to provide a detailed description of the testing plan of the ‘Enhanced Job and Candidate Application’ system. It will also provide specific information about the compatibility, functional, non-functional, integration, system and user acceptance testing.

## Scope

Enhanced Job and Search Candidate is an application developed by three students at Wayne State University for Computech Corporation. The goal of the application is to provide a web application service to job seekers to search for available jobs at Computech Corporation and apply to those that they are interested in. The application will provide filtering options to reduce the jobs displayed based on certain criteria. Job seekers will be able to store their profile information and resume path (resume will be stored in a folder) in the database, which can be used when they are applying for a job using a web interface. The application will also provide functionalities to an admin user to deactivate and delete users abusing the system.

## Definitions, Acronyms and Abbreviations

* EJCA - Enhanced Job and Candidate Application
* User – Job seeker who uses the application
* Admin – Admin/administrator who manages the users
* Admin portal - Part of the web application that provides special facilities to Admin
* Front End - The part of the application the user interacts with
* Back End – The part of application that manages data and is managed by developers.
* UI – User Interface which is the front end of the application
* Server –Machine that will host the web application as well as database.
* GUID – Global Unique Identifier

## References

* Microsoft ASP.NET MVC - <http://www.asp.net/mvc>
* Microsoft SQL Server - <https://msdn.microsoft.com/en-us/sqlserver/aa336270.aspx>
* Razor - <http://www.asp.net/web-pages/overview/getting-started/introducing-razor-syntax-(c)>
* JQuery - <http://jquery.com/>
* Bootstrap – <http://getbootstrap.com>

# 2. Functional Testing

### 2.1 Test Approach

Follow the test steps as described in the test cases.

### 2.2 Test Pass / Fail Criteria

If observed results are the same as expected results, the test case will pass; otherwise, test case will fail.

### 2.3 Test Entry / Exit Criteria

Entry Criteria:

* Test environment is ready to use.

Exit criteria:

* All use cases are tested.
* All test cases pass successfully.
* Defects found are documented to find alternate solution.

### 2.4 Test Deliverables:

|  |  |
| --- | --- |
| ID | FT\_1 |
| Item to Test | Search Jobs |
| Pre-Conditions | None |
| Test Steps | 1. Open website by entering www.computech.com/careers in the address bar. 2. Click on Search Open Positions |
| Expected Results | System displays all available jobs |
| Priority | High |
| Pass/Fail | Pass if system redirects user to JobSearch page. |

|  |  |
| --- | --- |
| ID | FT\_2 |
| Item to Test | Register |
| Pre-Conditions | None |
| Test Steps | 1. Click on Users button on header. 2. Click on Register link button. 3. Fill in First name - John,   Last name - Doe,  Email ID – john.doe@email.com,  Password - 123456,  Confirm Password - 123456.   1. Click on Register. |
| Expected Results | System displays confirmation page with a message about verification email sent to user provided email address. |
| Priority | High |
| Pass/Fail | Pass if system redirects user to Register page. |

|  |  |
| --- | --- |
| ID | FT\_2.1 |
| Item to Test | Register Fails – Empty First Name |
| Pre-Conditions | None |
| Test Steps | 1. Click on Users button on header. 2. Click on Register link button. 3. Fill in Last name - Doe,   Email ID – john.doe@email.com,  Password - 123456,  Confirm Password - 123456.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.2 |
| Item to Test | Register Fails – Empty Last Name |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Email ID – john.doe@email.com,  Password - 123456,  Confirm Password - 123456.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.3 |
| Item to Test | Register Fails – Empty email address |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name -Doe  Password - 123456,  Confirm Password - 123456.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.4 |
| Item to Test | Register Fails – Empty password |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name –Doe,  Email address – john.doe@email.com  Confirm Password - 123456.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.5 |
| Item to Test | Register Fails – Empty confirm password |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name –Doe,  Email address – john.doe@email.com  Password - 123456.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.6 |
| Item to Test | Register Fails – Invalid Password Length |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name –Doe,  Email address – john.doe@email.com  Password - 12345  Confirm Password – 12345.   1. Click on Register. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.7 |
| Item to Test | Register Fails – Password and Confirm Password do not match |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name –Doe,  Email address – john.doe@email.com  Password - 123456  Confirm Password – 12345.   1. Click on Register. |
| Expected Results | System displays error message about password and confirm password not matching. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_2.8 |
| Item to Test | Register Fails – Email previously registered |
| Pre-Conditions | None |
| Test Steps | 1. Click on Register link button. 2. Fill in First name - John,   Last name –Doe,  Email address – john.doe@email.com  Password - 123456  Confirm Password – 123456.   1. Click on Register. |
| Expected Results | System displays error message about email being registered before. |
| Priority | High |
| Pass/Fail | Pass if system displays error and keeps information in text boxes except password. |

|  |  |
| --- | --- |
| ID | FT\_3 |
| Item to Test | Email verification |
| Pre-Conditions | User has logged into their email account |
| Test Steps | 1. Click on Computech Capstone email verification email. 2. Click on the link in the email. |
| Expected Results | System displays message about verifying the email and then redirect to log in page. |
| Priority | High |
| Pass/Fail | Pass if system redirects user to confirmation page and redirecting user to Login page. |

|  |  |
| --- | --- |
| ID | FT\_3.1 |
| Item to Test | Email verification Fails – email already verified |
| Pre-Conditions | User has logged into their email account and has already verified their email address. |
| Test Steps | 1. Click on Computech Capstone email verification email. 2. Click on the link in the email. |
| Expected Results | System displays message about email already verified and then redirect to log in page. |
| Priority | High |
| Pass/Fail | Pass if system redirects user to confirmation page and redirecting user to login page. |

|  |  |
| --- | --- |
| ID | FT\_4 |
| Item to Test | Login |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill in email address – john.doe@email.com password - 123456. 4. Click on Login |
| Expected Results | User dashboard is displayed. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | FT\_4.1 |
| Item to Test | Login Fails – incorrect email address |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill in email address – john@email.com,   Password - 123456   1. Click on Login |
| Expected Results | System displays error message about incorrect login data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_4.2 |
| Item to Test | Login Fails – incorrect password |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill in email address – john.doe@email.com,   Password - 12345   1. Click on Login |
| Expected Results | System displays error message about incorrect login data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_4.3 |
| Item to Test | Login Fails – empty email address |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill Password - 12345 4. Click on Login |
| Expected Results | System displays error message about incorrect login data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_4.4 |
| Item to Test | Login Fails – empty password |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill email – john.doe@email.com, 4. Click on Login |
| Expected Results | System displays error message about incorrect login data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_4.5 |
| Item to Test | Login Fails – email not verified |
| Pre-Conditions | User has registered their email address but not verified their email. |
| Test Steps | 1. Click on Users dropdown list in the header. 2. Click on Login link button 3. Fill email – john.doe@email.com,   Password - 123456   1. Click on Login |
| Expected Results | System displays error message about incorrect login data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_5 |
| Item to Test | Forgot Password |
| Pre-Conditions | User has registered their email ID. |
| Test Steps | 1. Click on Users dropdown list button in the header. 2. Click on Login button. 3. Click on ‘Click here if you forgot your password’ link button. 4. Fill in email address – john.doe@email.com. 5. Click on Submit button. |
| Expected Results | System displays a confirmation message about sending email with forgot password link to user. |
| Priority | High |
| Pass/Fail | Pass if system displays confirmation message to user. |

