1. Choices & Reasons for Selection of Techniques

This section will illustrate what techniques will be used for eliciting, specifying, and prioritizing requirements and how these techniques are chosen.

1.1 Techniques for Eliciting Requirements

Inspired by 9 elicitation techniques (Ghoddousi, 2016) identified by BABOK, the following techniques will be used in the project.

Brainstorming:

Group members will first individually come up with as many as ideas on the topic based on our initial researches, analysis, and problem statement as well as the personas from the previous coursework. Adopting brainstorming at the first place is expected to produce numerous ideas in a short period of time and provide insights for further analysis. Since no discussion is required when team members first share their ideas, it is expected to trigger creative thinking of all members under a non-judgmental environment. Therefore, it is useful to encourage all non-experts team members to participate into the process.

Document Analysis:

Group members are encouraged to look for relevant documents and analyze on existing solutions. It is expected to provide insights from existing solutions and hopefully can exploit the unmet needs from target groups. Since all team members are not experts on engineering or software development projects, it is useful to provide examples, generate insights and benchmark the requirements elicited from other techniques.

Prototyping:

By visually presenting some imaginary interfaces of the web apps, team members are expected to have a clear understanding on what the solution will look like and it becomes convenient to find drawbacks and make improvements. Meanwhile, since the deliverables also require the team to submit evidence of designs, the technique aligns with another task, thus can boost the working process.

Interface Analysis:

Since the solution application will definitely include interfaces between users and the system, and between hardware devices, the interface analysis can help to identify the boundaries between applications and distinguish what each role needs to do. With the subsequent analysis, requirements are expected to be generated in more details. It also helps to think about non-functional requirements.

1.2 Techniques for Specifying Requirements

Since the previous coursework has determined to adopt Scrum methodology in software development process, user stories will be used to specify requirements as an essential part of Agile method. By refining user stories in each sprint, it helps to set the ground for the conversations that need to be held about the functionality. It provides an easy-to-follow framework for the team to better emphasize target user groups and think about real demands for them rather than detailed system description. It also drives momentum as the development team can feel a sense of achievement by taking challenge from each small stories. (Rehkopf, undated)

1.3 Technique for Prioritization

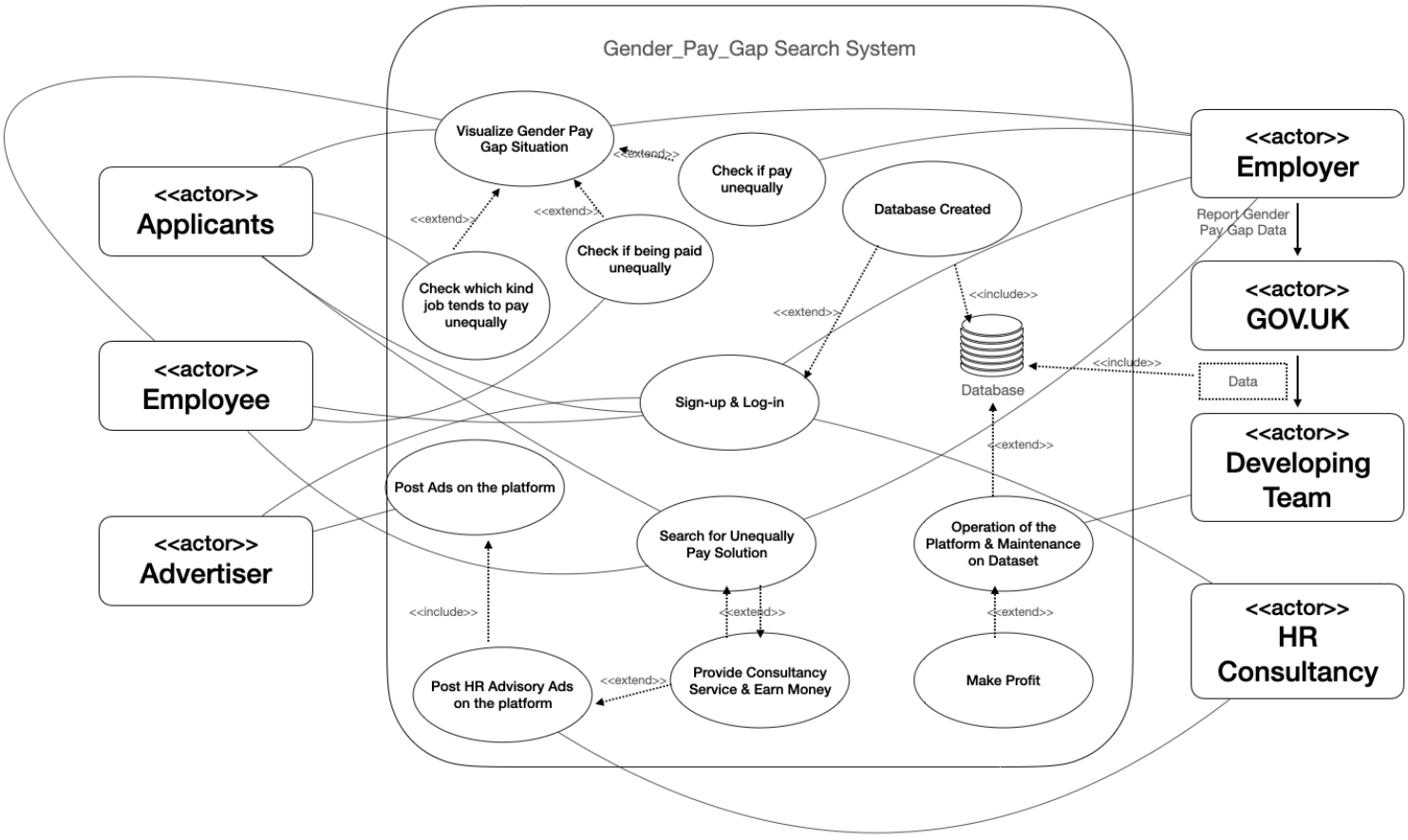
In order to prioritize our requirements, **“MoSCoW”** approach will be used to categorize requirements into four categories:

