

System Requirements Specification

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

Table of Contents

1. Introduction	<u>Page</u>
1.1 Purpose of This Document	3
1.2 References	3
1.3 Purpose of the Product	3
1.4 Product Scope	4
2. Functional Requirements	
2.1 Use Case 1	5
2.2 Use Case 2	6
2.3 Use Case 3	7
2.4 Use Case 4	7
2.5 Use Case 5	8
2.6 Use Case 6	8
2.7 Use Case 7	9
2.8 Use Case 8	10
2.9 Use Case 9	11
2.10 Use Case 10	11
3. Use Case Tests	
3.1 Use Case 1 Test - Adding Applicant	12
3.2 Use Case 2 Test - Editing Applicant	12
3.3 Use Case 3 Test - Removing Applicant	12
3.4 Use Case 4 Test - Logging in to the Database	12
3.5 Use Case 5 Test - Adding Database User	12
3.6 Use Case 6 Test - Editing Database User	12
3.7 Use Case 7 Test - Removing Database User	13
3.8 Use Case 8 Test - Editing Interview Questions	13
3.9 Use Case 9 Test - Viewing Interview Form	13
3.10 Use Case 10 Test - Filtering Applicants	13
4. Non-Functional Requirements	14
5. User Interface	14
6. Deliverables	15
7. Open Issues	15
Appendix A – Agreement Between Customer and Contractor	16
Appendix B – Team Review Sign-off	18
Appendix C – Document Contributions	20
Appendix D -- Addendum	21
Use cases 11-16	21 - 24

1. Introduction

1.1 Purpose of This Document

This document is designed to explain the features of the applicant tracker application, its functions, and its required conditions to operate. The intended audience is the Mosaic Learning development team as well as the faculty customer, Carol Curley.