|  |  |
| --- | --- |
| ID | FT\_5.1 |
| Item to Test | Forgot Password Fails – empty email |
| Pre-Conditions | User has registered their email ID. |
| Test Steps | 1. Click on Users dropdown list button in the header. 2. Click on Login button. 3. Click on ‘Click here if you forgot your password’ link button. 4. Click on Submit button. |
| Expected Results | System displays error message about empty fields |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_5.2 |
| Item to Test | Change Password without login |
| Pre-Conditions | User has requested change password link. |
| Test Steps | 1. Click on the link in the Computech Corporation forgot password email. 2. Fill in password – 111111   confirm password - 111111.   1. Click on Submit. |
| Expected Results | System displays confirmation message and redirects user to login page |
| Priority | High |
| Pass/Fail | Pass if system redirect user to change password page. |

|  |  |
| --- | --- |
| ID | FT\_5.3 |
| Item to Test | Change Password without login Fails – invalid password length |
| Pre-Conditions | User has requested change password link. |
| Test Steps | 1. Click on the link in the Computech Corporation forgot password email. 2. Fill in password – 11111   confirm password - 11111.   1. Click on Submit. |
| Expected Results | System displays error message about required fields. |
| Priority | High |
| Pass/Fail | Pass if system displays error message to user. |

|  |  |
| --- | --- |
| ID | FT\_5.4 |
| Item to Test | Change Password without login Fails – password and confirm password not matching |
| Pre-Conditions | User has requested change password link. |
| Test Steps | 1. Click on the link in the Computech Corporation forgot password email. 2. Fill in password – 111111   confirm password - 111.   1. Click on Submit. |
| Expected Results | System displays error message about password and confirm password not matching. |
| Priority | High |
| Pass/Fail | Pass if system displays error message to user. |

|  |  |
| --- | --- |
| ID | FT\_6 |
| Item to Test | Change Password after log in |
| Pre-Conditions | User has logged in. |
| Test Steps | 1. Click on the hamburger menu button in the header 2. Click on Change password link button 3. Fill in Current password – 123456,   new password – abc123  confirm new password – abc123.   1. Click on Submit |
| Expected Results | System displays confirmation message. |
| Priority | High |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | FT\_6.1 |
| Item to Test | Change Password after log in Fails – invalid current password |
| Pre-Conditions | User has logged in. |
| Test Steps | 1. Click on the hamburger menu button in the header 2. Click on Change password link button 3. Fill in Current password - 123   new password – abc123  confirm new password – abc123.   1. Click on Submit |
| Expected Results | System displays error message about invalid data. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_6.2 |
| Item to Test | Change Password after log in Fails – invalid password length |
| Pre-Conditions | User has logged in. |
| Test Steps | 1. Click on the hamburger menu button in the header 2. Click on Change password link button 3. Fill in Current password - 123   new password – abc12  confirm new password – abc12.   1. Click on Submit |
| Expected Results | System displays error message about invalid password length. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_6.3 |
| Item to Test | Change Password after log in Fails – new password and confirm new password not matching. |
| Pre-Conditions | User has logged in. |
| Test Steps | 1. Click on the hamburger menu button in the header 2. Click on Change password link button 3. Fill in Current password - 123   new password – abc123  confirm new password – abc.   1. Click on Submit |
| Expected Results | System displays error message about password and confirm password not matching. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_7 |
| Item to Test | Apply |
| Pre-Conditions | User has logged in. |
| Test Steps | 1. Click on Search open positions. 2. Click on Details button 3. Click on Apply button 4. Verify user information 5. Click on Submit |
| Expected Results | System displays confirmation message and redirects user to user dashboard page. |
| Priority | High |
| Pass/Fail | Pass if system displays confirmation message and redirecting user to user dashboard page. |

|  |  |
| --- | --- |
| ID | FT\_7.1 |
| Item to Test | Apply Fails – already applied |
| Pre-Conditions | User has logged in and already applied to job. |
| Test Steps | 1. Click on Search open positions. 2. Click on Details button 3. Click on Apply button 4. Verify user information 5. Click on Submit |
| Expected Results | System displays error message about already applied and redirects user to user dashboard page. |
| Priority | High |
| Pass/Fail | Pass if system displays error message and redirecting user to user dashboard page. |

|  |  |
| --- | --- |
| ID | FT\_8 |
| Item to Test | Filter jobs by location, customer |
| Pre-Conditions | None. |
| Test Steps | 1. Click on Search open positions. 2. Select parameters   City – Detroit,  Customer - Computech   1. Click on Filter Jobs button |
| Expected Results | System displays jobs based on criteria |
| Priority | Medium |
| Pass/Fail | Pass if system displays jobs matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_8.1 |
| Item to Test | Filter jobs by customer |
| Pre-Conditions | None. |
| Test Steps | 1. Click on Search open positions. 2. Select parameter   Customer - Computech   1. Click on Filter Jobs button |
| Expected Results | System displays jobs based on criteria |
| Priority | Medium |
| Pass/Fail | Pass if system displays jobs matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_8.2 |
| Item to Test | Filter jobs by city |
| Pre-Conditions | None. |
| Test Steps | 1. Click on Search open positions. 2. Select parameter   City - Detroit   1. Click on Filter Jobs button |
| Expected Results | System displays jobs based on criteria |
| Priority | Medium |
| Pass/Fail | Pass if system displays jobs matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_9 |
| Item to Test | Profile Information |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   Street – 1 Cass Ave,  City – Detroit,  State – MI,  Country – USA,  Phone Number – 1111111111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays confirmation message. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | FT\_9.1 |
| Item to Test | Profile Information Fails – street empty |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   City – Detroit,  State – MI,  Country – USA,  Phone Number – 1111111111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays error message about required fields. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_9.2 |
| Item to Test | Profile Information Fails – city empty |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   Street – 1 Cass Ave,  State – MI,  Country – USA,  Phone Number – 1111111111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays error message about required fields. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_9.3 |
| Item to Test | Profile Information Fails –state empty |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   Street – 1 Cass Ave  City – Detroit,  Country – USA,  Phone Number – 1111111111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays error message about required fields. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_9.4 |
| Item to Test | Profile Information Fails – country empty |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   Street – 1 Cass Ave  City – Detroit,  State – MI,  Phone Number – 1111111111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays error message about required fields. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_9.5 |
| Item to Test | Profile Information Fails – invalid phone number format |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile 3. Fill in   Street – 1 Cass Ave  City – Detroit,  State – MI,  Country – USA,  Phone Number – 111-111-1111,  Total years of experience – 2,  Skills – C#, ASP.NET, SQL   1. Click on Save Information. |
| Expected Results | System displays error message about invalid phone number format. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_10 |
| Item to Test | Upload Resume – pdf file |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile. 3. Click on ‘Click here to upload resume’ link button. 4. Click on Choose file button 5. Select a pdf file. 6. Click on Upload Resume button |
| Expected Results | System displays confirmation message. |
| Priority | High |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | FT\_10.1 |
| Item to Test | Upload Resume – doc file |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile. 3. Click on ‘Click here to upload resume’ link button. 4. Click on Choose file button 5. Select a doc file. 6. Click on Upload Resume button |
| Expected Results | System displays confirmation message. |
| Priority | High |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | FT\_10.2 |
| Item to Test | Upload Resume Fails – html file |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile. 3. Click on ‘Click here to upload resume’ link button. 4. Click on Choose file button 5. Select a html file 6. Click on Upload Resume button |
| Expected Results | System displays error message about ‘unsupported format’. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_10.3 |
| Item to Test | Upload Resume Fails –unsupported file |
| Pre-Conditions | User is logged in. |
| Test Steps | 1. Click on hamburger menu button in the header. 2. Click on Update Profile. 3. Click on ‘Click here to upload resume’ link button. 4. Click on Choose file button 5. Select any file without pdf, doc or docx extension 6. Click on Upload Resume button |
| Expected Results | System displays error message about ‘unsupported format’. |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_11 |
| Item to Test | Admin – Login |
| Pre-Conditions | Admin email has been stored in database. |
| Test Steps | 1. Click on Admin link button on homepage. 2. Fill in   email address – john.doe@computech.com  password - 123456   1. Click on Login |
| Expected Results | Admin dashboard is displayed |
| Priority | High |
| Pass/Fail | Pass if system redirects admin to admin dashboard. |

