CS261 Coursework Requirements Analysis

Group 17: *Ben Lewis, Dan Risk, Edmund Goodman, Jay Re Ng, John-Loong Gao, Rahul Vanmali, Tomás Chapmann Fromm*

1 Introduction

This document indicates our current understanding of the proposed project, and will be used alongside Deutsche Bank to clarify our position. We detail and analyse the requirements for a proposed mentoring software platform which should function internally at Deutsche Bank. Many employees want a broader understanding of how different departments within the company work in unison to successfully complete business projects. Alongside this, employees may also have weaknesses in certain business areas, which could be improved by working with more experienced people in different departments. A robust mentoring software platform would solve these issues, by appropriately assigning users knowledgeable mentors who could help them at a personal and professional level.

2 Glossary

- **Mentor:** The users who can approve a requested meeting and select confirm a date.
- **Mentee:** The users who can request a meeting in a series of targeted dates.
- **Plan of action:** The future task and achievement milestones that can be created by mentee and mentor, and mentee can mark it as complete or not.
- **Meeting:** This is the event that is held by a mentor and a mentee in a specific time duration. Mentors teach business knowledge to mentees. After the meeting, all users are available to provide feedback.
- **Workshop:** This is an event that is held by multiple people. The main purpose is to major in mentees' weaknesses.
- **Feedback:** Mentor and mentee can summarise a review with each other, based on their behaviours and sentiment during the whole training period.
- **Group session:** This function is nearly the same as a workshop, except there is no set topic.
- **Event:** A broad term that means meeting, workshop or group session.

3 User stories

User A has recently graduated from university, and has joined Deutsche Bank in the Human Resources department. Coming from a non-technical background and being thrown into a very tech focused environment has led the user to believe it would be beneficial for them to have a general idea of the technologies they were using worked under the hood. They would therefore like to be paired with an experienced mentor who knows the inner workings of Deutsche Bank's technology system.

User B has been working as a manager at Deutsche Bank for 7 years, and has mentored many employees throughout the years. The user notes that many of the mentees have a keen interest in how conflict in the workplace is handled from a managerial position. Rather than having to teach the same principles repeatedly for each mentee, the user would like a system that would allow them to create group sessions where these management principles could be taught to multiple mentees at once.

User C has recently started mentoring employees at Deutsche Bank, and would like to do as much as possible to ensure their mentoring actually provides value to the mentees. Whilst the user believes the sessions are productive, there's no formal way for them to attain mentee feedback. The user would therefore like a system where mentees could provide feedback after each session. This would then allow the user to make changes about the way they mentor, to maximise their value to each employee.

4 Functional requirements

Functional requirements	Justification
F00 (Must Have) The user can register an account. *The user will be able to create an account by providing an email and password, and add data about themselves.*	A user account system is necessary for users to identify themselves and to access the application's functionality.
F01 (Could Have) The website will teach new users with tutorials. *The first time after the account is created, a tutorial for the major features of the website will be provided.*	Users are provided enough information to fully understand the application's features and functionality.
F02 (Must Have) The user will be able to log into their account using their email and password and reset their password via email if they need to. *The system will check the database to see if a matching account exists, and send a password reset link if required.*	User accounts are required to match up individuals. Being able to reset the password by email is important as people often forget their passwords.
F03 (Must Have) The user will have a profile page, allowing them to view and modify data. *Data includes: their email address; their name; their job title; whether they want to be a mentor, mentee, or both; their business area; 1-5 strengths; and 1-5 weaknesses. A "save" button will be used to update any modified data and ensure that no essential data is left empty.*	Users must be able to update their profile page to keep the system up-to-date, and the profiles must include their basic information along with weaknesses and strengths in order for the system to suggest possible mentor and mentee pairing.