1.2 References

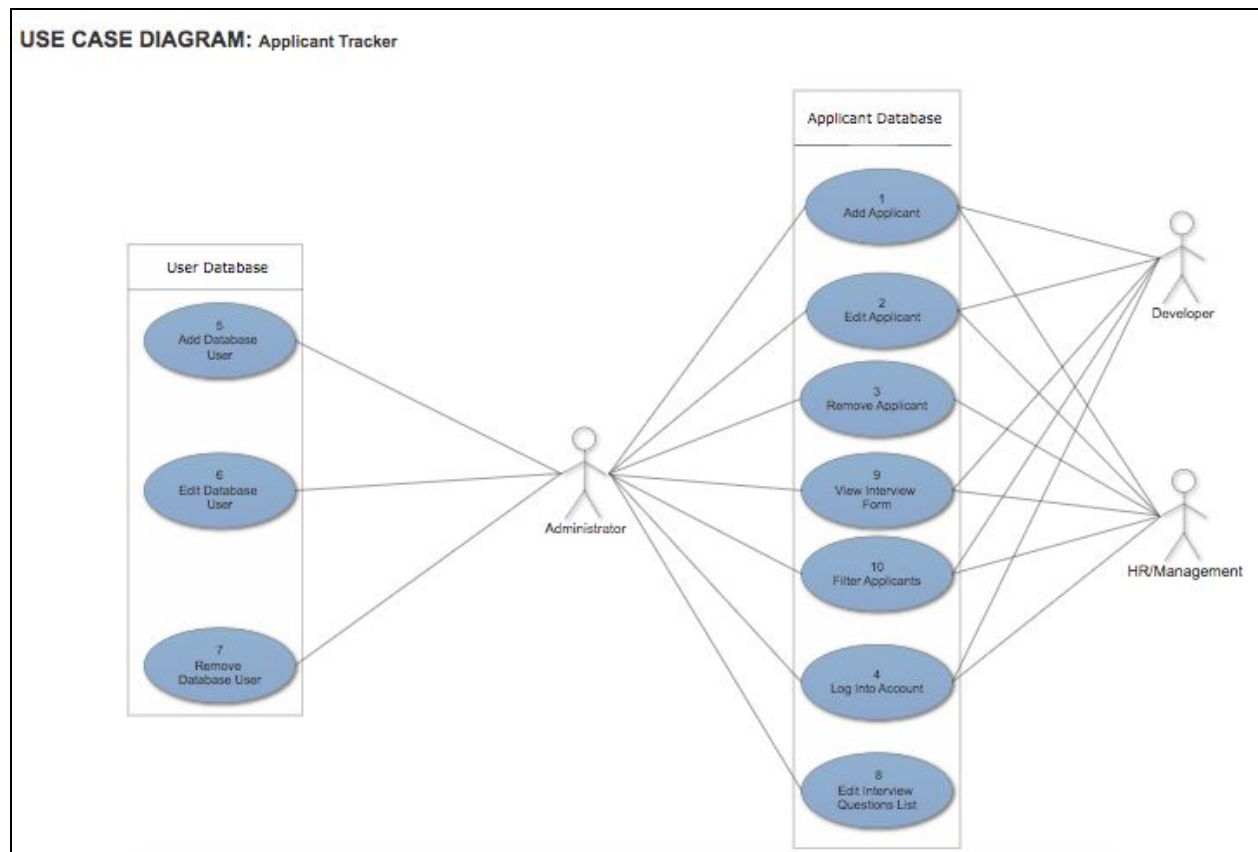
SmartDraw for Use Case Diagram

“Flowchart Maker Floor Plan Maker Org Chart Maker Diagram Maker.” SmartDraw - Create Flowcharts, Floor Plans, and Other Diagrams, www.smartdraw.com

1.3 Purpose of the Product

The applicant tracker application is an organization tool to view, search, and manage current applicants for a company. This application will serve as a database that multiple groups can access and interact with. Developers / Interviewers will be able to add comments to an applicant's record. Managers will be able to comment, add new applicants, and update manager comments & statuses. HR reps will be able to comment, view salary, and update HR comments & statuses. Admins will be able to add/remove new comment fields, add/remove users, as well as view and edit all fields for an applicant.

1.4 Product Scope



The applicant tracker is a web application that consists of use cases consisting of: adding, editing, removing database users and applicants, logging into accounts, filtering applicants and editing interview questions. For further understanding of what use cases pertain to certain primary actors refer to the diagram above.

2. Functional Requirements

See section 1.4 for a brief description of the functional requirements of this system. More detailed descriptions of use cases are below.

2.1 Use Case 1

Number	1	
Name	Add Applicant	
Summary	User adds a applicant to the database	
Priority	5 (<i>5 is high priority, 1 is low</i>)	
Preconditions	Database is not at maximum capacity	
Postconditions	Row of applicant's info is added to the table, with required fields filled out	
Primary Actor	Admin, Developer, or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User selects "Add Applicant" option	
Main Scenario	Step	Action
	1	System displays a blank form for the user to fill out (starred fields for required info)
	2	User fills out all possible info
	3	User confirms their entry to database
	4	System updates database
	5	System returns to home page
Extensions	Step	Branching Action
	2a	User selects "Cancel" : Return to homepage
	3a	User selects "Cancel" : Return to step 2
	4a	Not all required fields are filled out : Return to step 2
Open Issues	None	

2.2 Use Case 2

Number	2	
Name	Edit Applicant	
Summary	User edits an existing applicant in the database	
Priority	4	
Preconditions	The applicant is in the database	
Postconditions	Info is updated for the applicant	
Primary Actor	Admin, Developer, or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User selects “Edit” for a particular applicant	
Main Scenario	Step	Action
	1	System opens the applicant info form (according to user’s access level) for the user to edit (starred fields for required info)
	2	User edits or adds info to any desired fields
	3	User confirms their entry to database
	4	System updates database
	5	System returns to home page
Extensions	Step	Branching Action
	1a	User selects “Cancel” : Return to homepage
	3a	User selects “Cancel” : Return to step 1
	4a	Not all required fields are filled out : Return to step 2
Open Issues	Access level of user will limit fields they can access.	