|  |  |
| --- | --- |
| ID | FT\_11.1 |
| Item to Test | Admin – Login Fails – empty email address |
| Pre-Conditions | Admin email has been stored in database. |
| Test Steps | 1. Click on Admin link button on homepage. 2. Fill in password - 123456 3. Click on Login |
| Expected Results | System displays error message about ‘incorrect login data’ |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_11.2 |
| Item to Test | Admin – Login Fails –invalid email address |
| Pre-Conditions | Admin email has been stored in database. |
| Test Steps | 1. Click on Admin link button on homepage. 2. Fill in email address – john.doe@email.com,   password - 123456   1. Click on Login |
| Expected Results | System displays error message about ‘incorrect login data’ |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_11.3 |
| Item to Test | Admin – Login Fails – incorrect password. |
| Pre-Conditions | Admin email has been stored in database. |
| Test Steps | 1. Click on Admin link button on homepage. 2. Fill in email address – john.doe@computech.com,   password – 12345   1. Click on Login |
| Expected Results | System displays error message about ‘incorrect login data’ |
| Priority | High |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_12 |
| Item to Test | Admin – Search Users |
| Pre-Conditions | Admin is logged in. |
| Test Steps | 1. Fill in user first name – John,   last name - Doe.   1. Click on Search Users |
| Expected Results | Users matching the criteria are displayed. |
| Priority | Medium |
| Pass/Fail | Pass if system displays user matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_12.1 |
| Item to Test | Admin – Search Users by First name only |
| Pre-Conditions | Admin is logged in. |
| Test Steps | 1. Fill in user first name – John, 2. Click on Search Users |
| Expected Results | Users matching the criteria are displayed. |
| Priority | Medium |
| Pass/Fail | Pass if system displays user matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_12.2 |
| Item to Test | Admin – Search Users by last name only |
| Pre-Conditions | Admin is logged in. |
| Test Steps | 1. Fill in user last name - Doe. 2. Click on Search Users |
| Expected Results | Users matching the criteria are displayed. |
| Priority | Medium |
| Pass/Fail | Pass if system displays user matching the criteria. |

|  |  |
| --- | --- |
| ID | FT\_12.3 |
| Item to Test | Admin – Search Users Fails – invalid first name |
| Pre-Conditions | Admin is logged in and user with first name Jane doesn’t exist in database. |
| Test Steps | 1. Fill in user first name – Jane,   Last name - Doe   1. Click on Search Users |
| Expected Results | System displays message about no user found. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_12.4 |
| Item to Test | Admin – Search Users Fails – invalid last name |
| Pre-Conditions | Admin is logged in and user with last name Smith doesn’t exist in database. |
| Test Steps | 1. Fill in user first name - John   last name – Smith   1. Click on Search Users |
| Expected Results | System displays message about no user found. |
| Priority | Medium |
| Pass/Fail | Pass if system displays error message. |

|  |  |
| --- | --- |
| ID | FT\_13 |
| Item to Test | Admin – Deactivate Users |
| Pre-Conditions | Admin is logged in. |
| Test Steps | 1. Click on Deactivate link next to user – John Doe. |
| Expected Results | Confirmation message is displayed. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | FT\_14 |
| Item to Test | Admin – Delete Users |
| Pre-Conditions | Admin is logged in. |
| Test Steps | 1. Click on Delete link next to any user – John Doe. |
| Expected Results | Confirmation message is displayed. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

## 2.5 Test Suspension / Resumption Criteria

Testing will be suspended as soon as any test case fails. An alternate solution will be implemented and testing will then be resumed.

## 2.6 Test Environmental Needs

Testing must be carried out in each of devices and browsers mentioned in the compatibility requirements section above to ensure that environment differences do not affect usability of the web application.

# 3. Nonfunctional Testing

## 3.1 Test Risks / Issues

Thorough testing of nonfunctional aspects such as reliability and availability is necessary so that unanticipated problems such as long system maintenances do not make the application unusable.

## 3.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| NT\_1 | Application Reliability | Job search and recommendation algorithms return correct results within reasonable time |
| NT\_2 | Application Availability | Web application is loaded within a reasonable time in user’s web browser whenever it is requested |

## 3.3 Test Approach

Given that the functionality of the web application has proved to be working correctly, nonfunctional testing will verify that the overall functionality of the site is available and reliable. This is done by two simple user-based tests in which the reliability of the search results and site availability are verified. The reliability test involves examination of search results for logical soundness and the availability test involves timed access of the site from a user device.

## 3.4 Test Pass / Fail Criteria

Both of the tests pass if web pages are loaded in the two second target time, i.e. if the application is available from the webserver. The reliability test may fail if the correct data is not retrieved from the database, which would signal an incorrect query. Also, the availability test may fail if the tester receives an error notification in the web browser. In either case, the bug would have to be fixed and the test would have to be rerun.

## 3.5 Test Entry / Exit Criteria

To run these nonfunctional tests, the web application must first be hosted on the Computech server and available via a URL. The exit criteria for both nonfunctional tests are that they have all passed without errors.