F04 (Must Have) Any changes to the profile which will break the rules of mentoring between current relationships will notify the user. *The notice is a warning and still allows the user to break the relationship.*	This forms a good compromise between maintaining the rules of mentoring, and not accidentally breaking good relationships
F05 (Must Have) Allow mentees to request a mentor, who will have strengths matching their weaknesses. *The system will add the user to the list of people looking for mentors.*	In order for mentors to be paired with mentees, mentees must first seek a mentor by specifying an area of weakness.
F06 (Must Have) When selecting a mentee, a mentor will get access to their profile to help the mentor make a decision. *Data will include their name, strengths, weaknesses, job title and business area.*	This allows the mentor to gauge which mentee is most suitable to them, adding a human element to the matching process.
F07 (Must Have) Mentors will be able to select mentees to mentor from a list of mentees who satisfy the rules of mentoring, ordered by the likelihood of them being a good match with the mentor. *The list will be filtered to exclude invalid mentees, e.g. same business area, and the total order for the list of mentees to select from will be based on matching metrics using data stored by the system.*	This complements the above requirement by supplementing the human element of matching with a computational one which filters invalid matches, and encourages ones which would be good based on feedback and aggregated metrics.
F08 (Must Have) When a mentor selects a mentee, the mentee will be able to accept or reject the offer of mentoring. *The mentor profile is shown, and, if rejected, the mentee will then not be shown to that mentor for a specified period of time.*	This allows mentees to view the profiles from their available mentors and choose who they see the most fit, and avoids mentors repeatedly requesting mentees in a short time.
F09 (Must Have) Mentors will be prompted to select mentees if they are not mentoring anyone. Mentees will only be allowed to have one mentor at a time. *Mentors can have several mentees if they want, however the system must follow the rules of mentoring.*	Mentors should always have someone to mentor if there is any mentee compatible with them, as otherwise there may not be enough mentors.
F10 (Must Have) Both mentors and mentees will be able to terminate the relationship at any time, and feedback should be required for why it was terminated. *This feedback includes a numerical rating and text to be shown to the other person.*	Termination of any mentor/mentee relationships can be due to reasons of poor fit. Feedback is also necessary for mentors to improve their mentorship.
F11 (Must Have) Mentees will be able to propose a meeting, providing a brief description of the agenda of the meeting, and/or the category it relates to. *The agenda will be free text whereas the category will be from the discrete list of strengths/weaknesses.*	This follows the first rule of mentoring where the mentee must drive the relationship whenever they need unblocking.
F12 (Should Have) Mentors will be prompted to suggest three meeting start/end times that would work for them (and info about where it will be). *Must ensure that the meetings are not at the same time as another event in the system.*	This is to allow for flexibility in meeting time proposals by the mentor, whilst still following the rule of mentors giving up time.
F13 (Must Have) Mentees will be able to either accept one of the meeting start/end times, or send a request back to the mentor asking for more meeting start/end times. *The mentor must give up time to the mentee so this cycle is required.*	For a meeting to go ahead, both parties must be free for it. This confirms this, given the above requirement of mentors suggesting times for meetings.
F14 (Must Have) After the end time of the meeting has passed, both users will be prompted to provide feedback on the meeting. *They will summarise the meeting with a free text field, and numeric metrics, and whether the meeting was attended and be able to see the other's feedback.*	This is to allow the recording of discussions and for the users to reflect on the meeting where necessary.
F15 (Could Have) Sentiment analysis will be used on the free text feedback provided by users. *This will provide a supplementary metric which can be analysed for use in planning by the system.*	Additional metrics mean more data which can improve the way events or mentees are presented by the system, encouraging users to make decisions the system believes will be more optimal.
F16 (Must Have) Every user will start with their own empty plan of action when they create an account, to which they can add milestones. *Users can have plans of actions without a mentor, and the milestones will contain a description, a priority, and a target completion date*	Plans of action have milestones to keep track of the progress made by the mentee. It is important for the mentee to have set goals and priorities to work towards.
F17 (Must Have) When a user starts being mentored, their mentor will be able to see their plan of action. *The mentor will only be able to see milestones after they started mentoring the mentee.*	The long term goals of a user are not necessarily linked to a mentor, so a plan of action which persists across mentors is beneficial. However, such plans of action may contain sensitive information from the past a mentor should not have access to.
F18 (Should Have) Milestones will be able to be marked as completed by mentees, and if a milestone has not been completed after its target completion date, mentees will be prompted to mark it as complete or update its target date. *This prompt will be given the next time the mentee signs in.*	This ensures that the mentee is able to see their progress, and the mentor can provide further guidance to the mentee if they remain on a milestone for a prolonged period.

