

Testing Report

Applicant Tracker

Client

Mosaic Learning - Carol Curley

Team 5 “Agile, Handle with Care”

Arvin Siva

Brian Wilson

Vihar Patel

James Baker

Steven Hargrove

Logan Rites

11/29/17

Table of Contents

[1. Introduction](#)

[1.1 Purpose of This Document](#)

[1.2 References](#)

[2. Testing Process](#)

[2.1 Description](#)

[2.2 Testing Sessions](#)

[2.3 Impressions of the Process](#)

[3. Test Results](#)

[3.1 Testing Suite](#)

[3.1.1 Adding an applicant](#)

[3.1.5 Adding User](#)

[3.1.6 Editing a User](#)

[3.1.7 Removing a database user](#)

[3.1.8 Editing interview Questions](#)

[3.1.9 Viewing Interview Form](#)

[3.1.10 Filtering applicants](#)

[3.1.11 Add to Question List](#)

[3.2 Test Results](#)

[3.2.1 Table of Tests Done](#)

[3.2.2 Testing Screenshots.](#)

[Appendix A – Team Review Sign-off](#)

[Appendix B – Document Contributions](#)

1. Introduction

1.1 Purpose of This Document

This report was made to document and convey the various testing procedures our development group went through to make sure our system has minimal errors/bugs in it. We will go over different capabilities that were tested and how they were done, going over good and bad inputs. We will also convey when these tests were done and by who.

1.2 References

Use cases from the System Requirements Specification document

2. Testing Process

2.1 Description

Our testing is based on whether we satisfied our requirements laid out by the use cases in our SRS document. Some testing was done immediately after these requirements were implemented, but more formal tests were performed later on.

The testing described in this document will mostly include our formal testing sessions after the implementation of our system. If there were no bugs along the way, we refer to our initial testing of the requirements immediately after these functions were implemented.

2.2 Testing Sessions

Date	Location	Start	End	Performed By	Use Cases
11/27/17 M	School	4:00 PM	4:45 PM	Brian Wilson	Writing Assessment Notes
11/20/17 M	School	4:30 PM	5:30 PM	James Baker	Add to/Edit Question List
11/17/17 F	School	2:30 PM	4:00 PM	Vihar Patel	Filter Search
11/15/17 W	School	4:00 PM	5:00 PM	Steven Hargrove	Add/Edit Job Role
11/13/17 M	School	4:00 PM	5:30 PM	Logan Rites	Add/Edit User
11/10/17 F	School	2:30 PM	4:15 PM	Arvin Siva	Test Login/Change Password

2.3 Impressions of the Process

The use cases laid out in the SRS conveyed the prime functionality of our system, so by using the system in ordinary and not-so-ordinary ways, we tested whether the system could perform its job correctly, as well as handle unexpected input properly. We based all functionality on the use cases laid out in the SRS and by testing the use cases, we are testing that the system does what our client wanted it to do. We feel that our testing process was quick and simple, at least more so than the implementation itself.

Going over drop-down and written input was the main component of our testing. Being able to add jobs, applicants, and users was mostly what the testing was for. Making sure the database behaved as expected was the main objective of our tests. We also wanted to make sure all input information could be edited or removed without any bugs arising. This part was relatively simple compared to other tests.

The admin control panel was the final frontier of our development. Adding jobs added complexity to this project, since many of our functions depended on what job the applicant was being hired for. Working out these dependencies meant a lot of testing had to be done for it. Something like filtering through the applicants proved to be a lot simpler of a test.

3. Test Results

Since our product is more centered on qualitative data storage, boundary testing ended up being a rather small part of our testing process. Instead, our testing relied more heavily on testing what was valid and invalid input. Also, our tests mainly focused on how the database stored good and bad information, and whether or not bad input was handled properly. Also we tested the system's organizational skills because that was what our product was mainly designed for. Instead of making boundaries, we focused a lot on usability testing, while also making sure all types of input were handled properly. If the database only allowed valid information to be stored and allowed users to edit it effectively, then our testing resulted in a success.

