PA-203-ANEMOIA-3
TECHNICAL REPORT

Version 1.1.1

Start Date: **26/05/2023**

Authors:

- **Crystal Yvonne Wolfe, s320684**
- **Mark Huang, s3779569**
- **Nicholas Kabissios, s3840379**
- **Bao Ngan Nguyen, s3804803**
- **Jeremy Gilbert Christopher, s3860988**

DOCUMENT CONTROL

Version #	Implemented By	Implementation Date	Reviewed By	Approval Date	Reason
1.0.0	Nicholas, Jeremy, Mark	10th June 2023	Crystal, Bao Ngan	11th June 2023	Written sections: Introduction, Architecture, Implementation, Testing specification
1.1.0	Mark, Crystal, Bao Ngan	11th June 2023	Nicholas, Jeremy		Written sections: Implementation, Test Results, Cyber security, impacts on stakeholders
1.1.1	Bao Ngan, Nicholas, Crystal, Mark, Jeremy	12th June 2023	[Supervisor]		Proofread and edited the report.

.

TABLE OF CONTENTS

Contents

1 EXECUTIVE SUMMARY....4

2 INTRODUCTION....4

3 REQUIREMENTS....4

FUNCTIONAL REQUIREMENTS SPECIFICATION 4

NON-FUNCTIONAL REQUIREMENTS SPECIFICATION 5

4 ARCHITECTURE 5

5 TECHNICAL FRAMEWORK....5

6 IMPLEMENTATION 6

7 DEPLOYMENT INSTRUCTIONS....7

8 TEST SPECIFICATIONS 8

9 TESTING RESULTS IN 11

10 CYBER SECURITY 11

11 IMPACT ON STAKEHOLDERS 11

1 EXECUTIVE SUMMARY

<Write here the executive summary of the project (half page)>

2 INTRODUCTION

The Red Book intends to be a social network for microblogging and communication between users that allows reviewing and evaluating employers' behaviour and work environment. Specifically, the project's primary goal is to create a platform where employees who do not have an outlet to talk about their work and experiences can do so. In making the project, the application will empower employees to shape the specific requirements and features of the project.

The project's primary stakeholder is Bej'a Christmas, the founder and CEO of Anemoia Studios. Bej'a and Anemoia Studios want to create a space that facilitates connecting people, uniting creatives, and artists, sharing opportunities, and building community and harmony. Furthermore, specific goals involve:

- Empowering women
- Respect and career excellence for artists
- Unionization of artists and creatives
- Support for those who face discrimination in the industry
- Explore new opportunities
- Create awareness for new artists and creatives about the industry

3 REQUIREMENTS

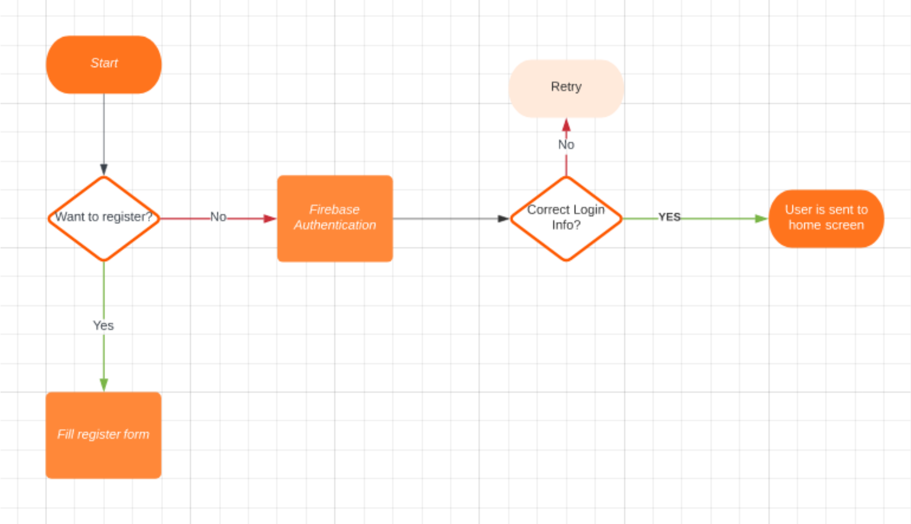
FUNCTIONAL REQUIREMENTS SPECIFICATION

Req. ID	**Description**	**Priority**
1	The system must allow users to start a live symposium	H
2	The system must allow users to send messages to the symposium	H
3	The system must allow users to set reminder notifications for symposiums	M
4	The system must allow a host to select topics for the symposium they are creating.	M
5	The system must allow a host to select speakers for the symposium they are creating.	M
6	The system must allow the user to view their upcoming and past symposiums	M
7	The system must allow a user to view details about symposiums	L
8	The system must allow users to chat with other users in a symposium	M
9	The system must allow users to sign up if they do not have an account	H
10	The system must allow users to log in if they have an existing account	H

NON-FUNCTIONAL REQUIREMENTS SPECIFICATION

Usability: the app needs to follow a pattern design so it can be intuitive to use.
Performance: The app should have any noticeable freeze.

