D3 – DEMONSTRATION REPORT

HELPXPERT

AHMED DARKYE – 38557207 (GROUP LEAD)

RALPH SAM MENSAH – 38570009

CALEB APEDOH – 38557193

CONFIDENCE ALABI - 38600579

OVERVIEW

HelpXpert is an android application which is intended for the purpose of allowing the Staff of Lancaster university to make reports about environmental concerns to the Facility Department. The goal of this project is to contribute to SDG 4, "Better Education". At the time of writing this report, the HelpXpert application has been completed and ready for use. Two versions of the application have been developed. The Facility App and the Staff App. Each app has been implemented with functionalities which fulfill the important interests of both intended users. The application was developed with the purpose of achieving simplicity, and getting tasks accomplished. To attain this, the team decided to incorporate Shneiderman's 8 golden rules of user interface in the implementation process. First time users will have little to no difficulty in navigating and using the application because every action is direct and clearly comprehendible. The two versions of the application are connected to one another with the use of same backend resources. Both users are able to send text messages to one another with the use of a simple chat system. The facility department can then fill out a report and publish it to the database. The report information is then made available to staff according to unique credentials. The status of each report is also made available to the staff users. Members of the facility department are also able to view all reports. In the earlier stages of implementation, the team decided to split the application into two versions for various reasons. Two different users on the same application suggests an unnecessary large file size, and the team wanted to avoid this. Secondly, for security purposes, the users will have separate interfaces to avoid attempted account access of a different user type. The programmers of the application tried as much as possible to avoid bugs in the code during the implementation stage. A lot of exception handling was incorporated into the program to handle bugs. At the end of the implementation process, the team members are proud of every aspect of both versions of the application, but the outstanding features are the use push notification and the implementation of the chat system. An android phone of newer operating system, a good internet connection, and a large storage between 32 to 64 gigabytes will be required the enjoy the most pleasing results out of the proposed software.

USER INTERFACE AND EASE OF ACCESS (MANUAL)

In terms of user interface, the two versions of the application are introduced to their correct audience. Both applications serve a similar design, with only slight differences. All actions have been given direct and clear labels to help users navigate the applications with ease. Official colours of Lancaster University were used in designing HelpXpert. The Facility App has 11 different screens while the Staff App has 9 different screens. Below are descriptions of all the screens, as well as snapshots from the actual application.

FACILITY APP

- The Sign-In Screen:

This is the first screen which appears when the app is launched. Users must type in credentials to gain access to the system.

