4TH YEAR PROJECT TECHNICAL REPORT

Auto Messenger

BSc (Hons) in Computing

Specialization: Gaming and Multimedia

Stephen Roberts
X13114760@student.ncirl.ie

Declaration Cover Sheet for Project Submission

SECTION 1 *Student to complete*

Name:
Student ID:
Supervisor:

SECTION 2 Confirmation of Authorship

The acceptance of your work is subject to your signature on the following declaration:

I confirm that I have read the College statement on plagiarism (summarised overleaf and printed in full in the Student Handbook) and that the work I have submitted for assessment is entirely my own work.

Signature:____ Date:____________

NB. If it is suspected that your assignment contains the work of others falsely represented as your own, it will be referred to the College's Disciplinary Committee. Should the Committee be satisfied that plagiarism has occurred this is likely to lead to your failing the module and possibly to your being suspended or expelled from college.

Table of Contents

Auto Messenger Technical Report

Student no. X13114760

1. Introduction

1.1 Background

The idea for this app came to me when I was reading an article about a young girl who had come up with the idea for nail polish that would change in colour if the person wearing it had been spiked. The article got me thinking, how can we use people's mobile phones to make them in some way safer when going about daily life, I specifically thought mobile phones because it's something people nearly always have on them where ever they go. Further I thought, it's got to be something that users don't consciously have to be using for it to function correctly, something they can have set before they go about their day and not have to worry about remembering while already out. The main reason and focus behind the idea of this app is based around the user's security and personal safety.

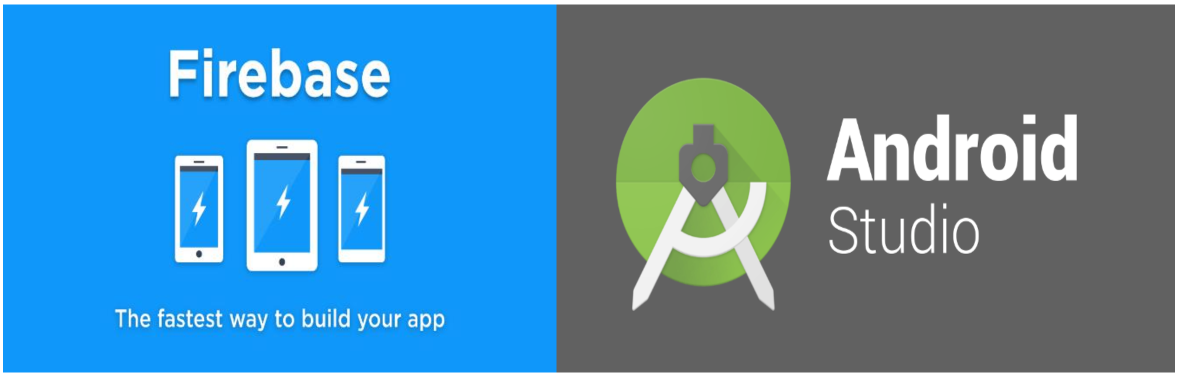
1.2 Aims

The aim of the project is to develop a mobile messenger application that utilizes an Auto Message feature where the user can allow for, and set a location to a specific WI-FI network or multiple networks. For example, a user's home WI-FI network can have a location set to "home." When the user's phone automatically connects to an activated WI-FI network the app will implement the Auto Message feature and send out a templated message, the message will be sent to chosen groups or individuals. The message will be based on the locations that have been set for the specific Wi-Fi networks, and what Wi-Fi network the phone has connected to ("work" "home" "college" "Johns house") the templated message may look like this ("user-name" has arrived at "set location" or "Paul" has arrived at "home"). As specified in the background, the main overall goal of this app is to achieve something that positively contributes towards the safety of the person using it.

1.3 Technologies

Android Studio is the official IDE for android platform development and is freely available under the apache license. It is designed specifically for Android development and I am using it to develop the app, as thus far the app is exclusive to android.

Google Firebase is a mobile platform for helping to develop app, and increase your user base. It's made up of features that can be mixed and matched to fit the needs of the user. Most of the features on Firebase are free to use. I will be using Firebase for app sign in options and authentication and to maintain a user database.



2. System

2.1 Requirements

2.1.1 Functional Requirements

2.1.1.1 Navigation:

- The application must make seamless transitions between activities or pages.
- Different pages must be defined clearly for the user.
- Features must be clearly visible to users.
- Available options on each page should be clearly visible to users.
- Instructions for Auto Message should be available within the pages.

