CS261: Software Engineering Project Requirements Analysis Report - Group 23

Harry Fallows, Simon Heap, Bhavik Makwana, Pavnish Mohan, Antonia Newey

Preface

This report details the project commissioned to design an automated recruitment system, primarily in technology. The report is intended to be read by the client, the project manager, developers, testers and those maintaining the code to reduce ambiguity surrounding the project whilst explaining any technical terms used throughout the report.

Introduction

Large numbers of candidates may be deal with shifting trends within the industry. They are consistently being challenged in the search for new talent; this is caused in part by frequent changes in the technological requirements of candidates. Because of this, the traditional application and interview process needs to be replaced by a smart application system, which is the aim of this project.

This software will make decisions based on candidates CVs and performance in various tests. This will produce a list of candidates to be put through for interview, reducing the need for human filtering and allowing the client to focus solely on the candidates best-suited for the job. A report will be returned to the client containing the list of candidates, along with justification for choosing them. This report will allow the client to evaluate the system and, if required, refine the hiring criteria.

Project Stakeholders

The Client Client who commissioned the project
Dr. James Archbold: Module leader for CS261
Marcus King: Supervising tutor for the project

Glossary

Candidate: Refers to persons applying to the system
Client: Refers to the
Level A: Refers to the Web Content Accessibility Guidelines (WCAG) 2.0 level system[1]

Questions

Q: *Does the deliverable need to be a live system OR run locally as a proof of concept?*

A: The final deliverable is fine to be run locally.

Q: *What would be the role of the project supervisor?*

A: To oversee the groups progress and ensure the project is going down the right path.

Q: *Does the system need to be developed to vet candidates outside of technology? Or is that outside of the scope of the requested product?*

A: This idea was mentioned by James but was not listed as a requirement, therefore it is not high priority.

Q: *Is it necessary for the web app to be rated according to the Web Content Accessibility Guidelines (e.g. A, AA, AAA)?*

A: This is not specifically required by the client, especially in a prototype, but would be good to consider.

Q: *Are external systems such as Hackerrank allowed to be implemented?*

A: Open-source software is acceptable to use; it must be unlicensed as the product will be commercial.

User Requirements Design

Functional

- **R.1** Handling application details: The candidate will be able to submit data through a website - Must Have
- **R2** Online candidate non-technical testing: Candidates will complete multiple choice quizzes each with four options available to test their numeracy, literacy, abstract reasoning and situational judgment skills. - Must Have
- **R.3** Online candidate technical testing: Candidates will complete a short programming challenge - Should Have
  - **R.3.1** This challenge will be written in the language relevant to the job/role they are applying for - Could Have
- **R.4** Community contributions: Candidates will be able link their git repositories and technology forums, so the system can take into account their community contributions - Should Have
- **R.5** Candidate feedback:
  - **R.5.1** Candidates will receive quantitative feedback on their results in the numeracy, literacy, abstract reasoning, and situational judgment tests - Must Have
  - **R.5.2** Candidates will receive quantitative feedback on their results in the technical online tests - Should Have
  - **R.5.3** Candidates will receive feedback on their CV - Could Have
  - **R.5.4** Candidates will be sent a copy of this feedback - Could Have
- **R.6** Candidate login
  - **R.6.1** Candidates will be able to login to view their feedback report. - Should Have
  - **R.6.2** Candidates details are saved for use in future applications within the company - Could Have
- **R.7** Application selection
  - **R.7.1** The candidates' CVs will be compared to the job specification and scored relatively
  - **R.7.2** Allow the client to select either:
    - **R.7.2.1** An application cut-off date for a number of candidates to be selected by - Should have
    - **R.7.2.2** A report of candidates whose volume and regularity are chosen by the client (e.g. a weekly report of 10 candidates) - Should have
  - **R.7.3** Allow the client to specify how many candidates should be selected by the system per position - Should Have
    - **R.7.3.1** If no candidates pass the interview stage the client will be given the option to re-open the application process - Should Have
  - **R.7.4** Allow the client to stop the system from selecting potential candidates - Should Have
  - **R.7.5** Evaluate applications based on their CV using a machine learning model - Must Have
  - **R.7.6** Evaluate candidates based on their performance on the online tests. Select the candidates that meet the minimum threshold for each test from R.2, R.3 - Must Have

  - **R.7.7** Evaluate candidates based on their community contributions - Should Have
  - **R.7.8** Select candidates by combining their performance on the tests, the contents of their CV (and their community contributions - Should Have) - Must Have