4 ARCHITECTURE



The above diagram indicates the first process that the application runs through. The approach is to register/log in.

The essence of The Red Book mobile application is that the front end, being what the user sees, is created using React-Native. The front end sends and fetches data from the back end. The back end is entirely processed through Firebase, including the project's database, firestore. The Firestore setup in Firebase is the primary process for the back end, where the database is called, and data is sent over to the front end to be displayed. Furthermore, user account authentication is also done through Firebase and the chat feature accessed through symposiums. Firebase facilitates all these features in the mobile app through pre-setup functions that are easy to implement, requiring some configuration relative to the application being worked on.

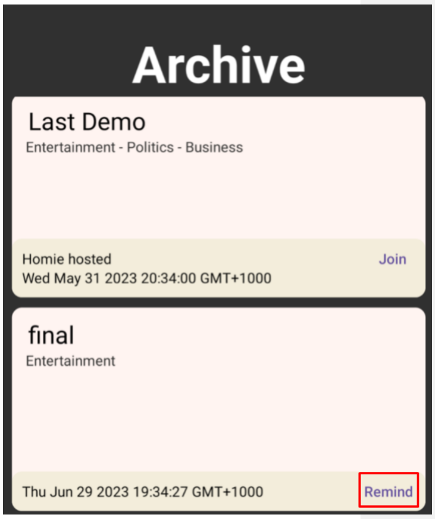
5 TECHNICAL FRAMEWORK

Slack – Used to communicate with the client and between group members
Zoom and Teams – Held meetings with the client (zoom) and the group (teams).
Trello – Used to outline tasks for sprints and assign tasks to group members. The Trello board shows the current state of development for the project at the time.
GitHub – Used for version control when developing, allows branches to facilitate group members working in parrel.
Gitmoji – Used to style commit messages in GitHub to identify what changes have been made more accessible.
JavaScript – The primary programming language used
React Native - The JavaScript framework used to build the front-end
Firebase – The backend system used for storing data and user authentication

Node Package Manager (NPM) - Used to install and manage the JavaScript packages
Expo – Used to test the application and allows for live updating and testing changes
Figma – Used to design wireframes to be followed when developing and designing the application

6 IMPLEMENTATION

1. Modal View
   There are several methods to create a modal view in React Native. We experimented with h couple of packages: “React-native-paper”, “React-native-core”, and “React-native-navigation” Each of them has its drawbacks due to its inability to cover both Android and iOS platforms, but since we used “React-native-navigation” for the navigation part of the app, we ended up using that package to present a modal in iOS or a bottom dialogue in Android.
2. Messaging
   The Messaging feature in the red book app enables real-time communication within symposium channels, promoting collaboration and knowledge sharing. It uses the Firestore Storage from Firebase to collect the input data from other users and to save it so other Users can exchange messages and participate in discussions. The system ensures message delivery. With the current implementation, other users can see who has typed in the existing symposium by their profile picture or icon. There is a feature where they can vote on whether to disagree or agree with the messages that have been delivered. They cannot choose two options, so only one can be picked, but you can change it whenever you want if the message is still at the symposium.
3. In-app notification
   Push notifications are handled locally through the user’s device. Expo contains packages that facilitate the red book’s implementation of local notifications. The user sets these notifications by pressing the ‘remind’ text in the archive page for the upcoming symposium (see figure). When the device’s date and time reaches the date and time of the symposium, the user is notified that the symposium has started. However, there are limitations to the current implementation, as these notifications are not shared across devices, and the notifications only remind the user that the symposium has started and doesn’t allow for the app to be opened through interacting with the symposium.



Archive

Last Demo
Entertainment - Politics - Business
Homie hosted
Wed May 31 2023 20:34:00 GMT+1000

final
Entertainment
Thu Jun 29 2023 19:34:27 GMT+1000

4. Authentication
   Authentication is handled using Firebase’s authentication system, including password resets and additional user data (profile), which is stored within Firestore as a document with the user id; this contains information like names and images and can

be linked to speakers for symposiums and message owners within the chatrooms. The application can handle logging in and signing up with partial profile data.

5. Local Storage

   We opted for SQLite as our go-to database to enable offline use of the app because of its popularity and ease of use. At first, we attempted to use native “SQLite” packages, but there were some compatibility issues with the other packages we were using. We couldn’t set it up to work with the Expo framework. Fortunately, Expo has its library for local storage. We were able to convert our code for the other tools to Expo rather quickly because they are still SQL under the hood, which we have already been familiar with.

7 DEPLOYMENT INSTRUCTIONS

The application runs on two sides, the front end and the back end. The backend is designed to unitise serverless functionality using Google’s Firebase service.

Creating a Firebase application is necessary for the application to function. After completing the application as a Web App, download the JavaScript file and copy over the `firebaseConfig` object. Then deploy the application using Expo’s deployment tools.