- The Sign-Up Screen

To access this screen, users must tap on "Create New Account" on the Sign-In screen. New users can create an account on this page to have access to the system.
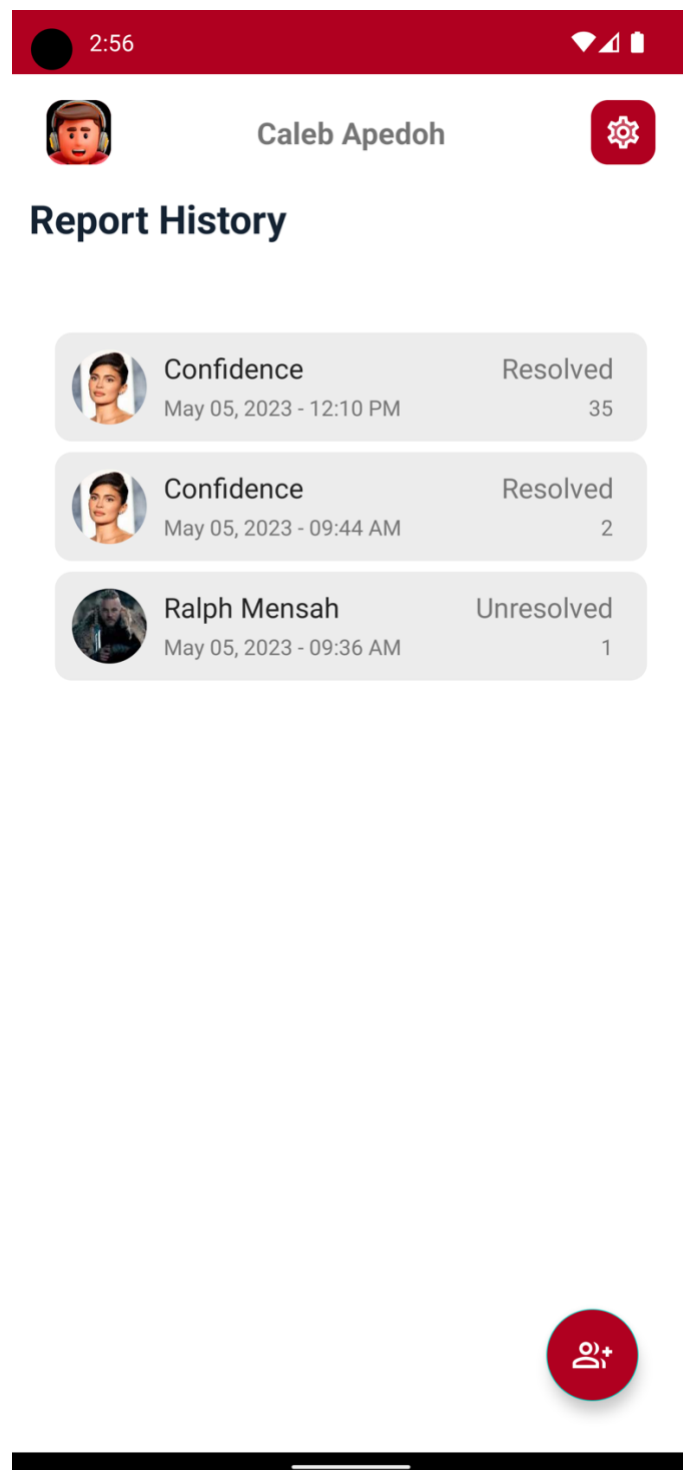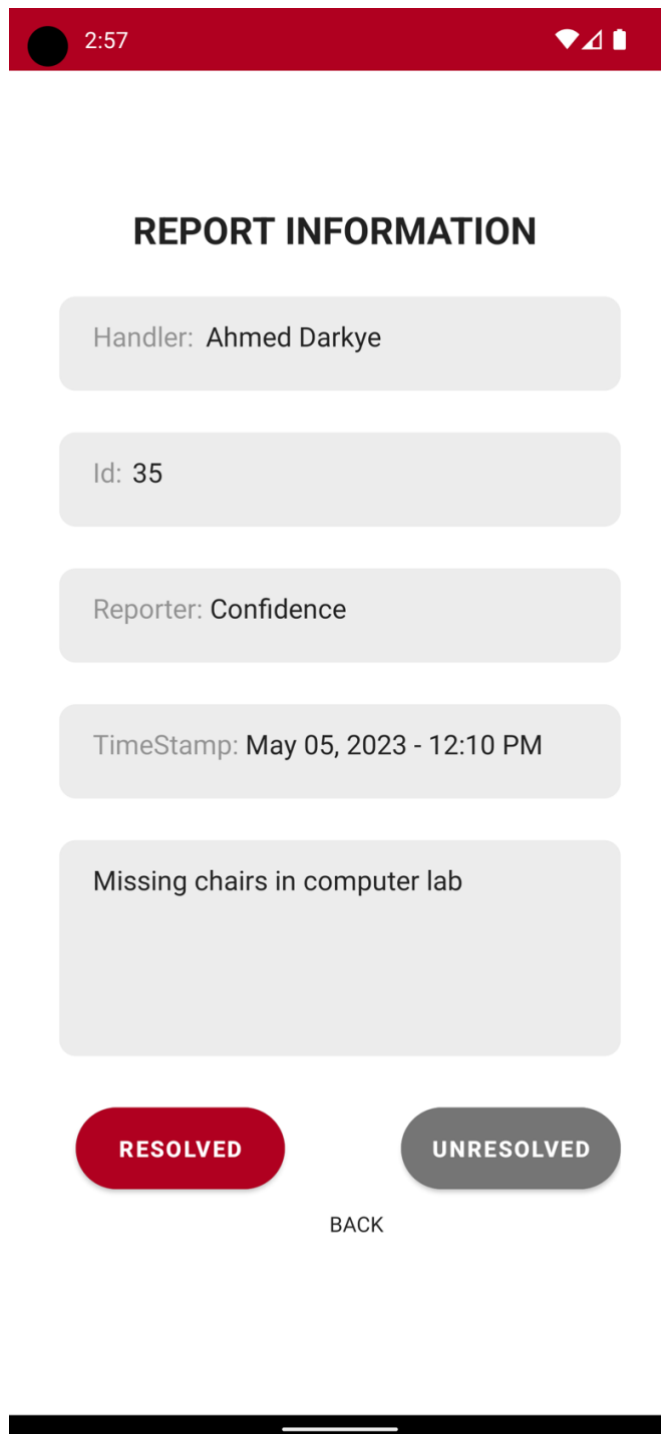
- The Forgot Password Screen

To access this screen, users must tap on "Forgot Password?" on the Sign-In screen. Users can change their forgotten passwords on this screen with their secret phrase.

- The Report History Screen

Users are automatically brought to this screen once signing in is successful. All reports are displayed on this screen with the necessary information. Users can tap on "Report History" to manually reload reports from database in the case of lagging in automatic reload.

- The Report Information Screen

Users can access this page by tapping on a report on the Report History screen. All information regarding a specific report is displayed on this screen. Users are also able to change the status of a report on this page.

- The Settings Screen

Users can access this screen by tapping on "⚙" symbol on the top right corner of the Report History screen. Users can change email and password, toggle notification on or off, view Facility Department Users screen, and Logout on this screen. Users can also move to the previous screen by tapping on "Settings" on the top left corner of the screen.

