Version 1.0

15/10/2018



CV MAnagement system

QA Consulting

overall documentation

# Team Structure

## Team Roles

Lucy Hamilton (Project Leader): Tester

Melvin Kamau (Scrum Master): Tester

Manish: Backend (MongoDB)

Divine:

Jordan:

Emile:

Karun (Scribe):

## Team objectives:

### User Stories

The 5 user stories that have been given by our product owner are:

* As a trainee, I want to upload my CV so that it can be sent out by the sales department.
* As a trainer, I want to find a particular trainee’s CV so that I can monitor their progress.
* As a user, I want to log in to the CV management System so that I can update and maintain my account.
* As a Training manager, I want access to all CVs so that I can keep an eye on all of the trainees, past and present.
* As a member of the soft skills team, I want to “flag” any CV that I see as inappropriate, so that it can be reviewed by the training manager.

# SPRINT 1

(Week 1: 15th October – 19th October)

**JIRA:** [**https://qacacademyyellow.atlassian.net/secure/RapidBoard.jspa?projectKey=CCS&rapidView=13**](https://qacacademyyellow.atlassian.net/secure/RapidBoard.jspa?projectKey=CCS&rapidView=13)

**Definition of Ready**: Karun/Jordan/Emile: - Have a relatively good understanding of how to use AWS, create buckets, and integrate with the cloud. - Interact with files and managing it. -Install AWS on all computers that we are using. -Install STS. Manish/Divine: -Mongo server up and running. -Spring app capable of database. **Definition of Done**: Backend: - Great understanding of AWS and cloud integration. Manish/Divine: - When ready to test the CRUD commands of the Mongo to Spring. -Code been reviewed. -Acceptance criteria met (POST information to mongo and GET the information) -Product owner accepts the information that can be posted/retrieved. AGREED BY EVERYONE IN THE TEAM.

## mUst: Log In system

* Log In system with a “Username” and “Password”, which also returns the Role of the user which will allow the front end to decipher which page to navigate to.
* The back-end logic checks if both are correct, and if one or another is incorrect, it returns ‘Person Not Found’ awaiting front end to develop this further.

## MUST: Upload cv

* MongoDB to SpringBoot working, CV stored as BSON.
* CRUD commands have been successfully tested.
* Component Testing completed on the repository.
* Still need to pass the CV from the front-end to the back-end.

## should:

* Front-end using React (trainee page)

SPA:

|  |
| --- |
| SPA: Design. |
| LOG IN page:   * Username * Password * Submit Button |
| Rough Idea of the Page Plan. Once Logged In. |
|  |

## could: use aws

* A lot of time into researching AWS, buckets.

# SPRINT 2

# SPRINT 3

# Test Plan: CV management systems

### Testing Goals

* Achieve 80% Statement Coverage

## USer REquirements

### Ammended User Stories (& ADditional)

### Specific Role REquirements

Brainstorm of all the requirements each user should be able to have.

| Role | TEST  ID | Functionality | Priority (1(high)-4(low)) | Pre-Requisite |
| --- | --- | --- | --- | --- |
| Any User (A) | A1 | Log In using ID(Email) (Form Validation) | 2 |  |
| Trainee (E) | E1 | Access only their own CV | 1 | E3 |
| E2 | (Optional) Access 3 CV’s of their own | 4 | E4 |
| E3 | Upload CV | 1 |  |
| E4 | (Optional) Upload 3 CV’s | 4 | E3 |
| E5 | Edit and Maintain CV | 1 | E3 |
| E6 | Delete CV | 3 | E3 |
|  | E7 | Successfully Log In and Access the correct Trainee page | 1 |  |
| Trainer (T) | T1 | Access ALL CV’s | 2 |  |
|  | T2 | Search box to retrieve CV | 1 |  |
| Manager Trainer (M) | M1 | Access ALL CV’s | 2 |  |
|  | M2 | Able to alter status of CV (flagged, approved, unapproved) | 3 |  |
| Soft Skills Team (K) | K1 | Access ALL CV’s (Read Only) | 2 |  |
|  | K2 | Able to alter status of CV (flagged, approved, unapproved) | 3 |  |
| Sales Department (S) | S1 | Access only approved CV’s (Read Only) | 2 |  |
|  | S2 | Send off CV’s | 3 |  |

# Tests & Tools

Identifying all the specific different tests and requirements that are needed.

## CRUD commands

### Backend (Mongo DB to Spring)

Test ID’s (E3, E5, E6, T2)

### Component TEsting

### Repository

| Command | Description | Priority | Complete |
| --- | --- | --- | --- |
| RetrieveByEmail | Checking the PersonRepository | 1 | PASSED |

### 

### Controller

| Command | Description | Priority | Complete |
| --- | --- | --- | --- |
| Log In | Checking the logic of the backend log in layer | 2 |  |

### Integration TEsting

| Command | Description | Priority | Pre-Requiste | Complete |
| --- | --- | --- | --- | --- |
| Create | POST details to the mongoDB | 1 |  | PASSED |
| Read | GET ALL CV’s | 2 | Create | PASSED |
| GET CV’s | 1 |  | PASSED |
| Update | PUT details to particular CV | 1 | Create | PASSED |
| Delete | DELETE particular CV | 3 | Create | PASSED |

## Statement Coverage

### successful log in

For each of the 5 different roles listed, it will need to be tested that once they have successfully logged in, that they only have access to the correct contents on the ‘CV management system’.

### tools

Cucumber, Selenium, RESTAssured, WebDriver.

|  |
| --- |
| Feature File in Cucumber + RESTAssured. Automated testing using Selenium to ensure that the log in with the correct username and password, and also that they log in under the correct role. |
|  |

|  |  |
| --- | --- |
| https://documents.lucidchart.com/documents/3688199e-f736-417d-b3df-69f8ac298679/pages/YGcM5DNywbTK?a=1319&x=-13&y=-987&w=1314&h=3214&store=1&accept=image%2F*&auth=LCA%201147f1884cbdba3c11158a8e2614234da2f661c6-ts%3D1539862401 | Flow Chart for the log-in and to check if it leads to the right page. |

### Exit Conditions

Document the expected exit conditions.

## Test documents

Use the test documentation table below to access each documentation from all the tests from ‘CV management system’ project, brief description, who was responsible and last updated (date).

### Test Documentation table

| Document | Recipients | Description | Date |
| --- | --- | --- | --- |
| Report.html | Name | Responsibility | Number |

## Risks and issues management

### Potential Risks, Exceptions & Problems

* List all potential problems that might arise during the project, and list their causes, symptoms, consequences, and possible solutions.

### Appropriate corrective measures

For each issue, identify the optimal way to resolve the issue and then identify the steps that your team needs to take in order to implement the resolution.

### Incident Reports

In the following table, we will record all the incident reports.

| Date recorded | Risk description | Probability | Impact | Mitigation plan |
| --- | --- | --- | --- | --- |
| Date 1 | Description | Probability | Impact | Plan |

### Extent Reports

In the following table, we will record all the extent reports.

| Date recorded | Risk description | Probability | Impact | Mitigation plan |
| --- | --- | --- | --- | --- |
| Date 1 | Description | Probability | Impact | Plan |