

Requirements Analysis Report

Jack McCullough, Luca Orita, Maria Bracegirdle, David Yang Zhang, Rebecca French

1 Preface

This document defines a set of requirements which will fulfil the project specification provided by the customer, Deutsche Bank. [1] These requirements are the culmination of multiple project team meetings held between 14/01/2021 and 07/02/2021. The requirements laid out in this document will guide both the implementation and testing process and will aid in understanding the relationship between components once the system is completed. Table 1 below outlines the version history of this document.

Date	Version	Purpose
04/02/2021	0.1	First complete draft - sent to tutor
06/02/2021	0.2	Minor changes based on tutor feedback
07/02/2021	1.0	Final version

Table 1: A table showing the version history of this document

2 Introduction

This project aims to create a feedback assessment system that meets Deutsche Bank's product specifications. This is to provide feedback to improve upon the next time a workshop is held. Also, a live identification feature will be implemented so that the presenter can adjust and improve based on live responses. Given the large amount of information that will be sent to the system, the data will need to be sorted and displayed concisely, so that the event host can instantly get an overall feel for the audience's sentiments. [2] Furthermore, the product will be designed to reduce human intervention, making it easier to use as well as to optimise the workshop's time. Due to the differences in each video-telephony program, we have adjusted our design so that that it is not dependent on the platform used. To reach a large customer base, we have decided to make the feedback system accessible from a user's browser.

3 Glossary

Host: The person who creates an event.

Attendee: The person who creates an event.

Plan-Driven Approach: An approach where all of the activities to be completed are planned and success is measured against whether these activities have successfully been completed.

Waterfall Model: A plan-driven process model, in which activities to be completed are completely linearly, measuring progress against the initial requirements that are set out.

4 System Requirements Specification

The following section describes the functional and non-functional requirements of the proposed system, given in terms of customer-facing requirements and developer-facing requirements. The requirements have been ranked in order of priority, with number 1 being the highest priority.

4.1 Functional Requirements

Hosts

1. **C:** A host can create an event which will prompt them to select a feedback form template and a name for the event.
D: On the main page, there is a button labelled 'Create Event'. Upon pressing this, a host is taken to a screen where they select a template feedback form out of a set of options, including any that they have created. Once this is selected, the host enters a name for the event in a text box and then the event is created.
2. **C:** Once the host has selected a template feedback form for users to fill out, a 'room' is created and will be supplied with a unique code related to their room which should be shared with attendees.
D: Once the event is created, the event is given a random, unique code made up of numbers. The code should be between 5 and 8 characters in length. This is to ensure that the code is long enough to allow for enough combinations of room codes to existing such that every room is unique, as well as preventing a user from accidentally typing in an incorrect code.
3. **C:** Hosts can view the results of feedback from the past x minutes or from all time throughout the course of their event, where x can be changed by the host throughout the event to see the sentiment over some time.
D: There will be a tick box where the host can select to see general feedback from all of the event or select to see feedback from a set interval by inputting the number of hours and minutes to show feedback from. This option can be changed throughout the course of the event. By default, this option will be set to display general feedback from the event as a whole.
4. **C:** Once the host has chosen to create an event and has selected a template, the host can decide how often to acquire user feedback, allowing feedback to be recorded at set intervals, allowing users to submit feedback when they feel it necessary, or when prompted to give feedback by the host.
D: Once the host has selected the 'Create Event' button and chosen a feedback form to use, the host selects how often feedback should be acquired. The host chooses between acquiring feedback at set intervals, acquiring feedback whenever the attendees choose to submit it or collecting feedback when the host presses a button on the application to prompt attendees to fill out the feedback form. If the latter is chosen, a button should be presented to the host throughout the event which they can press to prompt attendees to submit feedback.
5. **C:** The host can press a button on the application to end the event.
D: When feedback is being collected, the host can press a button to end the event and stop the collection of any more feedback. Because the event has ended, the code should no longer let attendees join this event. Because this code is no longer in use, it is now acceptable for another event to use this code.
6. **C:** At the end of the event, the host can view the feedback collected throughout the event and how it changed over time so the host can see a general sentiment.
D: Once the host has pressed the button to end the event, a summary of the feedback collected, showing how the sentiment changed over time, including the general mood from sentiment analysis. The host can also view the specifics of each of the feedback forms.
7. **C:** A host can design a template feedback form to be displayed to attendees that will be specific to their event. Templates can be saved and re-used so that the host can easily reuse feedback form templates they have created. They can also remove template feedback forms they have created if they no longer need them.
D: A host can design their template feedback forms and save them to a database. The host should then be able to re-use a template feedback form in the future without having to remake

it. The host can also choose to delete a template feedback form they have created, which will remove it from the database.

8. **C:** Feedback is only shown to the host until at least 10% feedback forms have been submitted by attendees so that the feedback is representative of the general sentiment.