Commented [BNN1]: Need to clarify that this is for security purpose. When we need to set up different environment, Firebase console for production, no backend setup is required for testing @Nicholas Kabissios

Commented [BNN2R1]:

8 TEST SPECIFICATIONS

Test Case ID	Req. covered	Test Objective	Preconditions	Steps	Test data	Expected result
0	9	Account Creation	Has the app opened? Ensuring the account does not exist.	1. Press “Register” as a secondary button 2. Type your email, password and name 3. Press “Register.”	Email: s3779569@student.rmit.edu.au Password: TestPassword123* Name: [any text]	Home screen to be shown with name displayed
1	6, 7	View list of available symposiums	Log into an account.	1. Click on the “Archive” button 2. Is navigated to the Archive page.	N/A	A list of symposiums is displayed. Each symposium card view contains a title, topics, speakers’ names, scheduled dates, and a Call-to-Action button: Remind or Join.
2	3, 10	Symposium reminder	Account and upcoming symposium.	1. Login User 2. Go to Archive 3. Press “Remind.”	ID: 1xNrmnbvDMM4l7tJfHvh	Receive a notification when it starts
3	6, 7	Join a live symposium [1]	Log into an account. A symposium is created.	1. Navigate to the “Archive” page 2. Click on the “Join” button of a symposium card view	N/A	A Pill-shaped view floats to the top: the view contains the title, the first three speakers and their avatars, and a clickable “End” button.

4	6, 7	Join a live symposium [2]	Log into an account. A symposium is created.	1. Navigate to the “Archive” page. 2. Click on any symposium card view. 3. a modal pop-up. 4. Press “Join.”	N/A	A Pill-shaped view floats to the top: the view contains the title, the first three speakers and their avatars, and a clickable “End” button.
5	N/A	Leave a live symposium	Log into an account. Already joined the d symposium.	1. On the floating symposium, press “End.”	N/A	That floating view should be dismissed.
6	2, 7, 8	Message Synchronisation	Log into an account. Symposium created.	1. Open the Archive page 2. Tap on the chat icon 3. Send Message A using User A 4. Send Message B using User B	Message A: A Message B: B	MA shows on User B’s screen and in reverse
7	8	React to message. The message list should be updated based on the vote, as the app will order messages based on the number of votes.	Log into an account A symposium is open A message is created for that symposium	1. On the detail modal view of a symposium, tap on the “Chat” button – button with a bubble chat icon, next to the “Join” button 2. Navigate to the Chat page 3. Find an existing message 4. Tap on the like or dislike button.	N/A	You should see several votes (either for like or dislike) increase by 1.
8	8	Can react only once per each message	Log into an account A symposium is opened A message is created for that symposium	1. [Follow the same steps as test 7] 2. switch to click on another button than the one you previously selected for that message	N/A	You should see the number of your previously selected button decrease by one and the number of your current selection increase by 1.

9	1, 4, 5, 6, 7	Create Symposium [happy path]	Log into an account.	1. Press “Create.” 2. Enter the subject, 3. select speakers 4. [Optional] Select topic, date, time 5. Press “Submit.”	Subject: A Topic: IT Speaker: Speaker 1 Time: 2038-01-19T03:14+10:00	Symposium shows up in Archive.
10	1	Create symposium [edge case]	Log into an account.	1. Press “Create.” 2. Press “Submit.”	N/A	The alert prompt shows up.
11	N/A	Logout	User account	1. Login User 2. Press Logout	N/A	Returned to the login screen

9 TESTING RESULTS

All the tests stated have passed successfully without any issues. These tests mainly test authentication, messaging and parts of the database (firebase).

Known bugs:

- The notch of the iPhone can hide a floating view. SafeView needs to be implemented.
- “End” button on the floating is hard to press. You may need to kill the app to dismiss the floating view.

10 CYBER SECURITY

- Authentication:
  - Login verification is handled on the Firebase server.
  - Messages and symposiums are assigned authors from authenticated users.
- Database rules:
  - Disallow read/write when the user is not authenticated.
  - Only allow authorised users and administrators to edit profile data.
  - Only allow message authors and administrators to delete and edit messages, including creation, to only the author.
- Cons:
  - Authentication: Emails are not yet being verified.
  - Authorization: Need to be able to differentiate administrative and casual users.

All information, including login credentials and messages, is sent using a secure channel to the Firebase server.

However, some security concerns should be addressed, being the lack of access control on Firestore; this must be solved server side to prevent rouge apps from running queries on data it should not be allowed to. This is on the list to be completed.

11 IMPACT ON STAKEHOLDERS

The impact the stakeholder and users that this application can create is that it would enable businesses and organisations to share information without any discrimination, allowing several people and countries to talk about companies or artists.

The actual stakeholders impacted would be Anemoia, as they are the ones that tasked us to create the application. In doing so, Bej’a and Zenobia will be the ones most impacted, being the Founder/CEO and Scrum Master, respectively, because whatever has been created needs to be to their expectation because their reputation may be affected by how well the application does when complete.

One of the primary users is artists, who would like an area to share their experience. We also enable the application to be on both iPhone and Android, allowing several people to use it.

Currently, the only stakeholder that will be impacted would be those that currently reside in the US as it is based in the US, but in the future, if the client wants it will become worldwide would then impact more stakeholders and users, but at this current moment that is not applied cause it currently only locally where Anemoia Studio is based.