## 3.6 Test Deliverables

|  |  |
| --- | --- |
| ID | NT\_1 |
| Item to Test | Application Reliability |
| Pre-Conditions | The web application is hosted on the Computech server |
| Test Steps | Execute the following ten times:   1. In the job search window select criteria from each filter 2. Execute the search and start stopwatch 3. Compare results to items selected in search window |
| Expected Results | Search results match criteria selected by the user logically AND’ed together; all search results are loaded to the web page within two seconds |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | NT\_2 |
| Item to Test | Application Availability |
| Pre-Conditions | The web application is hosted on the Computech server |
| Test Steps | Execute the following ten times:   1. Type the web application URL (computech.com/careers) into the web browser and press enter 2. Begin stopwatch |
| Expected Results | Web application loads within in an average of two seconds and doesn’t display a connectivity error message |
| Priority | High |
| Pass/Fail |  |

## 3.7 Test Suspension / Resumption Criteria

Testing will be suspended as soon as any bug is found that would make normal use of the application difficult. After a solution to the bug is implemented, testing will resume from the beginning.

## 3.8 Test Environmental Needs

Testing must be carried out in each of devices and browsers mentioned in the compatibility requirements section above to ensure that environment differences do not affect usability of the web application. Also, the user machine must have 1 GB of available RAM, and internet download speeds of at least 2 MBPS.

# 4. Compatibility Testing

## 4.1 Test Risks / Issues

There are no significant risks to the compatibility testing.

## 4.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| CT\_1 | Search Jobs | Users are able to search available job listing. |
| CT\_2 | Register | Users are able to create an account. |
| CT\_3 | Email verification | Users are able to open the link sent in the verification email. |
| CT\_4 | Login | Users are able to login. |
| CT\_5 | Forgot Password | Users are able to request link and change their password. |
| CT\_6 | Change Password | Users are able to change their password after logging in. |
| CT\_7 | Filter Jobs | Users are able to select criteria and filter jobs. |
| CT\_8 | Profile Information | Users are able to access their profile page and update their information. |
| CT\_9 | Apply | Users are able to apply to jobs. |
| CT\_10 | Resume Upload | Users are able to upload their resume. |
| CT\_11 | Admin - Login | Admin is able to login. |
| CT\_12 | Admin – Search users | Admin is able to search users by their first and last name. |
| CT\_13 | Admin – Deactivate user | Admin is able to deactivate user. |
| CT\_14 | Admin – Delete user | Admin is able to delete user. |

## 

## 4.3 Test Approach

The User Interface will be one of the first features to be completed on the project and therefore will be among the first set of tests executed. The test cases for compatibility will be run manually through user interaction.

As outlined in the requirements document, the EJCA application must be functional on the Safari web browser on the iPhone 6 and iPad Air with iOS 8.1.3, and on the Google Chrome browser on Samsung Galaxy S5 with Android 5.0 (Lollipop). It should also be accessible on the Internet Explorer, Firefox, and Safari browsers on desktop/laptop environments, given that all browsers have up-to-date JavaScript engines.

We will test on the following hardware devices that are available to us:

|  |  |  |
| --- | --- | --- |
| Device | Operating System | Browsers |
| Laptop / Desktop | * Windows (any) * OS X | * Google Chrome * Mozilla Firefox * Safari * Internet Explorer > 9 |
| Tablets   * iPad Air | * iOS 8.1.3 | * Safari |
| Phone   * iPhone 6 * Samsung Galaxy S5 | * iOS 8.1.3 * Android v5.0 (Lollipop) | * Safari * Google Chrome |

## 

## 4.4 Test Pass / Fail Criteria

Pass when 100% of test cases completed

* Fully working in Chrome and Safari on the mobile devices listed above.
* Fully working in IE, Firefox, and Safari on desktops/laptops.

Fail on any found fault with UI compatibility except for CT\_10 (Resume Upload) which is compatible only on desktop/laptop environment.

## 4.5 Test Entry / Exit Criteria

Entry Criteria:

* Verify that the test devices are ready to use.
* Verify if the proper browsers are installed with up-to-date JavaScript engine and ready to use.

Exit criteria:

* All tests have been run successfully and no high priority bugs are left uncorrected.

## 4.6 Test Deliverables

|  |  |
| --- | --- |
| ID | CT\_1 |
| Item to Test | Search Jobs |
| Pre-Conditions | None |
| Test Steps | Repeat the below process for each supported Device and Browser type     1. Click on SearchOpenPositionsbutton |
| Expected Results | System should display all available jobs to the user. |
| Priority | Medium |
| Pass/Fail | Pass if system redirect to JobSearch page and display jobs to user. |

|  |  |
| --- | --- |
| ID | CT\_2 |
| Item to Test | Register |
| Pre-Conditions | None |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on Users dropdown list on the header. 2. Click on Register link. 3. Fill in First Name, Last Name, Email Address, Password and Confirm Password. 4. Click on Register |
| Expected Results | System should send an email to user containing email verification link and display confirmation page to user. |
| Priority | High |
| Pass/Fail | Pass if system redirect user to Register page. |

|  |  |
| --- | --- |
| ID | CT\_3 |
| Item to Test | Email Verification |
| Pre-Conditions | User has created an account |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on link in the email received. |
| Expected Results | System should display verification confirmation page to user and redirect user to login page within 7 seconds. |
| Priority | High |
| Pass/Fail | Pass if system displays email verification confirmation message to user and then redirect user to login page. |

|  |  |
| --- | --- |
| ID | CT\_4 |
| Item to Test | Login |
| Pre-Conditions | User has verified his email address |
| Test Steps | Repeat the below process for each supported device and browser   1. Click on Users dropdown list in the header. 2. Click on Sign In link. 3. Fill in Email address and password 4. Click on Login button. |
| Expected Results | System should redirect user to User dashboard |
| Priority | High |
| Pass/Fail | Pass if system redirects user to user dashboard page. |

|  |  |
| --- | --- |
| ID | CT\_5 |
| Item to Test | Forgot Password |
| Pre-Conditions | User has created an account before. |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on Users dropdown list in the header. 2. Click on Sign In link. 3. Click on Click here if you forgot your password link. 4. Enter email address in the textbox on the Forgot Password page 5. Click on Submit button |
| Expected Results | System should send an email to user containing change password link and display confirmation page. |
| Priority | High |
| Pass/Fail | Pass if user receives an email containing a link to change the password. |

|  |  |
| --- | --- |
| ID | CT\_6 |
| Item to Test | Change has create an account. |
| Pre-Conditions | None |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on link in the forgot password email. 2. Enter password and confirm password. 3. Click on Submitbutton. |
| Expected Results | System should display confirmation message. |
| Priority | High |
| Pass/Fail | Pass if user redirects user to change password page. |

|  |  |
| --- | --- |
| ID | CT\_7 |
| Item to Test | Filter Jobs |
| Pre-Conditions | None |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on link Search Open Positions button 2. Select criteria from City and/or Customer 3. Click on FilterJobs button. |
| Expected Results | System should display jobs matching the criteria. |
| Priority | Medium |
| Pass/Fail | Pass if user displays jobs matching the criteria. |