D: No general feedback should be displayed to the host, instead of displaying a warning that not enough data has been collected to show feedback at present until a minimum of 10% of feedback forms have been submitted by attendees so that the feedback is representative of the general sentiment when the general feedback is displayed.

Attendees

1. **C:** Attendees can join a room by entering a code given to them by the host which is unique to the room.

D: Attendees join a room using an alphabetic unique code which is assigned to a host's event by pressing a button saying 'Join room', inputting the event code, and pressing submit.

2. **C:** The attendee completes the feedback form chosen by the host.

D: The attendee completes the feedback form which is then used to provide live feedback to the host.

3. **C:** There will be an option on the feedback form allowing the feedback supplied by this user to be anonymous.

D: A checkbox on each feedback form will allow an attendee to submit their feedback anonymously. If they do not opt to be anonymous, the attendee will be prompted to enter their name.

4. **C:** Optionally, the attendee can give a rating out of 5 stars on the feedback form for a question.

D: If no star rating is selected, then no rating is delivered. However, if the attendee selects a star rating for their review, this is submitted along with their feedback.

5. **C:** An attendee can type feedback in a text box to add to their feedback form.

D: The user provides feedback in a textbox, which is then analysed with sentiment analysis to provide a general mood of the audience for the host.

6. **C:** The percentage of responses that have been submitted is displayed to the host so that the host can have an understanding of how representative the feedback they are seeing is. **D:** Whenever a user submits feedback, the percentage of attendees who have submitted feedback out of all attendees is calculated and displayed to the host.

4.2 Non-Functional Requirements

1. **C:** When an attendee submits feedback the response should be submitted in under 5 seconds.

D: When feedback is submitted it should be passed to and stored in the database in under 5 seconds.

2. **C:** The web page should be accessible from multiple devices including PCs and smartphones.

D: The web page should be designed in such a way that it is compatible with both PCs and smartphones and can be accessed from both. [3]

3. **C:** The web page must be displayed correctly on any devices used to access it so that no images or text boxes are floating off the edge of the screen or otherwise inaccessible.

D: The website's HTML and CSS should be written such that whether the site is being accessed by smartphone or by PC all resources are available as intended. For example, all buttons are of an appropriate size, data entry forms can be selected and data inputted and all images or other graphics are displayed to the user.

4. **C:** The submitted feedback must be stored efficiently and must be accessible to the feedback analysis system in under 5 seconds.

D: If feedback is needed for analysis it should be retrieved from the storage system and passed to the relevant process in under 5 seconds to prevent delay to the user.

5 Team Organisation

We have planned to meet at least once a week and more often if necessary to discuss our progress and plan what to do in the following week. We have decided to use a plan-driven waterfall model, completing each stage before moving on to the next one. We chose to do this because we do not expect the requirements to change throughout the development of the project so is it unlikely that we will have to accommodate changes in the requirements of the project. Furthermore, the waterfall model allows us to carefully plan how we will achieve each of the project requirements. We can test each component of the system as it is added and ensure that all components of the system are completed on time.

The allocations below in Table 2 were made based on individuals' strengths and interests. The responsibilities below will be that person's main focus throughout development, however, responsibilities will not be strictly confined to those assigned and team members may contribute to other tasks if necessary.

Name	Role	Justification	Responsibilities
Rebecca French	Project Manager	Has coordinated other projects and understands and uses management templates to ensure a well working team	Project Management, monitoring progress, and ensuring deadlines are kept to
Luca Orita	Business Analyst	Has experience from different competitions, creating and designing business plans for other startups, using information from relevant modules of WBS	Understand the business goals of the project and understanding the needs of the stakeholder
Jack McCullough	Software Engineer	Has previous experience with creating and using databases	Backend development, creating the database, and ensuring that testing is completed on both
David Yang Zhang	Software Engineer	Has done small Machine Learning projects in the past, including having done research when applying for ML jobs	Backend developing, sentiment analysis machine learning, and ensuring testing is completed
Maria Bracegirdle	Software Engineer	Has experience in building user interfaces from previous projects	Frontend developing and ensuring testing is completed

Table 2: Table showing team role allocations and individual responsibilities

References

- [1] Warwick. Software requirements. <https://warwick.ac.uk/fac/sci/dcs/teaching/material/cs261/project>. Accessed 26 January 2021.
- [2] Saif M. Mohammad. Sentiment Analysis: Detecting Valence, Emotions, and Other Affectual States from Text. <https://www.saifmohammad.com/WebDocs/emotion-survey.pdf>. Accessed 4 February 2021.
- [3] Interaction Design Foundation. User Interface Design. [https://www.interaction-design.org/literature/topics/ui-design#:~:text=User%20interface%20\(UI\)%20design%20is,e.g.%2C%20voice%2Dcontrolled%20interfaces](https://www.interaction-design.org/literature/topics/ui-design#:~:text=User%20interface%20(UI)%20design%20is,e.g.%2C%20voice%2Dcontrolled%20interfaces). Accessed 4 February 2021.