- **R.8** Client interface:
  - **R.8.1** Generate a report for the client for the relevant job - Must Have
    - **R.8.1.1** The report will list each candidate recommended for interview - Must Have
    - **R.8.1.2** The report will contain the CV information of each candidate - Should Have
    - R.8.1.3 The report will contain the online test scores of each candidate - Should Have
  - **R.8.2** The client will be able to log into the system - Must Have
    - **R.8.2.1** The client account will contain candidate reports - Must Have
    - **R.8.2.2** The client account will be able to create new job listings - Must Have
  - **R.8.3** The client will be to create job listings with specific requirements, such as experience in different programming languages - Must Have
  - **R.8.4** The client will be able to create new job listings from the clients existing job specification creation system - Should Have
    - **R.8.4.1** The client will be able to select the amount of candidates to proceed to the interview stage - Should Have
    - **R.8.4.2** The client will be able to select a deadline for the specified position - Should Have
    - **R.8.4.4** The client will be able to view the constantly updating candidate list - Could Have
  - **R.8.5** The client will be able to comment on the success of the model, by rating the quality of candidates on a scale of 1-10. This will help to train the model - Must Have
    - **R.8.5.1** The client can reject the models chosen candidates. This is regarded as a failure of the model - Must Have
    - **R.8.5.1** The client can accept the models chosen candidates. This is regarded as a success of the model - Must Have
  - **R.8.6** The client will be able to immediately contact the candidate, invite them to an interview through the report - Could Have
  - **R.8.7** Mobile application - Should Have
    - **R8.7.1** Developed for Android - Should Have
    - **R8.7.2** Developed for iOS - Could Have
    - **R8.7.3** View all the candidates that have been invited to the interview stage - Must Have

Non-Functional

- **R.9** Ease of Use:
  - **R.9.1** The system will be suitable for non-technical users - Should Have
    - **R.9.1.1** The system interface makes no use of any technical language - Should Have
  - **R.9.2** The system requires no prior training to use - Should Have
    - **R.9.2.1** The system can be accessed through a graphical user interface with a clearly defined structure and well-labelled buttons and menus - Should Have
  - **R.9.3** The system conforms to web accessibility guidelines - Level A[1] - Should Have
    - **R.9.3.1** The user interface has a high colour contrast between text and the background to ensure readability - Should Have
    - **R.9.3.2** Colour is never used as the only distinguishing feature of an item on screen - Should Have

    - **R.9.3.3** All non-text content will have alternative text captions to allow use of screen-readers or other assistive technology - Should Have

- **R.10** The system is responsive - Must Have
  - **R.10.1** The system will be able to display correctly on most screen sizes - Must Have
    - **R.10.1.1** The system loses no functionality on smaller screens - Must Have
    - **R.10.1.2** The system remains visually appealing on smaller screens - Must Have
  - **R.10.2** System performance is validated - Must Have
    - **R.10.2.1** The system should aim to keep processing times to a minimum and display a progress icon for longer wait times - Must Have
  - **R.10.3** The system remains up to date - Should Have
    - **R.10.3.1** The system should aim to meet the clients most recently stated requirements - Should Have
    - **R.10.3.2** The client will be able to change their requirements and the machine learning model will adapt appropriately. - Should Have

Outside of Scope

Implementing security practices into the system to lower the risk of a successful cyber attack on the application.

Decision on Group Management

This project will use the Scrum software development methodology; scrum is an agile framework geared towards rapid development of software. To ensure maximum effort and input from every member in the group the Scrum master will be rotated on a weekly basis, therefore there will be no permanent team leader.

Meetings

The group will meet three times a week. Due to the remote location of group members, there will not be in-person daily Scrum meetings.

- **Monday 1pm-2pm:** Weekly sprint review. Backlog of tasks is monitored and each members tasks for the week are identified.
- **Thursday 11am-11:30am:** Short meeting with the project supervisor to ensure the development approach is oriented correctly for the client.
- **Friday 1pm-4pm:** Communal work session. Work topic for each member is selected from those allocated in the sprint meeting earlier in the week.

Technical Roles

Technical roles will need to be assigned to each member of the group, this will ensure the scope of work for each member is clear and will keep each member accountable for their contribution to the project.

- **Front End:** The front-end developers will be focusing on the creation of the client and user-facing parts of the application. - Pavnish, Antonia
- **Back End:** The back-end developers will be working on the processing of data that is given to the application. - Simon, Harry
- **Machine Learning:** This role will concentrate on implementing the adaptive CV selection. - Bhavik

References

[1] Eric Eggert SAZ. How to Meet WCAG 2. W3C;. https://www.w3.org/TR/WCAG20/.