|  |  |
| --- | --- |
| ID | CT\_8 |
| Item to Test | Profile Information |
| Pre-Conditions | User is logged in |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on link hamburger menu button on the header. 2. Click on Update Profilelink. 3. Change information in any text box.   (First Name, Last Name, Street, City, State, Country are required fields)   1. Click on SaveInformation button. |
| Expected Results | System should display confirmation message. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | CT\_9 |
| Item to Test | Apply |
| Pre-Conditions | User is logged in |
| Test Steps | Repeat the below process for each supported Device and Browser type   1. Click on Details button (either in JobSearch page, Featured Jobs or Recommended Jobs) 2. Click on Applybutton. 3. Verify Information. 4. Click on Submitbutton. |
| Expected Results | System should display confirmation page. |
| Priority | High |
| Pass/Fail | Pass if system redirects user to confirmation page and then redirect user to user dashboard. |

|  |  |
| --- | --- |
| ID | CT\_10 |
| Item to Test | Resume Upload |
| Pre-Conditions | User is logged in |
| Test Steps | 1. Click on link hamburger menu button on the header. 2. Click on Update Profile link. 3. Click on Browse File button. 4. Select any file with pdf, doc or txt extension. 5. Click on Upload Resume button. |
| Expected Results | System should display confirmation message. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message to user. |

|  |  |
| --- | --- |
| ID | CT\_11 |
| Item to Test | Admin – Login |
| Pre-Conditions | Admin email exists in database. |
| Test Steps | 1. Click on Admin link in the homepage. 2. Fill in email id and password. 3. Click on Login. |
| Expected Results | System should display admin dashboard. |
| Priority | High |
| Pass/Fail | Pass if system redirects admin to admin dashboard page. |

|  |  |
| --- | --- |
| ID | CT\_12 |
| Item to Test | Admin – Search users |
| Pre-Conditions | Admin is logged in and in admin dashboard page. |
| Test Steps | 1. Enter first / last name. 2. Click on Search. |
| Expected Results | System should users matching the criteria. |
| Priority | Medium |
| Pass/Fail | Pass if system displays users matching the criteria. |

|  |  |
| --- | --- |
| ID | CT\_13 |
| Item to Test | Admin – Deactivate user |
| Pre-Conditions | Admin is logged in and in admin dashboard page. |
| Test Steps | 1. Click on Deactivatelink button next of any user |
| Expected Results | System should display confirmation message. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

|  |  |
| --- | --- |
| ID | CT\_14 |
| Item to Test | Admin – Delete user |
| Pre-Conditions | Admin is logged in and in admin dashboard page. |
| Test Steps | 1. Click on Deletelink button of any user |
| Expected Results | System should display confirmation message. |
| Priority | Medium |
| Pass/Fail | Pass if system displays confirmation message. |

## 4.7 Test Suspension / Resumption Criteria

Testing will be suspended as soon as 2 bugs are found and find solution to fix the bug. Once implemented, testing will resume from the beginning.

## 4.8 Test Environmental Needs

Testing will be completed on actual hardware devices as well as emulator on Chrome browser. Actual hardware devices are desktop and laptop. IPad Air, iPhone 6 and Samsung Galaxy S5 will be emulated on Chrome browser.

# 5. System Testing

## 5.1 Test Risks / Issues

During system testing it’s important that the system as a whole doesn’t have any issues while all the functions are being carried out.

## 5.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| ST\_1 | Register | Register a new user and verify email. |
| ST\_2 | Login and Apply | Login, and then apply for a job. |
| ST\_3 | Login As Admin | Login with admin credentials. |
| ST\_4 | Deactivate and Delete users. | Deactivate a user and Delete another user. |

## 5.3 Test Approach

System Testing will focus on the functionality of the system as a whole. No unit tests will be used. The system will be tested by interacting with UI.

## 5.4 Test Pass / Fail Criteria

If observed results are the same as expected results and the System is still stable and functional, the test case will pass; otherwise, test case will fail.

## 5.5 Test Entry / Exit Criteria

Entry criteria:

* Prior to testing, web application must be running on one of supported devices mentioned in Compatibility testing.

Exit criteria:

* All the tests have been passed. System remains functional throughout the testing and after.

## 5.6 Test Suspension / Resumption Criteria

Testing will be suspended once an issue is discovered. Issue will be documented and then fixed. Testing will resume from the beginning once the fix has been implemented.

## 5.7 Test Deliverables

|  |  |
| --- | --- |
| ID | ST\_1 |
| Item to Test | Register |
| Pre-Conditions | Web Application is running on one of the compatible devices. |
| Test Steps | * From home screen click on Register. * Fill out the Registration Form with valid data and click submit. * Check the email entered at registration for a verification email. * Click on the verification link in the email. |
| Expected Results | * Registration form is displayed. * After clicking submit, Message is displayed saying verification link has been sent to the email. * Clicking verification link takes user to the web application and displays a message saying verification was successful and the user can login. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_2 |
| Item to Test | Login and Apply |
| Pre-Conditions | Web Application is running on one of the compatible devices. User has been registered and verified. |
| Test Steps | * Click on Login from homepage. * Enter the credentials of a registered and verified user. * Click on a Job. * Click Apply * Attach resume and click submit. |
| Expected Results | * Login form with username and password is displayed after click on login. * user dashboard is displayed with featured jobs. * Clicking on a job shows the job details. * User is able to browse for resume file and attach it. * Submitting application shows a message saying application was successful. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_3 |
| Item to Test | Login as Admin |
| Pre-Conditions | Web Application is running on one of the compatible devices. Admin credentials have been added to the Database. |
| Test Steps | * Click on Login from homepage. * Enter the credentials of the admin. * Click submit |
| Expected Results | * Login form with username and password is displayed after click on login. * Admin dashboard is displayed. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_4 |
| Item to Test | Deactivate and Delete users. |
| Pre-Conditions | Web Application is running on one of the compatible devices. Admin credentials have been added to the Database.  Admin has been logged in(ST\_3) |
| Test Steps | * Click on deactivate for a user. * Click on delete for another user. |
| Expected Results | * Message is displayed saying the user has been deactivated. * Message is displayed saying the user has been deleted. |
| Priority | High |
| Pass/Fail |  |

## 5.8 Test Environmental Needs

Testing must be carried out in each of devices and browsers mentioned in the compatibility requirements section above to ensure that environment differences do not affect the System.

# 6. Integration Testing

## 6.1 Test Risks / Issues

Given the compactness of the ASP.NET MVC framework, the project does not involve any large disjointed architectural components that are integrated together. Therefore, integration testing is only pursued insofar as it is necessary to confirm that the model, view, and controller components work correctly together to perform the functionality of the web applications.