3.1 Testing Suite

3.1.1 Adding an applicant

Use Case	Add Applicant
Purpose	User adds an applicant to the database
Valid situation	User adds applicant to the system
Invalid situation A	User attempts to submit invalid information in a form
Expected results for valid situation	User can see the applicant is now in the table of applicants
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	Entering numbers in the name field, letters in salary, or having input exceed max allowed characters

3.1.2 Editing an applicant

Use Case	Edit Applicant
Purpose	User edits information of an applicant in the database
Valid situation	User edits applicant information with valid information
Invalid situation	User edits applicant information with invalid information
Expected results for valid situation	User can see that the applicant information is updated properly
Expected results for invalid situation	An error message describing which input is invalid/missing.
Boundary Conditions	Entering numbers in the name field, letters in the salary field, or having input exceed max allowed

3.1.3 Removing an applicant

Use Case	Removing an Applicant
Purpose	Test login ability and access privileges
Valid situation	A user has ID and uses correct password
Invalid situation A	A user ID does not exist
Invalid situation B	A user supplies wrong password
Expected results for valid situation	User can login, and view appropriate information given their login level
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	Invalid login credentials result in an error being displayed.

3.1.4 Logging In

Use Case	User Login
Purpose	Test login ability and access privileges
Valid situation	A user has ID and uses correct password
Invalid situation A	A user ID does not exist
Invalid situation B	A user supplies wrong password
Expected results for valid situation	User can login, and view appropriate information given their login level
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	Invalid login credentials result in an error being displayed.

3.1.5 Adding User

Use Case	Adding user to access list
Purpose	Test ability to create new database users
Valid situation	Admin enters proper input onto page
Invalid situation	A field is left empty, or dropdown list left to default.
Expected results for valid situation	Question successfully added and displayed for appropriate job type and question type
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	none.

3.1.6 Editing a User

Use Case	Editing a database user
Purpose	Test ability to create modify database users
Valid situation	Admin enters proper input onto page
Invalid situation	A field is removed and left empty
Expected results for valid situation	The selected user has the pertinent info changed upon submission.
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	none.

3.1.7 Removing a database user

Use Case	Editing a database user
Purpose	Test ability to create modify database users
Valid situation	Admin enters proper input onto page
Invalid situation	none
Expected results for valid situation	The user no longer shows up in table of users
Expected results for invalid situation	none
Boundary Conditions	none.

3.1.8 Editing interview Questions

Use Case	Editing a database user
Purpose	Test ability to create modify database users
Valid situation	Admin enters proper input onto page
Invalid situation	none
Expected results for valid situation	The user no longer shows up in table of users
Expected results for invalid situation	none
Boundary Conditions	none.

3.1.9 Viewing Interview Form

Use Case	View Interview Form
Purpose	Display interview questions that are to be asked to applicant
Valid situation	Show questions in question database for particular job role or nothing if there is nothing in the question database
Invalid situation	None
Expected results for valid situation	Display interview questions
Expected results for invalid situation	None
Boundary Conditions	No questions or an absurd amount of questions

3.1.10 Filtering applicants

Use Case	Filtering Applicants
Purpose	Test searching & filtering ability of main table
Valid Situation	Search query is submitted
Invalid situation	None
Expected results for valid situation	If applicants match search query, they are shown in table, else if no matching applicants the table is shown empty.
Expected results for invalid situation	None
Boundary Conditions	none

3.1.11 Add to Question List

Use Case	Add to Question List
Purpose	Test ability to edit list of questions
Valid situation	A new question is added
Invalid situation	A field is not entered correctly
Expected results for valid situation	Question successfully added and displayed for appropriate job type and question type
Expected results for invalid situation	An error message describing the error is displayed.
Boundary Conditions	none.