2.3 Use Case 3

Number	3	
Name	Remove Applicant	
Summary	User removes existing applicant in the database	
Priority	4	
Preconditions	The applicant is in the database	
Postconditions	Info is updated for the applicant	
Primary Actor	Admin or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User selects “Remove” option for a particular applicant	
Main Scenario	Step	Action
	1	System asks for confirmation of removal of the applicant
	2	User selects “Confirm”
	3	System updates database
	4	System returns to home page
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
Open Issues	None	

2.4 Use Case 4

Number	4	
Name	Log in to account	
Summary	The user logs into the site	
Priority	5	
Preconditions	The user already has an account in the system	
Postconditions	The user is logged in to the site	
Primary Actor	Admin, Developer, or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User selects “Log in” option	
Main Scenario	Step	Action
	1	The user enters their account info into the login form
	2	The user is logged in to the system
	3	System loads home page
Extensions	Step	Branching Action
	2a	Wrong credentials were entered
	2b	Prompt is displayed, and user & password fields are cleared, return to step 1
Open Issues	None	

2.5 Use Case 5

Number	5	
Name	Add database user	
Summary	Admin ability to add user	
Priority	5	
Preconditions	User account does not already exist	
Postconditions	A new user account is added	
Primary Actor	Admin	
Secondary Actors	User Database	
Trigger	Admin selects add account option	
Main Scenario	Step	Action
	1	Admin enters user account info
	2	System prompts for confirmation
	3	Admin confirms
	4	System adds new account
Extensions	Step	Branching Action
	2a	A field(s) was not properly filled out
	2b	Prompt is displayed, return to step 1.
Open Issues	Account with same login name exists.	

2.6 Use Case 6

Number	6	
Name	Edit database user	
Summary	Admin ability to edit user login info	
Priority	3	
Preconditions	User account already exists	
Postconditions	User info is updated	
Primary Actor	Admin	
Secondary Actors	User Database	
Trigger	Admin selects edit account option	
Main Scenario	Step	Action
	1	Admin edits user account info (name,username, password,etc.)
	2	System prompts for confirmation
	3	Admin confirms
	4	System changes account info accordingly
Extensions	Step	Branching Action
	2a	A field(s) was not filled out
	2b	Prompt is displayed, return to step 1
Open Issues	None	

2.7 Use Case 7

Number	7	
Name	Remove database user	
Summary	Admin ability to remove user	
Priority	5	
Preconditions	User account already exists	
Postconditions	User account is removed	
Primary Actor	Admin	
Secondary Actors	User Database	
Trigger	Admin selects remove account option	
Main Scenario	Step	Action
	1	Admin removes account
	2	System prompts for confirmation
	3	Admin confirms
	4	System removes account from user database
Extensions	Step	Branching Action
	2a	User hits “Cancel”, returns to homepage
Open Issues	None	

2.8 Use Case 8

Number	8	
Name	Edit Interview Questions List	
Summary	Ability to edit/add/remove interview questions	
Priority	3	
Preconditions	User is logged in with Admin level access	
Postconditions	Questions are updated with the new changes	
Primary Actor	Admin	
Secondary Actors	Applicant Database	
Trigger	User selects “Edit Interview Questions” option	
Main Scenario	Step	Action
	1	System opens the interview question form
	2	Admin edits the question form
	3	Admin confirms their changes
	4	System updates interview questions
	5	System shows preview of question layout
	6	Admin confirms layout
	7	System returns to home page
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
	5a	User selects “Cancel” : Return to step 1
Open Issues	None	

2.9 Use Case 9

Number	9	
Name	View Interview Form	
Summary	Ability to access and answer interview questions	
Priority	4	
Preconditions	Certain applicant is in the database that the user wants to look at/write their interview answers	
Postconditions	User is on the page to look at/fill out interview questions	
Primary Actor	Admin, Developer, or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User wants to look at/fill out interview answers	
Main Scenario	Step	Action
	1	System accesses answers for interview questions for particular user, if any
	2	System displays page with interview questions and respective answer fields to fill out
	3	User adds, removes, or makes changes to applicant interview answers
	4	User confirms changes, if any
	5	System goes back to homepage
Extensions	Step	Branching Action
	3a	User selects “Cancel” option: System returns to homepage
Open Issues	None	

2.10 Use Case 10

Number	10	
Name	Filter Applicants	
Summary	Ability to condense table to specific criteria	
Priority	2	
Preconditions	User is at the applicant table	
Postconditions	Table matching search criteria is displayed	
Primary Actor	Admin, Developer, or HR/Manager level user	
Secondary Actors	Applicant Database	
Trigger	User wants to filter applicants	
Main Scenario	Step	Action
	1	System displays check boxes for different criteria
	2	User selects search criteria, hits “Search” button
	3	System displays matching records/users
Open Issues	None	

3. Use Case Tests

3.1 Use Case 1 Test - Adding an Applicant

Fill out the applicant form with test information and confirm it. Then check and see if the applicant table has been successfully updated with the new applicant.