## 6.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| IT\_1 | Home Page | Home page url request goes to the home controller and home page is viewed. |
| IT\_2 | Register | Action Register in User Controller invoked and users are able to create an account. |
| IT\_3 | Email verification | Action Register in User Controller invoked and users are able to open the link sent in the verification email. |
| IT\_4 | Login | Login Action in User Controller invoked and users are able to login. |
| IT\_5 | Forgot Password | ForgotPassword Action in User Controller invoked and users are able to request link and change their password. |
| IT\_6 | Change Password | ChangePassword Action in User Controller invoked and users are able to change their password after logging in. |
| IT\_7 | Profile Information | Profile Action in User Controller invoked and Users are able to access their profile page and update their information. |
| IT\_8 | Resume Upload | UploadResume Action in User Controller invoked and Users are able to upload their resume. |

## 6.3 Test Approach

Integration testing tests the entire software stack working together. Given that functional and nonfunctional testing has already been performed, integration testing involves completing typical use case routines on the user interface of the application. These routines are listed in the box above. Should an error occur, a possible solution may lie in the system integration, although, due to the reasons mentioned above, this is unlikely for our system; thus, a more forthcoming solution may be to perform unit testing instead.

## 6.4 Test Pass / Fail Criteria

Pass when 100% of test cases completed, fail if an unhandled exception is thrown or other error occurs, preventing full application functionality.

## 6.5 Test Entry / Exit Criteria

Entry Criteria:

* Functional and nonfunctional testing has been completed to verify application functions as expected and integration errors, if any, may be isolated

Exit criteria:

* All tests have been run successfully and no high priority bugs are left uncorrected

## 6.6 Test Suspension / Resumption Criteria

Testing will be suspended as soon as a bug is found and find solution to fix the bug. Once implemented, testing will resume from the beginning.

## 6.7 Test Environmental Needs

In order to view different architectural components of the software project should the need arise, a desktop or laptop environment with Microsoft Visual Studio Ultimate is required to complete integration testing, along with the project source code.

# 7. Performance and Load Testing

## 7.1 Test Risks / Issues

We will be testing performance using a Visual Studio-generated load stress test, with the risk that this test will not adequately reflect the performance of the application in the event of simultaneous use of the site more than fifty users.

## 7.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| PT\_1 | 50 simultaneous users | We will simulate 50 simultaneous login requests to the server looking for an average response time of 2 seconds or less |

## 7.3 Test Approach

We will be conducting a very standard automated load test using the Microsoft Visual Studio tool designed for use with ASP.NET web applications. On average, it is anticipated that only a few users will be accessing the EJCA at the same time, however to be safe we are testing using fifty simulated logins, as this involves not only requests to the webserver but also to the database.

## 7.4 Test Pass / Fail Criteria

The application will pass performance and load testing when fifty users can simultaneously log into the application with a two second average load time.

## 7.5 Test Entry / Exit Criteria

The criteria necessary for entry of performance unit tests are:

* All previous functional and nonfunctional unit test cases have been run and passed
* A stress test plan has been configured in the Visual Studio Ultimate testing suite

The exit criteria for performance and load testing are passing of the Visual Studio and self-timed tests.

## 7.6 Test Deliverables

|  |  |
| --- | --- |
| ID | PT\_1 |
| Item to Test | User Login |
| Pre-Conditions | A Visual Studio stress test has been written that does the following:   * Simulate 50 browsing sessions * Request the root URL * Request the user login screen * Enter 50 valid login usernames and passwords |
| Test Steps | 1. Create a new load test using the Load Test Wizard. 2. In the wizard, specify the test type and project to test. 3. Page through the wizard and fill in necessary test details. 4. Execute the Visual Studio stress test. |
| Expected Results | Two second average load time for all simulated users, as specified by the auto generated test report. |
| Priority | High |
| Pass/Fail | Result log from the Visual Studio stress test |

## 7.7 Test Suspension / Resumption Criteria

Given that we have only one performance or load test, our goal is that our application will pass the test. However, if we encounter a failure, we will suspend testing, investigate and resolve the cause of failure, and retest until we attain a successful outcome.

## 7.8 Test Environmental Needs

Performance testing requires access to a machine running Visual Studio Ultimate in order to make use of the built-in stress testing software tool for ASP.NET web applications. Use of the Visual Studio tool also requires the EJCA project source code.

# 8. Security Testing

## 8.1 Test Risks / Issues

The only risk is that the security tests do not cover all security loopholes in the web application. Thus, security testing must be performed thoroughly on each security feature of the web application so that users feel safe accessing the portal and submitting their personal data. Security breaches could result in loss or user data, eroding user confidence in the application.

## 8.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| ST\_1 (\_) | Secure User Login | Only users who have registered and created an account may access the portal and apply for jobs |
| ST\_2 (\_) | Admin Mode | Only a single administrator may access the admin portal and manipulate user account information |
| ST\_3 (\_) | Password Encryption | User passwords are stored encrypted in the database and are unreadable to anyone accessing the database |
| ST\_4 (\_) | SSL Encryption | Data transmitted on the internet between the user’s device and the webserver is encrypted and uses HTTPS protocol |

## 8.3 Test Approach

Security tests will be conducted using the web application’s user interface, by accessing Geocerts.com SSL certificate checking site, and by reviewing records in the Users table of the application database.

## 8.4 Test Pass / Fail Criteria

The web application passes security testing when each of the test cases passes. If any of the tests fail, the issue will be debugged and resolved before the application is retested.

## 8.5 Test Entry / Exit Criteria

Security testing will begin after the functional and nonfunctional unit testing of the application is finished and will be complete when all possible security vulnerabilities have been detected and resolved.

## 8.6 Test Deliverables

|  |  |
| --- | --- |
| ID | ST\_1.1 |
| Item to Test | Secure User Login |
| Pre-Conditions | All other functional and nonfunctional application testing is complete |
| Test Steps | 1. Load the web application home page. 2. Click on the Users Icon dropdown 3. Click on Login. On the login screen that appears:    1. Enter “test123@test123.com” for username/email    2. Enter “test123” for password 4. Click submit |
| Expected Results | A message must be shown that invalid user login credentials were entered. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_1.2 |
| Item to Test | Secure User Login |
| Pre-Conditions | All other functional and nonfunctional application testing is complete |
| Test Steps | 1. Load the web application home page. 2. Click on the Users Icon dropdown 3. Click on Login. On the login screen that appears:    1. Enter “wesley.trescott@gmail.com” for username/email    2. Enter “westres2993” for password 4. Click submit |
| Expected Results | The user dashboard screen will load. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_2.1 |
| Item to Test | Admin Mode |
| Pre-Conditions | All other functional and nonfunctional application testing is complete |
| Test Steps | 1. Load the web application home page. 2. Click on the Users Icon dropdown 3. Click on Login. On the login screen that appears:    1. Enter valid Admin credentials 4. Click submit |
| Expected Results | The admin dashboard will load, providing the option to deactivate or ban users |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_2.2 |
| Item to Test | Admin Mode |
| Pre-Conditions | Admin user is logged in |
| Test Steps | 1. From the admin dashboard screen, select Deactivate User for a particular user entry 2. Load the ASP.NET MVC project solution in Visual Studio. 3. Open the application database .mdf file. Right click on Tbl\_Users and select “View Table Data.” 4. That the flag is set in the relevant column for the deactivated user. |
| Expected Results | The deactivated user can no longer log into the web application |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_3 |
| Item to Test | Password Encryption |
| Pre-Conditions | All other functional and nonfunctional application testing is complete |
| Test Steps | 1. Load the ASP.NET MVC project solution in Visual Studio. 2. Open the application database .mdf file. Right click on Tbl\_Users and select “View Table Data.” 3. Verify that no user password is a human readable string (i.e. that each password is a long hash code string). |
| Expected Results | Each user password is encrypted in a string that is unreadable |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | ST\_4 |
| Item to Test | SSL Encryption |
| Pre-Conditions | All other functional and nonfunctional application testing is complete |
| Test Steps | 1. Navigate to https://www.geocerts.com/ssl\_checker. 2. In the domain name box, enter the following IP address: http://115.113.203.117/. 3. Click the “Check SSL” button. 4. Review server and SSL certificate information, and verify that SSL certificate is still valid. |
| Expected Results | A message must be shown displaying the correct server information and verifying that the server has an up-to-date SSL certificate. |
| Priority | High |
| Pass/Fail |  |