F19 (Must Have) General feedback will be able to be provided at any time. *Including free text submissions and updateable numerical metrics, and feedback for specific topic areas will be provided in the same way to refine the suggestion process.*	This is so feedback can be provided outside of just meetings and termination of relationships.
F20 (Should Have) If there are enough people with a specified weakness, people identified as experts in that area will be prompted to organise a workshop on it. *Using prior feedback, the system can determine if a person is an expert.*	Experts are able to run workshops to cover a specific 'weakness' topic extensively and potentially through a group setting.
F21 (Must Have) To create a workshop, the user will set a workshop start/end, and a brief description of what the workshop will cover within the specified weakness category. *The workshop will then be registered as an event within the system that anyone can go to.*	Information on the workshops can be viewed on the dashboard on the site. The description should provide attendees an idea of what the workshop will focus on.
F22 (Must Have) Group sessions will be created, run, and reviewed in the same way as workshops.	Group sessions are equivalent to workshops within the system, with the one difference outlined in the glossary not being in the system.
F23 (Must Have) Signed in users will be able to submit feedback/bug reports to the system. *There will be a button at the top of the dashboard that allows you to go to the feedback screen.*	Feedback/bug reports is crucial to the maintenance and further development of the application.
F24 (Should Have) The user will be able to disclose security issues. *There will be a security.txt file [1] for responsible disclosure ofsecurity issues - not necessarily requiring the user to be signed in.*	System administrators must be made aware of any security issues, so they can resolve them promptly.
F25 (Must Have) The system will have a dashboard that gives a personalised view of the users current meetings, plans of actions, workshops/group sessions and access to the profile page. *This will be the main page the user gets placed on when they log in.*	Users must be able to navigate the site easily, so a dashboard where important notifications are shown will facilitate this.
F26 (Must Have) Users will be able to see a list of current workshops. *It will be an ordered list of workshops grouped into weeks and then ordered by rating, which can be filtered by the users weaknesses.*	This will allow users to easily select workshops most relevant to them.
F27 (Won't Have) The prototype system will not use Deutsche Bank branding. *A simple generic theme will be used, which could then later be easily changed to include Deutsche Bank branding.*	This is specified as out of scope by the project specification, and it would be simple to change to include it in the future.
F28 (Won't Have) The system will not support video conferencing or messaging. *Meetings will be set up as having times, and can then have either physical locations or video conference links included in their description.*	This is beyond the scope of the project, as meetings are specified to happen in person, or on a different system. Messaging could be convenient for scheduling meetings, however, this is also out of scope.
F29 (Should Have) Users will have an inbox of notifications which will indicate upcoming events such as meetings or milestone targets. *The inbox will be presented as a date ordered list, notifications will be recorded as read or unread, so the number of unread notifications can be indicated to the user.*	This allows users to easily keep track of events which are upcoming, as otherwise it would be easy for example to forget about a meeting which was scheduled some time in the future.