2.1.1.2 Database Connection:

- Database must be constantly available for new user registration.
- Database must be updated upon new user registration.

2.1.1.3 Auto Message:

- The user shall be able to turn Auto Message on/off.
- The user may be able to select a group or individual to Auto Message.
- Auto Message must send a message if turned on and connected to a specified Wi-Fi network.
- Auto Message must include the user name and set location in a sent message.

2.1.1.4 Profile:

- The user shall be able to create a personalised username.
- The user shall be able to create a personalised password.
- The user may be able to use an alternative login i.e. login with Facebook or Google accounts.

2.1.2 User Requirements

- A user shall be required to have WI-FI enabled on their smartphone/device
- A new user shall be required to register or use a provided login method to gain access to the key features of the app.
- An existing user shall be able to login in using email and password or the alternate login method provided.
- Users shall be able to use the app from any smartphone/device that uses Android Operating System 2.3 Gingerbread or higher.
- A user may be able to set a location to be associated with a WI-FI network that the user has connected to.
- For the Auto Message feature to work a user must be able to connect to at least one WI-FI network.
- A user shall be able to turn the Auto Message feature on/off.
- A user may be able to select groups or individuals to receive an Auto Message.

2.1.3 Environmental Requirements

- A WI-FI network must be available.
- WI-FI network signal strength must be at a level where a device can make an authenticated connection.
- The connection must be held long enough for Auto Messages to be sent by the app.

2.1.4 Requirement 1: Start Application

2.1.4.1 Description & Priority

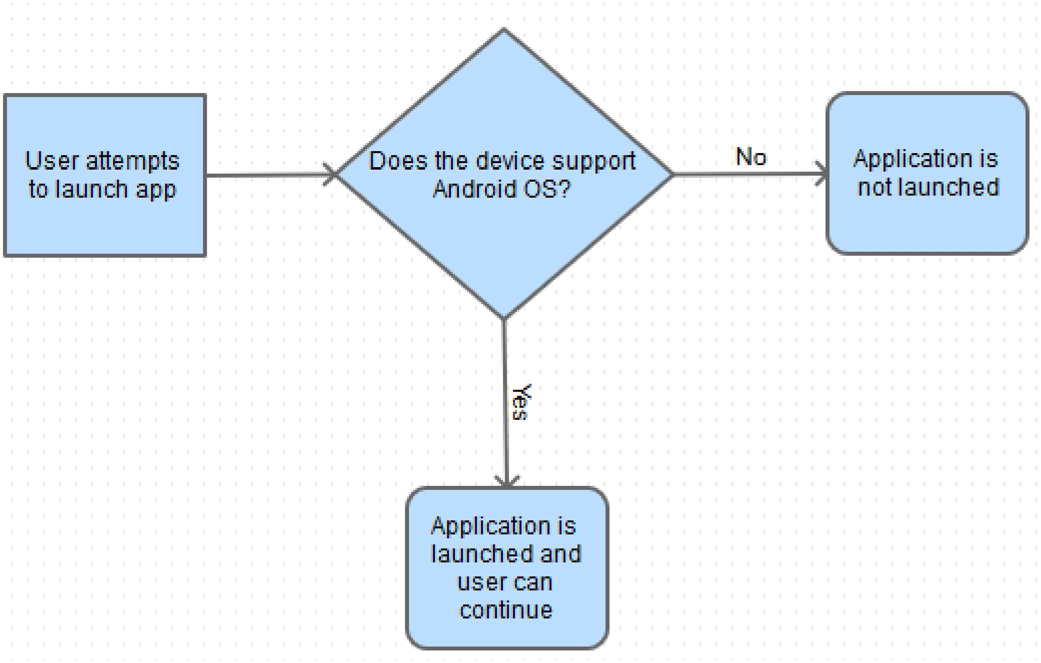
This use case is first in the priority list and describes the key functions that are offered to the user on starting up the application. If the user does not have the correct hardware they cannot launch the application.

2.1.4.2 Use Case

Scope

The scope of this use case is to describe how a user can launch the application and access the primary functions that are displayed to the user.

Use Case Diagram


User attempts to launch app
Does the device support Android OS?
No
Application is not launched
Yes
Application is launched and user can continue

Precondition

The application was downloaded and installed on a smart device that supports and runs Android OS.

Activation

This use case is activated when the user clicks on and starts up the application while in its idle state.