- The Software Users (Facility) Screen
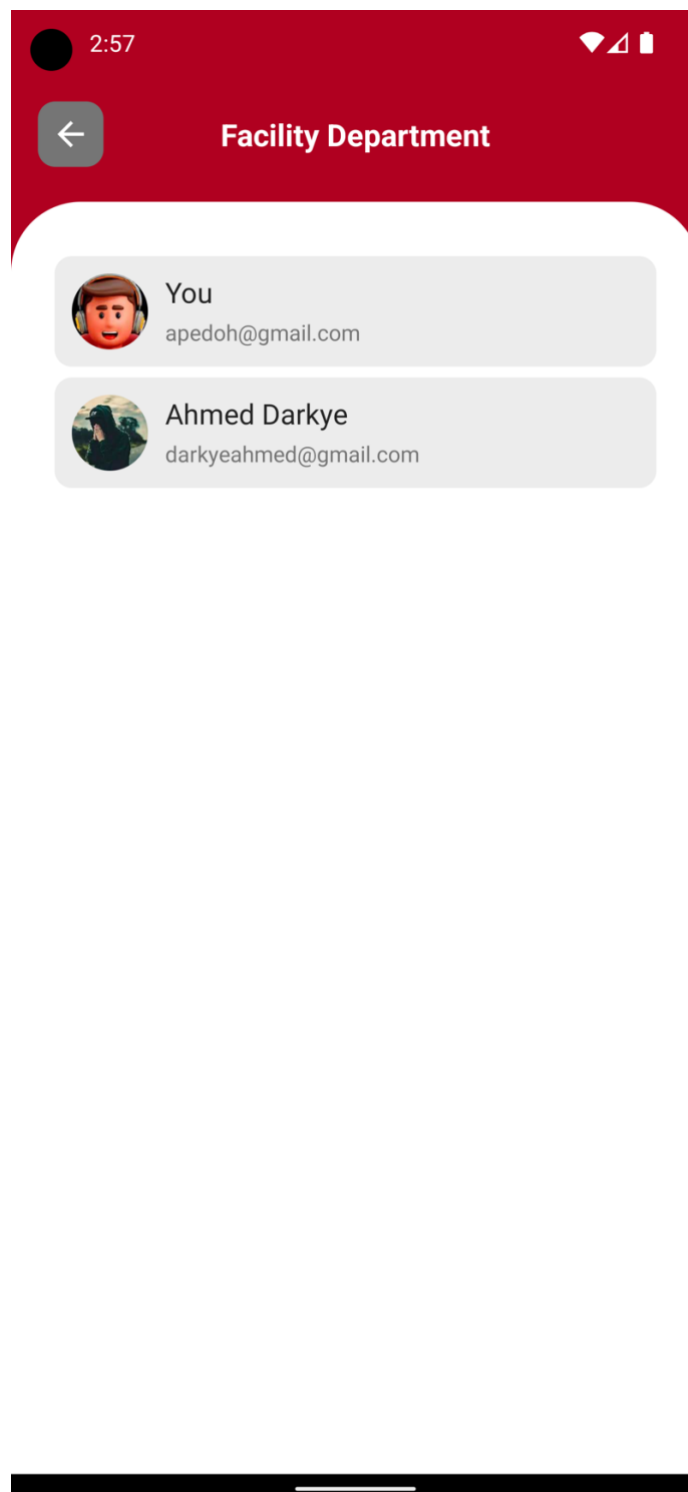
Users can access this screen by tapping on "Software Users (Facility)" on the Settings screen. All users of the facility department are displayed on this screen, including current user.