5 Non-functional requirements

Non-functional requirements	**Justification**
NF00 (Must Have) The system will be intuitive to use and navigate. *The system must have a UI that makes it inherently clear to the user what all parts of the interface do. Each part of the UI will have a clear, distinct and useful purpose that is unambiguous.*	The user is able to clearly identify the functionalities of each widget through concise and thorough UI design.
NF01 (Must Have) The system will be simple to use even for users with less technical experience. *The UI and the tutorial will make the system suitable for everyone.*	The user does not need any familiarity with the application before beginning to use it.
NF03 (Should Have) The system will be quick to respond to the user. *90th percentile of API request times will be under 1 seconds.*	The user should not feel any notable delay in the responsiveness of the system. This threshold is noted as the bound for retaining "user's flow of thought" by Jakob Nielsen [2].
NF04 (Won't Have) The system will be able to handle large and varied numbers of users. *This will not be explicitly supported, but design choices will be made to facilitate it in future development.*	This is out of scope of the prototype, however, it is useful to design so it can be easily scaled by future maintainers.
NF05 (Must Have) The system will be kept up to date. *The system will be easy to maintain and modularised so that each section of the software can independently be worked upon and inserted back into the system when doing maintenance. All code should be appropriately commented.*	The system should be regularly updated to improve performance and responsiveness.

NF06 (Must Have) The system will be easy to test and validate its properties. *The system will be modularised so that it can be tested on a system scale and so that each independent unit can be unit tested. Acceptance testing for the usability of the UI should be possible. CI/CD will be used to facilitate this.*	The system must be tested and validated to ensure the functionalities and UI work correctly and are following its intended design.
NF07 (Must Have) The system will follow all relevant laws. *The system will comply with data protection [3] and cookie laws [4], along with any other relevant legislation.*	It is paramount that the system is legal for it to be used in a corporate setting.
NF08 (Must Have) The system will employ good security practises. *Such as hashing/salting passwords and sanitising user input.*	Security is crucial to ensuring that any registered user's data is protected.
NF09 (Won't Have) The system will not have a dedicated mobile or desktop app, it will only run as a web app in a browser *Such as hashing/salting passwords and sanitising user input.*	It is out of scope to concurrently design multiple different types of app concurrently, so only the most widely accessible type will be developed.
NF10 (Should Have) The system will be accessible to those with disabilities [5]. *Techniques such as high contrast modes, enlarged text, and optimisation for screen readers will be used to facilitate use by all.*	It is important for companies to be inclusive, and ensuring that websites are usable by those with disabilities is required to achieve this.

6 Team organisation

6.1 Team roles

Team member	Role	Responsibilities
Dan Risk	Project manager	Schedule meetings, coordinate tasks for group members to balance team workload and ensure the project is on schedule to meet the deadline.
Ben Lewis Jay Re Ng John-Loong Gao	Website developer	Design front-end website layout and functions, interpret information sent from back-end and display it to the user.
Edmund Goodman Tomás Chapman Fromm Rahul Vanmali	Back-end engineer	Implement the system to interpret and process data received from the front-end, provide data to be displayed to the front-end, and manage effective data storage in the system database.

Additionally, Tomás Chapman Fromm was allocated as Business Analyst, and Ben Lewis and Edmund Goodman as team leaders for website development and back-end engineering respectively.

6.2 Scheduling

In-person group meetings Monday and Friday, as well as additional meetings or collaborative work time to help meet deadlines. Monday meetings to see progress on weekend tasks and set tasks for the week, Friday meetings to see progress on week and set tasks for the weekend.

References

[1] EdOverflow, Shafranovich, Yakov, "Security.txt standard." https://securitytxt.org/. Accessed: 2022-02-01.

[2] J. Nielsen, *Usability Engineering*. San Francisco: Morgan Kaufmann, 1993. ISBN: 0-12-518406-9.

[3] HM Government, "Data protection." https://www.gov.uk/data-protection. Accessed: 2022-02-01.

[4] HM Government Information Commissioner's Office, "Privacy and electronic communications regulations. cookies and similar technologies." https://ico.org.uk/for-organisations/guide-to-pecr/cookies-and-similar-technologies/. Accessed: 2022-02-01.

[5] W3C, "Introduction to web accessibility." https://www.w3.org/WAI/fundamentals/accessibility-intro/. Accessed: 2022-02-01.