## 8.7 Test Suspension / Resumption Criteria

Given the great importance of security to the successful functioning of the web application, and the interconnectedness of the security architecture, security testing will be suspended when a single vulnerability is detected. After the vulnerability is resolved, testing must be restarted to ensure that all security features continue their proper functionality.

## 8.8 Test Environmental Needs

To complete security testing, the test administrator needs access to the application database to verify user password encryption, as well as access to the Geocerts website to verify SSL certificates.

# 9. User Acceptance Testing

## 9.1 Test Risks / Issues

User Acceptance testing (UAT) is the last step in testing and will be done by the members of Computech quality assurance team.

## 9.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| UAT\_1 | Register | A new user is able to register. |
| UAT\_2 | Email verification | User is able to verify email after registration. |
| UAT\_3 | Login | A new user is able to login after registration. |
| UAT\_4 | Forgot Password | User is able to get link to reset password in email. |
| UAT\_5 | Change Password | A new user is able to change password from the link sent to his email. |
| UAT\_6 | Apply | Once logged in, a user is able to apply for jobs. |
| UAT\_7 | Job Details | A new user is able to view job details. |
| UAT\_8 | Filter Jobs | A new user is able to filter jobs using all the available filters. |

## 9.3 Test Approach

User Acceptance Testing is the last step after all other tests have been passed. UAT will be done by the members of Computech quality assurance (QA) team after the Web Application has been hosted on their server. After Computech QA signs off on UAT, there should be no more bugs that need to be fixed.

## 9.4 Test Pass / Fail Criteria

Pass when 100% of test cases completed successfully and Computech QA team signs off on the Application.

## 9.5 Test Entry / Exit Criteria

Entry Criteria:

Exit criteria:

* All tests have been run successfully and no high priority bugs are left uncorrected.

## 9.6 Test Deliverables

|  |  |
| --- | --- |
| ID | UAT\_1.1 |
| Item to Test | Register |
| Pre-Conditions | Web Application is hosted on Computech server. |
| Test Steps | Click on Register. Fill out the form and click submit. |
| Expected Results | A message must be shown that verification link has been sent to email. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_1.2 |
| Item to Test | RegisterFails |
| Pre-Conditions | Account already created with an email. |
| Test Steps | UAT\_1.1 repeated with the already registered email. |
| Expected Results | Message must be shown indicating Registration failed because an account has already been created with the entered email. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_2.1 |
| Item to Test | Email verification |
| Pre-Conditions | Execute UAT\_1.1 |
| Test Steps | Click on the verification link received in email. |
| Expected Results | Email should be verified with message "Thank you for verifying your email. You can now log in to the account, set up your profile and apply to jobs. The distance between you and your career has never been closer." |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_3.1 |
| Item to Test | Login |
| Pre-Conditions | User has been registered and Email has been verified |
| Test Steps | Click Login. Enter credentials used to register and verify. |
| Expected Results | Login should be successful and user redirected to User Dashboard. |
| Priority | True should be returned. |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_3.2 |
| Item to Test | LoginFails |
| Pre-Conditions | Username and Password have not been used to register. |
| Test Steps | Login with the Username and Password. |
| Expected Results | Error message displayed indicating Login failed due to invalid credentials. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_4.1 |
| Item to Test | Forgot Password |
| Pre-Conditions | User account has been created |
| Test Steps | Click on forgot password. |
| Expected Results | System sends an email and message “We sent an email with link to change the password. Please check your email.” is displayed. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_5.1 |
| Item to Test | Change Password |
| Pre-Conditions | User account has been created. UAT\_4.1 executed. |
| Test Steps | Click on change password link sent to email. |
| Expected Results | User password is successfully changed. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_6.1 |
| Item to Test | Apply |
| Pre-Conditions | User has logged in. Click on a job to go to job details. |
| Test Steps | Click on Apply. |
| Expected Results | Takes user to application page. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_7 |
| Item to Test | Job Details |
| Pre-Conditions | None |
| Test Steps | Click on a job on the dashboard. |
| Expected Results | Job details are displayed. |
| Priority | High |
| Pass/Fail |  |

|  |  |
| --- | --- |
| ID | UAT\_8 |
| Item to Test | Filter |
| Pre-Conditions | None |
| Test Steps | Filter by all the filter (One at a time and multiple multiple filters together). |
| Expected Results | Correct filtered jobs are displayed. |
| Priority | High |
| Pass/Fail |  |

## 9.7 Test Suspension / Resumption Criteria

Testing will be not suspended when a bug is found. Bugs will be reported to the development team for fixes. Once fixes are implemented, an updated version of the application will be provided to the QA team to continue UAT with the updated version. UAT will not be suspended until all issues are resolved and Computech QA team signs off on the application.

## 9.8 Test Environmental Needs

Application will be hosted on the Computech server and be accessible to anyone via internet on a supported device and web browser.

## 10. Unit Testing

### 10.1 Test Risks / Issues

Unit test code needs to thoroughly test all application functionality and have at least 75% code coverage, as inadequate tests will not detect actual bugs in the code and might display unreliable output.

### 10.2 Items To Be Tested

|  |  |  |
| --- | --- | --- |
| **ID** | **Item to Test** | **Test Description** |
| UT\_1 (\_) | Register | Call User.Register() to create user account |
| UT\_2 (\_) | Email verification | Call User.Confirmed() to verify user email. |
| UT\_3 (\_) | Login | Call LoginUser.IsValid() to verify credentials. |
| UT\_4 (\_) | Forgot Password | Call EmailManager.SendForgotPasswordEmail() to send an email to user. |
| UT\_5 (\_) | Change Password | Call ChangePassword.UpdatePassword() to update password in database. |
| UT\_6 (\_) | Apply | Call Apply.SubmitApplication() to apply to a job. |
| UT\_7 (\_) | Job Details | Call Details() in JobSearchController to display details of a job. |
| UT\_8 (\_) | Filter Jobs | Call Index() in JobSearchController to filter jobs based on parameters. |
| UT\_9 (\_) | Admin – Login | Call IsAdminValid() to verify admin credentials. |

### 10.3 Test Approach

Unit test cases will be written to test each method in all of the model classes. The test cases will include both valid and invalid input to test classes such as User, LoginUser, UserProfile, ChangePassword, ForgotPassword and Apply. The JobSearch controller will also be tested to verify whether the jobs are displayed and filtered based on criteria.