- The Conversations Screen

Users can access this screen by tapping on the red floating button on the bottom right corner of the Report History screen. All conversations or chats are displayed on this screen.
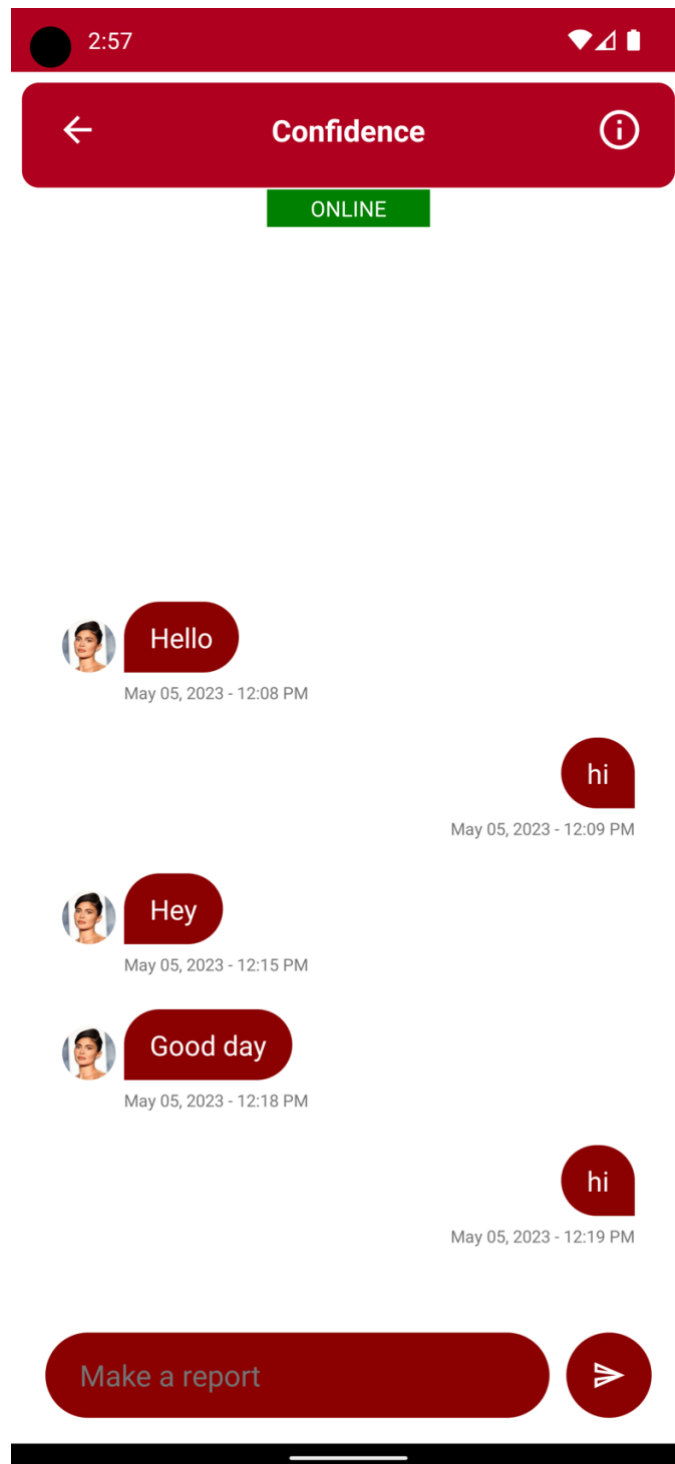
- The Chat Message Screen

2:57	

Confidence

ONLINE

Hello
May 05, 2023 - 12:08 PM

hi
May 05, 2023 - 12:09 PM

Hey
May 05, 2023 - 12:15 PM

Good day
May 05, 2023 - 12:18 PM

hi
May 05, 2023 - 12:19 PM

Make a report

Users can tap on each conversation on the Conversations Screen to access this screen. This screen can also be accessed by tapping on a user in the New Conversations Screen. Chat messages are displayed on this page, as well as users being able to send new ones. "Online" indicates the user is available on the application.

- The Report Form Screen
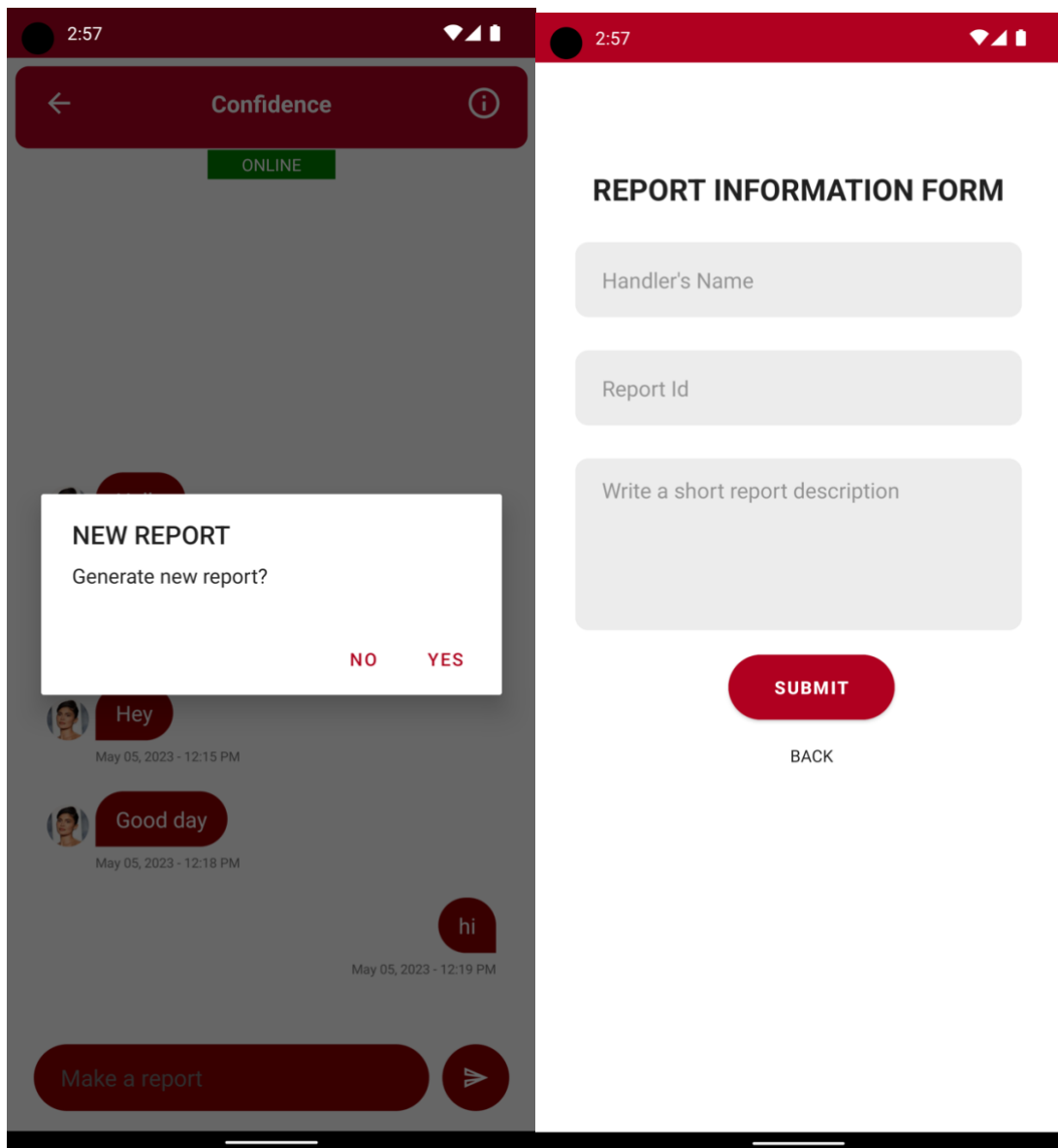
Users can access this screen by tapping on "ⓘ" on the top right corner of the Chat Message Screen. Users need to confirm an alert message before transferred to this screen. Users fill out a report form and submit on this screen. These reports are displayed on the Report History screen.