- **M = M**ust (have)
- **S = S**hould (have)
- **C = C**ould (have)
- **W = W**on’t (have)

The technique is selected as it is a quite simple way of categorization and is friendly no non-experienced developers in the team. After classification, a natural order of implementation is expected to appear. (However, there may be a lot of “must”s in practice since those who think out the requirements tends to believe their proposed requirements to be important. Therefore, our team decides to put a requirement as “must” only if at least 3 out of 4 group members agree with it.

2. Use Case Diagram

A use case diagram is drawn to help visually understand the connections among the system, target users, developers and other potential relevant stakeholders.



3. Specifying & Prioritizing Requirements

No.	User Story	Acceptance Criteria	Functional / Non-Functional	MoSCoW
US01	As the developing team, I want app users can sign-up and log-in so that I can know my users better.	1. Users should register before using the app by setting up user name and password; 2. Users can only access to the application when user name and password are matched when log in; 3. Users should at least choose their role from “employer”, “employee” & “applicant” when registration	Functional	M
US02	As the developing team, I want users registration data be store in a database so that I can build a foundation for future monetization.	1. User name and password will be stored to check log in information; 2. User information including gender, age, occupation / expected occupation, industry and city should also be included; 3. Usage data such as browsing time should also be included.	Non-Functional	M
US03	As the app user, I want to clarify my role when sign-up so that I can get my concerned information in the future.	1. Users can choose their role from “employer”, “employee” & “applicant” when signing up	Functional	M
US04	As the app user, I want it to be accessed on mainstream browsers including Chrome, Firefox & Safari so that I can have access conveniently.	1. Support recent 5 versions of the browser; 2. Comply with cookie policy	Non-Functional	M
US05	As the app user, I want my information to be private without consent so that I can keep my privacy secure.	1. Users should sign consents before entering the app; 2. Firewalls should be built to the database storing user information.	Non-Functional	M
US06	As an employer, I want to know whether my company is paying unequally compared to companies in the same industry so that I can know if I am being disadvantaged.	1. Able to get visualized results in a box-lot chart showing their companies’ level of salaries gender difference, bonus gender difference, and percentage of highly paid women compared with other companies in the same industry by inputting those information of their company and their industry; 2. Get the result in less than 2 seconds	Functional	S