Main Flow

Actor Action: User clicks on the application launcher icon in an Android device menu.

System Response: Application is launched.

Alternate Flow

Actor Action: User clicks on the application launcher icon in an Android device menu.

System Response: The application does not launch due to a system crash or the OS is not compatible.

Termination

This flow is terminated when the application has booted on the device.

Post Condition

The user is brought to the main menu of the application where the system enters a state of waiting.

2.1.5 Requirement 2: Login or Register

2.1.5.1 Description & Priority

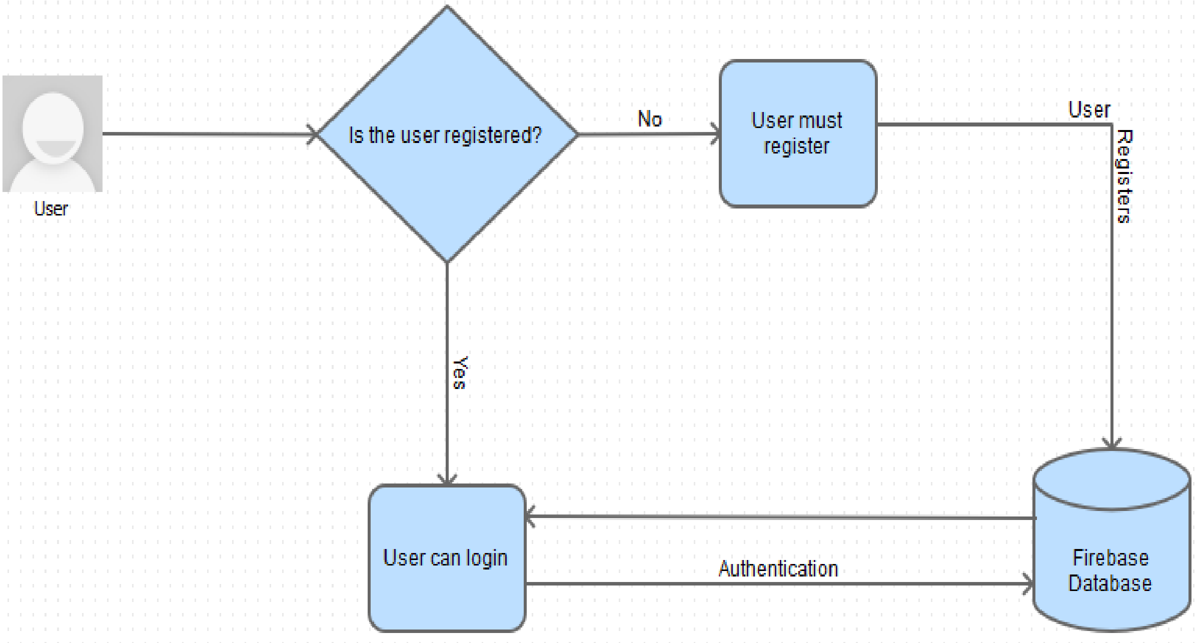
This use case is second in the priority list and describes the process of logging in and registering, once the application has booted up the user must be able to choose whether to login or register.

2.1.5.2 Use Case

Scope

The scope of this use case is to describe how the user can begin logging in or registering.

Use Case Diagram



Precondition

The application has been booted up and is running in a waiting state.

Activation

This use case will start when the user has launched the application.

Main Flow

Actor Action: User clicks on the login.

System Response: Login tasks (email, name, password) displayed for user.

Alternate Flow

Actor Action: The user is not registered and clicks on login.

System Response: The user is prompted to create an account.

Actor Action: The user does NOT click on login.

System Response: Login no register tasks will be launched.

Termination

This use case is terminated when the user logs in or registers successfully.

Post Condition

The user will register and be logged in.