### 10.4 Test Pass / Fail Criteria

Use Assert classes to verify the returned Boolean value / View with expected Boolean value/View. Pass if assert passes; otherwise fail.

### 10.5 Test Entry / Exit Criteria

Entry Criteria:

* Test environment is ready to use.
* Adequate test data is available.
* Test stubs are completed.

Exit criteria:

* All modules in class are tested.
* All test cases pass successfully.
* Defects found are documented to find alternate solution.

### 10.6 Test Deliverables

|  |  |
| --- | --- |
| ID | UT\_1.1 |
| Item to Test | Register |
| Pre-Conditions | None |
| Test Steps | 1. Call User.Register(firstName, lastName, email, password, guid)   {  firstName = “John”,  lastName = “Doe”,  email = “john.doe@email.com”,  password = “123456”,  GUID = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned |
| Priority | High |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_1.2 |
| Item to Test | Register Fails |
| Pre-Conditions | None |
| Test Steps | 1. Call User.Register(firstName, lastName, email, password, guid)   {  firstName = “”,  lastName = “”,  email = “”,  password = “”,  GUID = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_1.3 |
| Item to Test | Register Fails |
| Pre-Conditions | Account already created with email “john.doe@email.com” |
| Test Steps | 1. Call User.Register(firstName, lastName, email, password, guid)   {  firstName = “John”,  lastName = “Doe”,  email = john.doe@email.com,  password = “123456”,  GUID = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_2.1 |
| Item to Test | Email verification |
| Pre-Conditions | Account already created with email “john.doe@email.com” |
| Test Steps | 1. Call User.Confirmed(email)   {  email = “john.doe@email.com”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High. |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_2.2 |
| Item to Test | Email Verification Fails |
| Pre-Conditions | Account already created with email “john.doe@email.com” |
| Test Steps | 1. Call User.Confirmed(email)   {  email = “”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_3.1 |
| Item to Test | Login |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Call LoginUser.IsValid (email, password)   {  email = “john.doe@email.com”,  password = “123456”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High. |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_3.2 |
| Item to Test | Login Fails |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Call LoginUser.IsValid (email, password)   {  email = “”,  password = “”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_3.3 |
| Item to Test | Login Fails |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Call LoginUser.IsValid (email, password)   {  email = “”,  password = “123456”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_3.4 |
| Item to Test | Fails |
| Pre-Conditions | Email has been verified |
| Test Steps | 1. Call LoginUser.IsValid (email, password)   {  email = “john.doe@email.com”,  password = “123456abc”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_4.1 |
| Item to Test | Forgot Password |
| Pre-Conditions | User account has been created |
| Test Steps | 1. Call EmailManager.SendForgotPasswordEmail(firstName, email, guid)   {  firstName = “John”,  email = “john.doe@email.com”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_5.1 |
| Item to Test | Change Password |
| Pre-Conditions | User account has been created |
| Test Steps | 1. Call User.UpdatePassword(email, currentPassword, password, guid){   email = “john.doe@email.com”,  currentPassword = “”,  password = “123abc”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_5.2 |
| Item to Test | Change Password Fails |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call User.UpdatePassword(email, currentPassword, password, guid){   email = “john.doe@email.com”,  currentPassword = “123456”,  password = “123abc”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_5.3 |
| Item to Test | Change Password Fails |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call User.UpdatePassword(email, currentPassword, password, guid){   email = “john.doe@email.com”,  currentPassword = “111111”,  password = “123abc”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_5.4 |
| Item to Test | Change Password Fails |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call User.UpdatePassword(email, currentPassword, password, guid){   email = “john.doe@email.com”,  currentPassword = “”,  password = “”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_5.5 |
| Item to Test | Change Password Fails |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call User.UpdatePassword(email, currentPassword, password, guid){   email = “”,  currentPassword = “”,  password = “123abc”,  guid = “a6d7f3b6-c65d-4319-89d9-e5b518720e39”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_6.1 |
| Item to Test | Apply |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call Apply.submitApplication(email, jobId){   email = “john.doe@email.com”,  jobId = “17604”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_6.2 |
| Item to Test | ApplyFails |
| Pre-Conditions | User has logged in |
| Test Steps | 1. Call Apply.submitApplication(email, jobId){   email = “”,  jobId = “17604”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_7 |
| Item to Test | Job Details |
| Pre-Conditions | None |
| Test Steps | 1. Pass id in JobSearchController.Details(17604) 2. Store returned view in a variable. 3. Verify returned view with expected view. |
| Expected Results | Redirected View is “Details”. |
| Priority | Medium |
| Pass/Fail | Assert.AreEqual(returnedView,expectedView) should be true |

|  |  |
| --- | --- |
| ID | UT\_8 |
| Item to Test | Filter |
| Pre-Conditions | None |
| Test Steps | 1. Pass parameters in JobSearchController.Index(“Detroit”,”Computech”) 2. Store returned view in a variable. 3. Verify returned view with expected view. |
| Expected Results | Redirected View is “Index” |
| Priority | Medium |
| Pass/Fail | Assert.AreEqual(returnedView,expectedView) should be true |

|  |  |
| --- | --- |
| ID | UT\_9.1 |
| Item to Test | Admin – login |
| Pre-Conditions | Email exists in database. |
| Test Steps | 1. Call LoginUser.IsAdminValid (email, password)   {  email = “admin@email.com”,  password = “123456”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | True should be returned. |
| Priority | High. |
| Pass/Fail | Pass if IsAssert(true) is true |

|  |  |
| --- | --- |
| ID | UT\_9.2 |
| Item to Test | Admin LoginFails |
| Pre-Conditions | Email exists in database |
| Test Steps | 1. Call LoginUser.IsAdminValid (email, password)   {  email = “”,  password = “”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_9.3 |
| Item to Test | Admin LoginFails |
| Pre-Conditions | Email exists in database |
| Test Steps | 1. Call LoginUser.IsAdminValid (email, password)   {  email = “”,  password = “123456”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

|  |  |
| --- | --- |
| ID | UT\_9.4 |
| Item to Test | Admin LoginFails |
| Pre-Conditions | Email exists in database |
| Test Steps | 1. Call LoginUser.IsAdminValid (email, password)   {  email = “admin@email.com”,  password = “123456abc”  } and store returned value in Boolean variable.   1. Verify returned value with expected value. |
| Expected Results | False should be returned. |
| Priority | High |
| Pass/Fail | Pass if IsAssert(false) is true |

### 10.7 Test Suspension / Resumption Criteria

Testing will be suspended as soon as a bug is found, and the test administrator will then find solution to the bug. Once the solution is implemented, testing will resume from the beginning of that unit test.

### 10.8 Test Environmental Needs

Unit testing will be carried out using Microsoft’s NUnit testing environment in Visual Studio.