- The New Conversations Screen for Staff Users
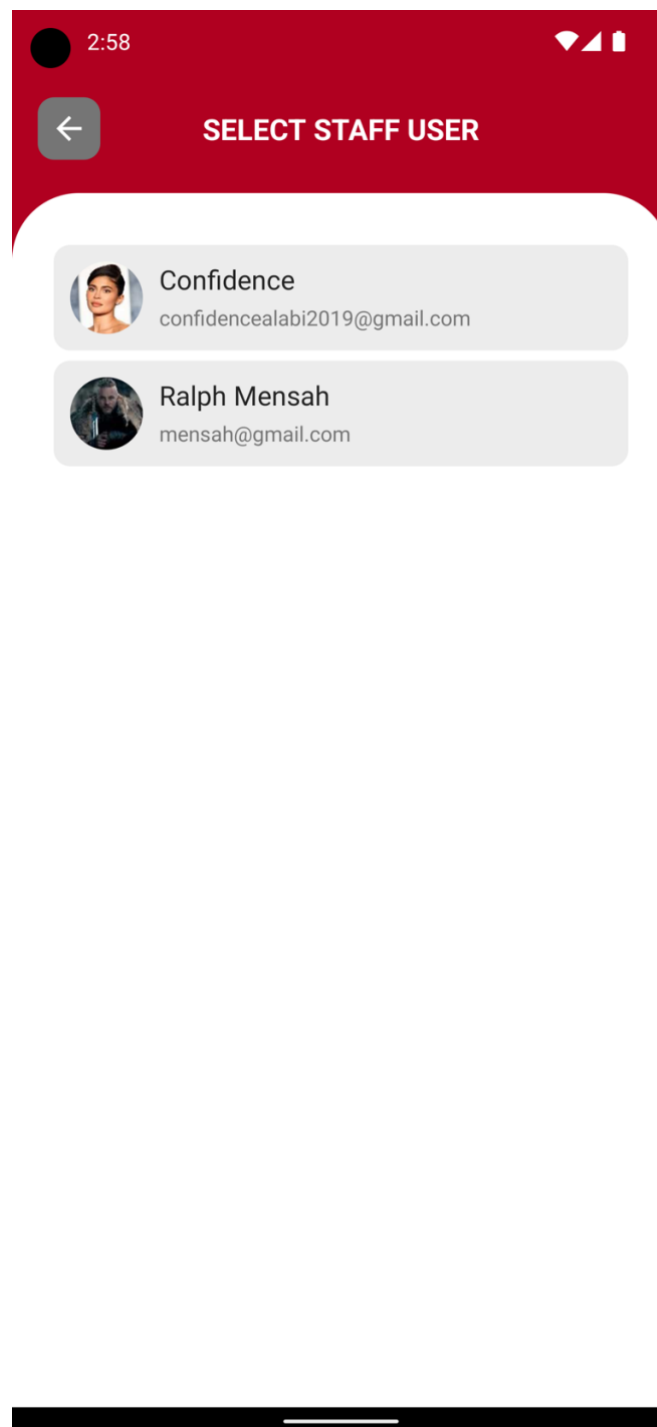
Users can access this screen by tapping on the floating red button with "+" at the bottom right corner of the Conversations Screen. All staff users on the Staff App are displayed on this page. Users can start a new chat by tapping on a user on this page.

STAFF APP
- The Sign-In Screen

This is the first screen which appears when the app is launched. Users must type in credentials to gain access to the system.

- The Sign-Up Screen

To access this screen, users must tap on "Create New Account" on the Sign-In screen. New users can create an account on this page to have access to the system.