2.1.6 Requirement 3: Sending a Message

2.1.6.1 Description & Priority

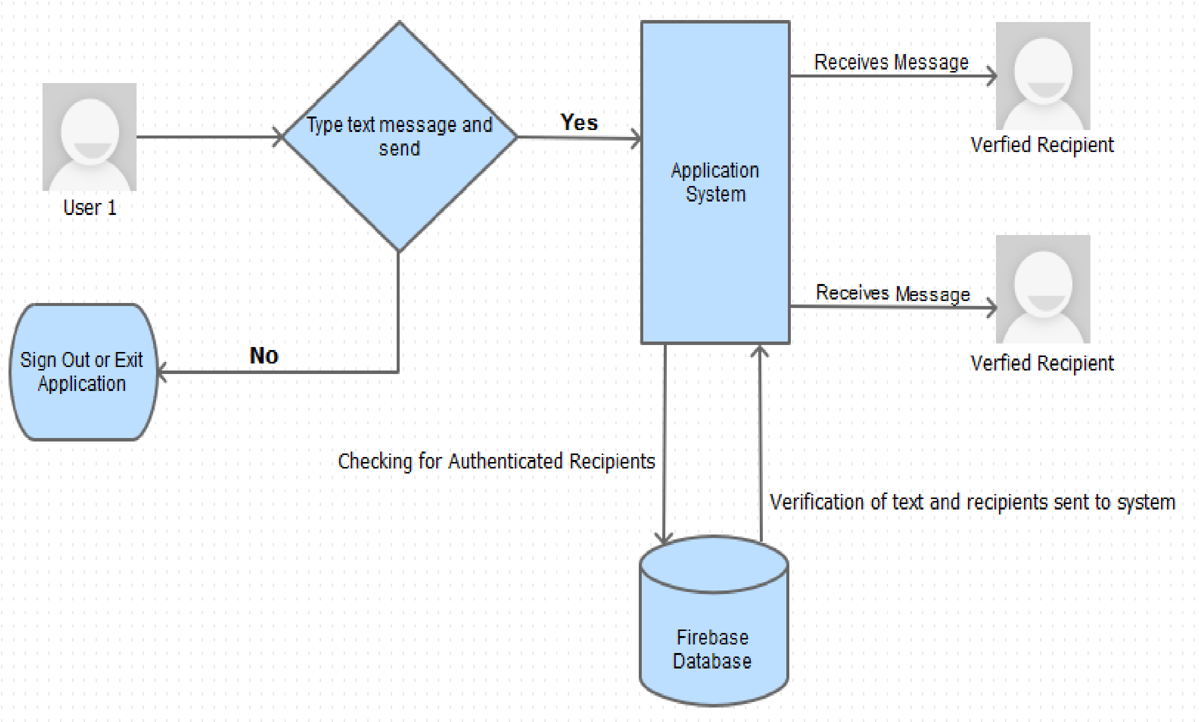
This use case is third in the priority list and describes the process of sending a message to another user via Auto Messenger.

2.1.6.2 Use Case

Scope

The scope of this use case is to describe the process of using Auto Messenger to send a message to another registered user.

Use Case Diagram


User 1
Type text message and send
Yes
No
Sign Out or Exit Application
Application System
Receives Message
Verfied Recipient
Receives Message
Verfied Recipient
Checking for Authenticated Recipients
Verification of text and recipients sent to system
Firebase Database

Precondition

The user s logged in as an authenticated user.

Activation

This use case starts when the user has logged in.

Main Flow

Actor Action: User begins to type a message and presses send.

System Response: Sending process initiated and executed.

Alternate Flow

Actor Action: The user does not send the message

System Response: System stays idle, user has the option to sign out or exit the application.

Termination

This flow is terminated when the user sends message or signs out.