US07	As an employer, I want to know whether my company is paying unequally compared to companies in the same region so that I can know if I am being disadvantaged.	1. Able to get visualized results in a box-lot chart showing their companies' level of salaries gender difference, bonus gender difference, and percentage of highly paid women compared with other companies in the same region by inputting those information of their company and their region; 2. Get the result in less than 2 seconds	Functional	S
US08	As an employer, I want to know whether my company is paying unequally compared to companies in the same company size so that I can know if I am being disadvantaged.	1. Able to get visualized results in a box-lot chart showing their companies' level of salaries gender difference, bonus gender difference, and percentage of highly paid women compared with other companies in the same company size by inputting those information of their company and their company size; 2. Get the result in less than 2 seconds	Functional	S
US09	As a job applicant, I want to know which industry is most likely to have highest gender pay gap so that I can judge which industry is worth or not worth going.	1. Can visualize gender pay gap situation including of salaries gender difference, bonus gender difference, and percentage of highly paid women in selected industry; 2. Able to get mean, median, maximum value, minimum value, and range of gender pay gap.	Functional	S
US10	As an employee, I want to know whether I am paid unequally against gender so that I can have reasons to argue or switch jobs.	1. Can obtain result of mean and median payment when input company size and industry; 2. Can compare the result with own payment situation and gain a sentence to conclude situation.	Functional	S
US11	As an HR consultancy service provider, I want to get clients from the app so that I can boost revenue.	1. The consultancy company can get information when users request; 2. The consultancy company can chat with needer of their service.	Functional	C
US12	As an employer, I want to leave contact information so that consultancy service can reach out to me when I found my company is paying unequally.	1. The employer can leave their contact information on the app	Functional	C
US13	As an employer, I want to leave a message so that consultancy service can know my situation before reaching out to me.	1. The employer can leave a message.	Functional	C

US14	As an employer, I want to chat with consultancy service so that I can get instant response.	1. The employer can chat with our partnered HR consultancy service provider with chatbot on the website.	Functional	C
US15	As an advertiser, I want to post ads precisely to my target user group so that return on investment can be improved.	1. Target groups can be sorted by gender, age and region when posting ad campaigns. 2. Able to visualize ad performance from dashboard. (Split)	Functional	W
US16	As an advertiser, I want to manage advertising campaigns from an advertising platform so that I can flexibly change ad strategy.	1. A separate advertising platform should be built for advertisers; 2. Advertisers can edit and post campaigns from the platform.	Functional	W
US17	As an advertiser, I want to visualize ad performance so that I can know where to improve.	1. A dashboard showing clickthrough rate & ROI can be viewed by advertisers; 2. Ad cost can be shown on the dashboard.	Functional	W
US18	As a developing team, I want the dataset to be automatically updated once GOV.UK releases new data so that results are up to date.	1. The database can be automatically updated when original data changes on GOV.UK to give real-time update to users.	Non-Functional	W

US04, as a non-functional requirement is listed as “Must Have” since it is essential for users to have access to the application, without it, it is non-sense to talk about other requirements. Meanwhile, two functional and one non-functional requirements on registration and log-in have been listed as “Must Have” as well, given it is designed as the first stage of using proposed application since the developing team wish to have a clear database on app users for future development. Requirement on protecting users’ data privacy has also been listed as mandatory.

The app users are expected to involve employers, employees and job searchers (applicants), for each types of user groups, the app should be able to present them data visualizations and summarized results from what users input and what information they demand. Therefore, from US06 to US10, functional requirements are listed as “Should Have” and will be selected in the later stage by developing team based on complexity of having specific function, hopefully all requirements can be met with the limited project development time. An ideal situation for the application is to expand its monetization ability by involving solution providers when business problems are found. Therefore, four functional requirements which can allow employers to connect with HR consultancy service providers have been listed as “Could Have”. It is not mandatory at this stage of the project and actually cannot be managed to do without reaching out to service providers. It is just a visionary expansion path of the project.

Finally, four requirements are prioritized as “Won’t Have”. Requirements from US15 to US17 are listed because they are technically challenging to the software development team and is far beyond what the basic task of the project requires with a ad platform. This function can be compromised by adding ad slots and join in some ad alliance so that advertisers can post ads on more mature platforms. However, this is not necessary for this project anyway. In addition, these requirements are not directly linked to the basic aim of the project so it is temporarily listed as “Won’t Have” but can be considered to be included in the future. US18 is technically not feasible for the developing team since the dataset released by GOV.UK is not always clean and require data processing. It cannot be perfectly done with preset algorithms and requires manual work.

Reference List

Ghoddousi, E. (May, 2016) *The BABOK Identifies 9 Elicitation Techniques*. LinkedIn. Available at: https://www.linkedin.com/pulse/babok-identifies-9-elicitation-techniques-elham-ghoddousi/ Accessed: Nov 2022

Rehkopf, M. (undated) *User Stories with Example And A Template*. Atlassian. Available at: https://www.atlassian.com/agile/project-management/user-stories Accessed: Nov 2022

Smartpedia (undated) *MoSCow Method*. t2informatik. Available at: https://t2informatik.de/en/smartpedia/moscow-method/ Accessed: Nov 2022