- The Forgot Password Screen

To access this screen, users must tap on "Forgot Password?" on the Sign-In screen. Users can change their forgotten passwords on this screen with their secret phrase.

- The Activity Screen (Personal Report History)
-

Users are automatically brought to this screen once signing in is successful. All personal reports are displayed on this screen with the necessary information. Users can tap on "Report History" to manually reload reports from database in the case of lagging in automatic reload.

- The Report Information Screen

Users can access this page by tapping on a report on the Report History screen. All information regarding a specific report is displayed on this screen.

- The Settings Screen

Users can access this screen by tapping on "⚙" symbol on the top right corner of the Report History screen. Users can change email and password, Logout, and toggle notification on or off on this screen. Users can also move to the previous screen by tapping on "Settings" on the top left corner of the screen.

- The Conversations Screen

Users can access this screen by tapping on Chats "💬" on the bottom right corner of the Report History screen. All conversations or chats are displayed on this screen.
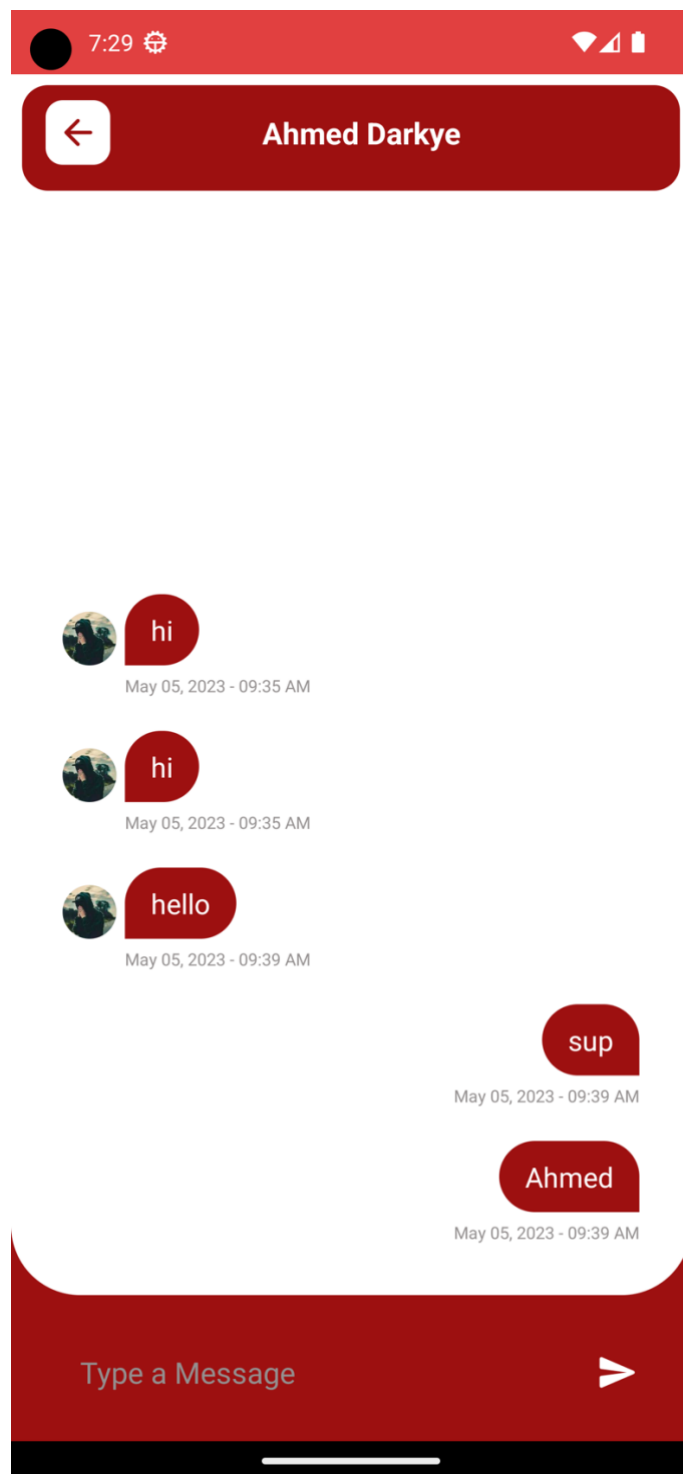
- The Chat Message Screen

Users can tap on each conversation on the Conversations Screen to access this screen. This screen can also be accessed by tapping on a user in the New Conversations Screen. Chat messages are displayed on this page, as well as users being able to send new ones.