3.2 Test Results

3.2.1 Table of Tests Done

Test Name	Case	Result
Login	Invalid user ID	Error message displayed
Login	Invalid password	Error message displayed
Login	No password entered	Error message displayed
Admin Login	An admin enters correct credentials	Admin controls are displayed upon login
User Login	A non-admin enters correct credentials	Admin privileges are hidden
User change password (admin)	User hits change password option for self (both normal / admin)	Password successfully changed
User change password	Incorrect previous password entered and/or non matching new passwords entered	Corresponding error message(s) are shown. Both if wrong previous password and non-matching
User change password	User enters correct old password, but uses same old password as the “new” one.	WARNING: This works, technically it’s not “wrong” but it’s poor practice. May be too much to implement in one week.
User privilege change	Admin switches roll for user	1.) Works, previous admins can not login when switched to “Dev”, 2.) promotion to admin also works for initial non-admins.
Add Job Role	Admin enters new job	New job pops up in menus
Add Question	Admin enters a new question for job role	New question shows up for both initial job roles, and newly added ones.
Add question	Admin does not enter text (blank entry)	Pass, Error message displayed.
Add question	Admin forgets to select Job type or Question type	Pass, both error messages are displayed.

View Applicant	Check an applicant is viewable by a user and see if proper info shows up	ERROR , salary is visible to all rolls
Edit Applicant	add/edit data in fields	WARNING , Assessment should note who made edits. All other fields work
Deactivate user	Deactivate a user so they can not login anymore	1.)Pass, user can not login again. 2.) WARNING : no way to reactivate a user, or check if ID has been previously created

3.2.2 Testing Screenshots.

Blank fields on adding applicant

The first name field is required.

The last name field is required.

The source field is required.

The location field is required.

The job title field is required.

Add Applicant

First Name:

Last Name:

Source:

Location:

Adding blank job

The old field is required.

The password field is required.

Change Password

Old Password

The old field is required.

Password

The password field is required.

Confirm Password

Submit

Change password error

The old field is required.

The password field is required.

Change Password

Old Password

The old field is required.

Password

The password field is required.

Confirm Password

Submit

Edit applicant blank fields

The first name field is required.

The last name field is required.

The source field is required.

The location field is required.

Edit Applicant

First Name:

Brian

Last Name:

Wilson

Source:

External

Location:

Baltimore, MD

Job:

Invalid login attempts

Login

E-Mail Address

blank

Passw

Please include an '@' in the email address. 'blank' is missing an '@'.

☐ Remember Me

Login

Forgot Your Password?

Attempt 2

These credentials do not match our records.

Login

E-Mail Address

blank@umbc.edu

These credentials do not match our records.

Password

☐ Remember Me

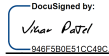
Login

Forgot Your Password?

Appendix A – Team Review Sign-off

All team members have reviewed this document and agree on both the content and the format. Any concerns are addressed in team comments below.

Name: Vihar Patel

DocuSigned by:

049F5B0E51CC48C

Signature

Date: 11/29/2017 | 17:47 PM EST

Comments: _____

Name: Steven Hargrove

DocuSigned by:

070B202EE8D0ACD

Signature

Date: 11/29/2017 | 17:48 PM EST

Comments: _____

Name: Logan Rites

DocuSigned by:

0C6A899C07108E1

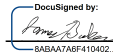
Signature

Date: 11/29/2017 | 17:47 PM EST

Comments: _____

(continued)

Name: James Baker

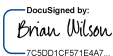
DocuSigned by:

BABAA7ABF410402...

Signature

Date: 11/29/2017 | 17:46 PM EST

Comments: _____

Name: Brian wilson


DocuSigned by:

7C8DD1CF571E4A7...

Signature

Date: 11/29/2017 | 17:29 PM EST

Comments: _____

Name: Arvin Siva

DocuSigned by:

A0211A55DE1C41A...

Signature

Date: 11/29/2017 | 17:49 PM EST

Comments: _____

Appendix B – Document Contributions

This document was worked on by all team members over two group meetings. Brian Wilson, James Baker, and Vihar Patel worked on the charts. Steven Hargrove, Logan Rites, and Arvin Siva worked on the longer paragraph descriptions and added screenshots.