3.2 Use Case 2 Test - Editing an Applicant

Make sure there is at least one applicant in the table. Next, pick any applicant, select the edit option, and change some of the information. After confirming, check and see if the applicant table has been successfully updated with the changed information for the selected applicant.

3.3 Use Case 3 Test - Removing an Applicant

Make sure you have Admin or HR/Management level privilege and there is at least one applicant in the table. Next, pick any applicant and select the remove option. Then check and see if the applicant table has been successfully updated and the selected applicant is no longer in the table.

3.4 Use Case 4 Test - Logging In

Go to the login page and enter a username and password for an account that is already in the system. Then check and see if the System successfully loads to the homepage. Also, try entering in an invalid username and password to make sure the system correctly gives an error message.

3.5 Use Case 5 Test - Adding a User

Make sure you have Admin level privilege, and then select the option to add user. Make a test account with sample information and confirm it. Then check the account database to see if the new account has been successfully added. Also, try putting in invalid information when adding user to see if the system correctly displays an error message.

3.6 Use Case 6 Test - Editing a User

Make sure you have Admin level privilege and have at least one other account in the system. Then select the option to edit user, and change some of the information fields. After confirming, check the account database to see if the information of the selected account has been successfully updated. Also, try putting in invalid information when editing user to see if the system correctly displays an error message.

3.7 Use Case 7 Test - Removing a User

Make sure you have Admin level privilege and have at least one other account in the system. Then select the option to remove selected user. After confirming, check the account database to see if the selected account is no longer in the system.

3.8 Use Case 8 Test - Editing Interview Questions

Make sure you have Admin level privilege and select the option to edit interview questions. Make changes to the questions and confirm. Then check and see if the questions have been successfully updated.

3.9 Use Case 9 Test - Viewing Interview Form

Make sure there is at least one applicant in the table. Choose an applicant and select the option to access their interview answers. Add or change any answers, then confirm changes. Then access the answers again to make sure the changes have been made.

3.10 Use Case 10 Test - Filtering Applicants

Go to the applicant table and specify some options to do a filtered search. After confirming, check and see if the returned table is consistent with the filter options.

4. Non-Functional Requirements

#	Item	Priority (1:Lowest, 5:Highest)
1	Database shall be secure	2
2	Reliable environment	5
3	Application should be easy to use	3
4	The system shall be written in JavaScript	2
5	Application should be extensible	4
6	Application is easy to maintain	4
7	All code should be well documented	3
8	Sensitive data should have privileged access	5
9	The application should not seem cluttered	2
10	The application should be visually appealing	1

5. User Interface

See “User Interface Design Document for Applicant Tracking Tool.”

6. Deliverables

Hard copies of each of the following:

- Systems Requirement Specification (Due: 10/11/17)
- System Design Document (Due: 10/25/17)
- User Interface Design Document (Due: 10/25/17)
- Administrator Manual (Due: 11/29/17)
- Code Inspection Report (Due: 11/29/17)
- Testing Report (Due: 11/29/17)

An electronic copy in a ZIP file containing the following (All Due: 12/11/17):

- Systems Requirement Specification
- System Design Document
- User Interface Design Document
- Administrator Manual
- All source code

7. Open Issues

- 1) Need to write the mechanism or develop a solution to limit different user classes from editing fields they are not supposed to have access to. (pg 6)
- 2) Need a plan of action if an admin creates an account with a previously existing login ID. (pg 8)

Appendix A – Agreement Between Customer and Contractor

Customer finds this *Applicant Tracker* acceptable and agrees to use our database system to search, review, and edit applicant records. Use cases are included in the functional requirements section above of the behavior between the system and user. Additional features will be provided in further development phases.