- The New Conversations Screen for Facility Department Users
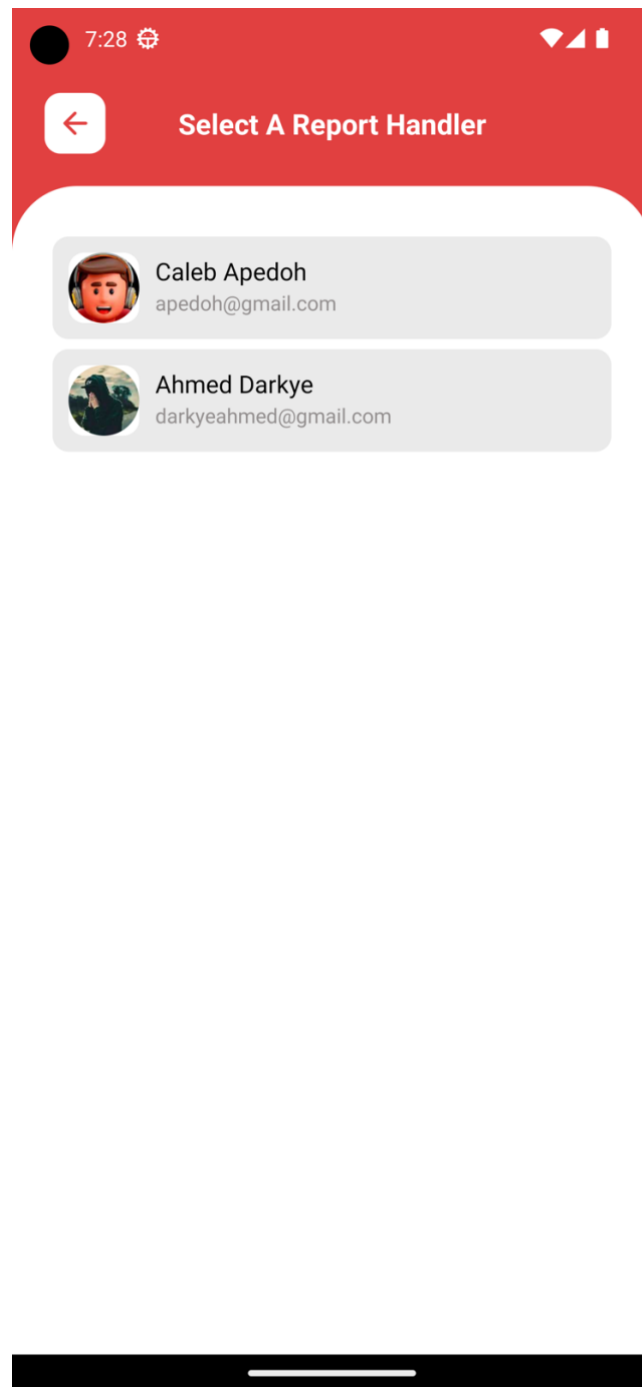
Users can access this screen by tapping on the floating red button with "+" at the bottom center of the Report History Screen. All facility department users on the Facility App are displayed on this page. Users can start a new chat by tapping on a user on this page.

Informative feedbacks are provided when in need. The other Shneiderman golden rules have been implemented across these applications. The interface boasts a very beautiful, yet simple design to allow users to work with efficiency.

DEMONSTRATION OF ALL MAJOR FUNCTIONS

At the completion of the applications, all major functions mentioned in the design report have been implemented in both applications. The fundamental function of HelpXpert is to allow Staff to track reports and send reports to the Facility Department. The Facility Department can in turn fill out a formal report form and publish them for the appropriate users to view. HelpXpert also allows the Facility Department to update the status of each report. The communication between the two types of users is facilitated by the implementation of a simple chat system between the two users. The log in and log out functionalities have also been incorporated into the two versions of HelpXpert. All the implemented major functions stay on track with reference to what has been described in the design report. Furthermore, the team added some minor functions to support the major ones to make the application more efficient.

MULTI-USER COORDINATION / CONCURRENCY MECHANISM

HelpXpert is equipped to handle multiple users on the system via Firebase Firestore. It also uses resources to manage the activities of multiple active users on the system. First and foremost, the team decided to develop two applications for various reasons. One being that the team found it inconvenient to deploy one large application containing a whole range of functionalities which is not entirely used. Secondly, to prevent the different types of users from attempting to access an unauthorized section of application. And finally, the most relevant reason for this section of the document, which is to divide user load on a single table in the database. Furthermore, the team decided to use an android utility called PreferenceManager as a concurrency mechanism. PreferenceManager can be described as a form of cache which stores data in key value pairs. To prevent multiple users from continuously accessing reoccurring data from the Firestore database, the team implemented this utility to store any information retrieved after every query from the database within the application for future usage. This prevents the need to always query commands to retrieve same data multiple times from Firestore. This eases the load on the database and allows every instance of the application to function independently. This allows the applications to only query renewed data such as messages from the database. This utility also allows the application to retrieve data faster than database queries. To prevent consumption of large cache memory, PreferenceManager has been configured to replace some data and clear all data after logout.

BUGS IDENTIFIED AND FIXES APPLIED

The team implemented several error and exception handling within the applications. Unit testing was conducted during the development of various parts of the application to ensure methods were delivering the desired outputs. After a few components have been developed, the developers conducted integration testing between separate activities to ensure that they are working together as intended. The team used both manual and automatic testing depending on what was being tested. ChatGPT was used as a tester to identify vulnerabilities and suggest methods of handling errors. A bug that was identified during the implementation phase were null pointer errors. The application received multiple fatal crashes because the application attempted to use the reference of an object which has been set to null. Queries are stored within PreferenceManager and then later used within methods. This error kept appearing because, in some cases the methods read no value from the PreferenceManager, which led to a null pointer error and fatal crash. To fix this, the developers identified that wrong attribute names were used in the queries which resulted in empty queries and resulted in feeding PreferenceManager null values. To developers checked all attribute names and corrected them. This resolved the null pointer errors. Another bug discovered in earlier stages was that the application always crashes after the creation of a new activity. The reason for this was because Android Studio was not giving the appropriate names for activities the Manifest file. To prevent this error from occurring, the developers always made the names in the Manifest file matched with the names on the activities created. The last and most troublesome bug detected has to do with push notifications. After the code for push notification was implemented, it was working perfectly. The team wanted to allow users to control this functionality. A switch button was implemented in the settings activity to control this. However, every time the settings was closed, the button reverted to its original state (off) which leaves notifications off even after enabling and leaving the Settings screen. The error was a logical error and therefore could not be read in the console at runtime. After research, the team discovered that an activity instance is destroyed together with its configurations whenever the user closes it. To correct this, the team used PreferenceManager to store the state of the switch even after closing. The value stored was then used to enable or disable notifications. Notifications functioned well until the simulator was changed for one of the applications. It suddenly stopped receiving notifications. After various testings, the team could not find any errors within the code. The bug was later discovered to be within the operating system of the simulator. By default, notifications are turned on in settings on the simulator. However, it was turned off and greyed out in Android 13. The team researched and found out that, it is a common bug in Android 13. The solution was to downgrade to Android 12, which solved the bug. These are the major bugs discovered in the development of the application. After the implementation, the team conducted a final software testing using the white box testing technique. The team did not face many bugs during the development of applications. Below are the test cases used for the complete software testing.

Test Cases for Facility Department App:

1. Verify that the user can create an account successfully by providing a valid email address, password, profile image and secret phrase.
2. Verify that the user can reset the password by providing the correct email and secret phrase.
3. Verify that after signing in or signing up, the user is redirected to the report history page.

4. Verify that all reports submitted by the user are displayed in the report history page along with reporter's name, profile image, report status, report id, and date of submission.
5. Verify that the user can access the report information screen by tapping on a specific report.
6. Verify that the report information screen displays all relevant details about a report and allows the user to change the status of the report.
7. Verify that the user can access the settings screen and can change email and password, toggle notification on or off, view Facility Department Users, and Logout.
8. Verify that the user can access the Conversations screen and see all present conversations.
9. Verify that the user can start a new conversation.
10. Verify that the user can send and receive messages in the chat message screen.
11. Verify users are able to receive notifications.

Test Cases for Staff:

1. Verify that the user can create an account successfully by providing a valid email address, password, profile image and secret phrase.
2. Verify that the user can reset the password by providing the correct email and secret phrase.
3. Verify that after signing in or signing up, the user is redirected to the report history page.
4. Verify that the user can access the report history page and only sees reports made by the current user.
5. Verify that the user can click on each report to see detailed information regarding the report, including the status.
6. Verify that the user can access the settings screen and can change email and password, toggle notification on or off, and Logout.
7. Verify that the user can start new conversations or reports .
8. Verify that the user can access existing conversations by tapping on a conversation in the conversation screen.
9. Verify that the user can send and receive messages.
10. Verify that the user can receive notifications.

OVERALL FUNCTIONALITY
In terms of overall functionality, a scenario would be used for each user.

FACILITY DEPARTMENT
A person from this department creates an account with a profile image, email, password, and secret phrase. The secret phrase is required to reset a forgotten password. This style of password restoration was inspired by the security behind cryptocurrency wallets. In the case of a forgotten password, the user needs the enter the correct secret phrase and email to change password. If signing in or signing up is successful, the user is taken to the report history page. On this page, reports submitted are displayed with information containing reporter's name, profile image, report status, report id, and date of submission. On this page the user can visit several screens: report information, settings, and conversations. In report information, a more detailed information is displayed about a report as well as the options to change the status of

a particular report. This is accessed by tapping the desired report on the report history screen. In settings, the user can change email and password, toggle notification on or off, view Facility Department Users, and Logout. In the Conversations screen, all present conversations are present here. From here, the user can access two screens. That is, tapping on a conversation and directed to a chat message screen or starting a new conversation screen. In the chat message, users and send a read message. Users can also generate and fill out report forms from this screen. The information is then displayed in the report history.

STAFF
The process of getting to the report history is the same as the Facility Department. However, the report history for Staff only contains reports for only that user. Users on this app do not see reports made by other users. This is done checking whether the reporter's id matches with the current user's id from Firestore. Users can click on each report to see detailed information regarding the report, including the status. From this page, users can visit settings and change email and password, toggle notification on or off, and Logout. Users can also start new conversations or reports by clicking on "+" and selecting a handler to make a report to. On the bottom right of the report history page, users can see their existing conversations and continue them.