When and if future changes to this document occur a drafted new document will be created. Both a hard and electronic copy of both versions will be presented to the client for review. Upon approval, the draft will be finalized and signed off by both parties

Client

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Team

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Name: _____

Signature

Date: _____

Appendix B – 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: _____

Signature

Date: _____

Comments: _____

Name: _____

Signature

Date: _____

Comments: _____

Name: _____

Signature

Date: _____

Comments: _____

Name: _____

Signature

Date: _____

Comments: _____

Name: _____

Signature

Date: _____

Comments: _____

Name: _____

Signature

Date: _____

Comments: _____

Appendix C – Document Contributions

This document was written equally by all team members over the course of multiple group meetings. Every member helped brainstorm ideas for the use cases. Logan Rites and Steven Hargrove wrote many of the functional and nonfunctional requirements. James Baker and Brian Wilson wrote many of the use cases. Vihar Patel drew up the UML diagrams and wrote the remaining use cases. Arvin Siva wrote the use case tests. All writing and editing was done in group meetings and input was accepted from all parties.

Appendix D – Addendum

2.11 Use Case 11

Number	11	
Name	Add Job Role	
Summary	User adds a job role to the database	
Priority	5 (5 is high priority, 1 is low)	
Preconditions	Database is not at maximum capacity	
Postconditions	Job Role is added to the table, and is accessible by the rest of the system	
Primary Actor	Admin	
Secondary Actors	Job Role Database	
Trigger	User selects “Add Job Role” option	
Main Scenario	Step	Action
	1	System displays a blank text field for the user to fill out
	2	User fills out name of Job Role
	3	User confirms their entry to database
	4	System updates database
	5	System returns to admin control panel
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
	4a	Text field not filled out : Return to step 2
Open Issues	None	

2.12 Use Case 12

Number	12	
Name	Remove Job Role	
Summary	User removes existing job role in the database	
Priority	2	
Preconditions	The Job Role is in the database	
Postconditions	Info is updated for the Job Role table	
Primary Actor	Admin	
Secondary Actors	Job Role Database	
Trigger	User selects “Remove” option for a particular Job Role	
Main Scenario	Step	Action
	1	System asks for confirmation of removal of the Job Role
	2	User selects “Confirm”
	3	System updates database
	4	System returns to admin control panel
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
Open Issues	None	

2.13 Use Case 13

Number	13	
Name	Edit Job Role	
Summary	Admin ability to edit job role title	
Priority	1	
Preconditions	Job Role already exists	
Postconditions	Job Role title is updated	
Primary Actor	Admin	
Secondary Actors	User Database	
Trigger	Admin selects Job Role id	
Main Scenario	Step	Action
	1	Admin edits Job Role title
	2	System prompts for confirmation
	3	Admin confirms
	4	System changes account info accordingly
Extensions	Step	Branching Action
	2a	A field(s) was not filled out
	2b	Prompt is displayed, return to step 1
Open Issues	None	

2.14 Use Case 14

Number	14	
Name	Add Source	
Summary	User adds a source to the database	
Priority	3 (5 is high priority, 1 is low)	
Preconditions	Database is not at maximum capacity	
Postconditions	Source is added to the table, and is accessible by the rest of the system	
Primary Actor	Admin	
Secondary Actors	Source Database	
Trigger	User selects “Add Source” option	
Main Scenario	Step	Action
	1	System displays a blank text field for the user to fill out
	2	User fills out name of Source
	3	User confirms their entry to database
	4	System updates database
	5	System returns to admin control panel
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
	3a	Field not filled out : Return to admin control panel
Open Issues	None	

2.15 Use Case 15

Number	15	
Name	Remove Source	
Summary	User removes existing Source in the database	
Priority	2	
Preconditions	The Source is in the database	
Postconditions	Info is updated for the Source table	
Primary Actor	Admin	
Secondary Actors	Source Database	
Trigger	User selects “Remove” option for a particular Source	
Main Scenario	Step	Action
	1	System asks for confirmation of removal of the Source
	2	User selects “Confirm”
	3	System updates database
	4	System returns to admin control panel
Extensions	Step	Branching Action
	2a	User selects “Cancel” : Return to homepage
Open Issues	None	

2.16 Use Case 16

Number	16	
Name	Edit Source	
Summary	Admin ability to edit source	
Priority	1	
Preconditions	Source already exists	
Postconditions	Source title is updated	
Primary Actor	Admin	
Secondary Actors	User Database	
Trigger	Admin selects Source id	
Main Scenario	Step	Action
	1	Admin edits Source title
	2	System prompts for confirmation
	3	Admin confirms
	4	System changes account info accordingly
Extensions	Step	Branching Action
	2a	Text field was not filled out
	2b	Prompt is displayed, return to step 1
Open Issues	None	