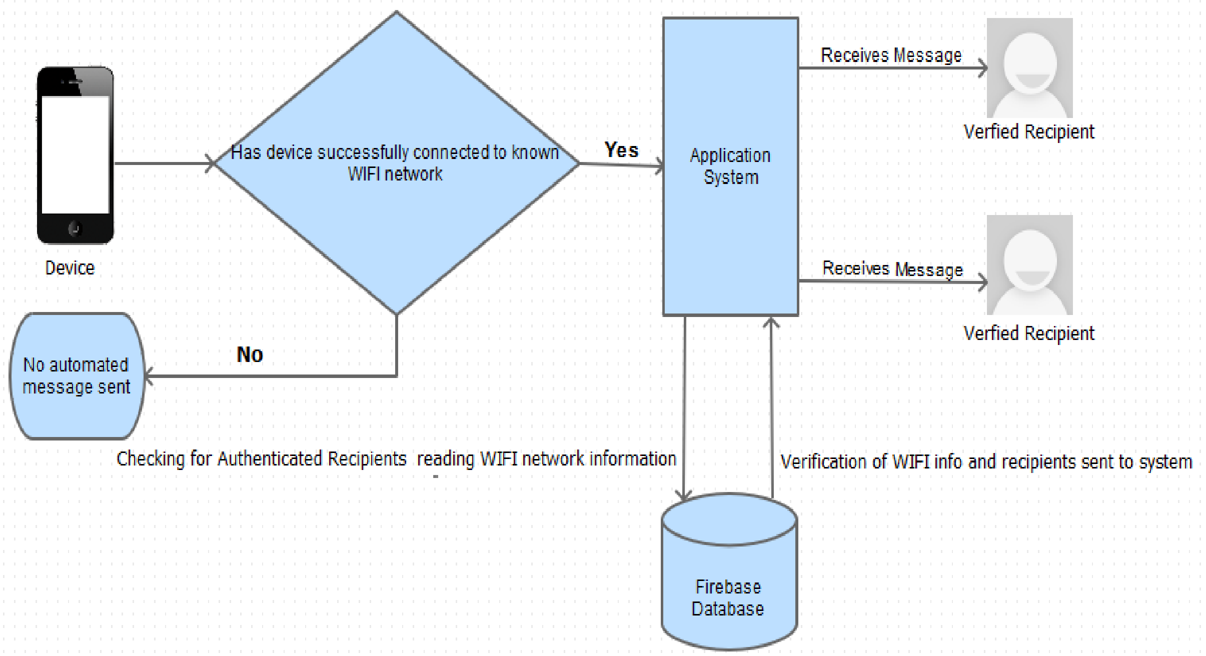
Post Condition

The system enters a wait state.

2.2 Auto Messaging System

This system is set up to send out an automated message to all verified chat recipients when a user's device has connected to a known WIFI network. When I say known, I mean a network that the device has already been connected to before and will continue to connect to automatically. On receiving verification of successful connection Auto Messenger will send a templated automated message "This user has connected to WIFI network: SSID" including the SSID of the WIFI network to all other users.

Use Case Diagram


Device
Has device successfully connected to known WIFI network
Yes
No
No automated message sent
Application System
Receives Message
Verfied Recipient
Receives Message
Verfied Recipient
Checking for Authenticated Recipients reading WIFI network information
Verification of WIFI info and recipients sent to system
Firebase Database

2.3 Non-Functional Requirements

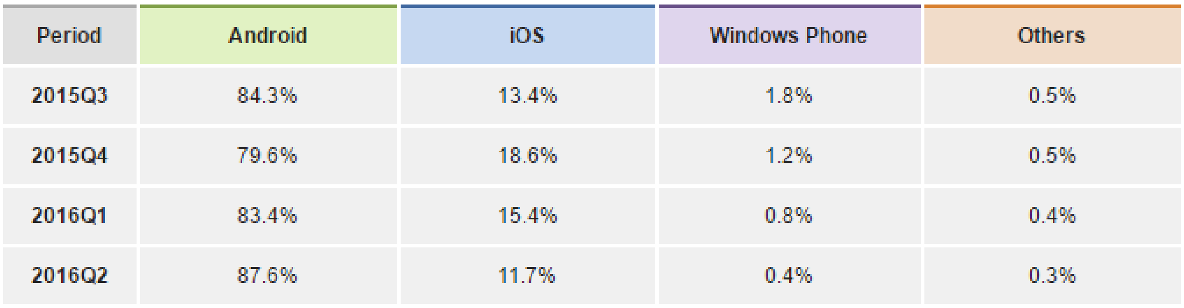
2.3.1 Performance & Response Time

The general advice on response time as per Jakob Nielsen’s book on usability is:

- 0.1 second is about the limit for having the user feel that the system is reacting instantaneously.
- 1.0 second is about the limit for the users flow of thought to stay uninterrupted.
- 10 seconds is about the limit for keeping the user’s attention focused on dialog.

2.3.2 Availability Requirement

The app will be built for smart devices that are built using Android Studio, this will make the app available to a wide market with the intention of making it available to IOS users in the future. Android has taken over the market in the last couple of years and has become the most popular OS for smart devices.



Period	Android	iOS	Windows Phone	Others
2015Q3	84.3%	13.4%	1.8%	0.5%
2015Q4	79.6%	18.6%	1.2%	0.5%
2016Q1	83.4%	15.4%	0.8%	0.4%
2016Q2	87.6%	11.7%	0.4%	0.3%

Source: IDC, Aug 2016

2.3.3 Security Requirement

When connecting to secured Wi-Fi networks users will be required to have access to the password for that network.

2.3.4 Maintainability Requirement

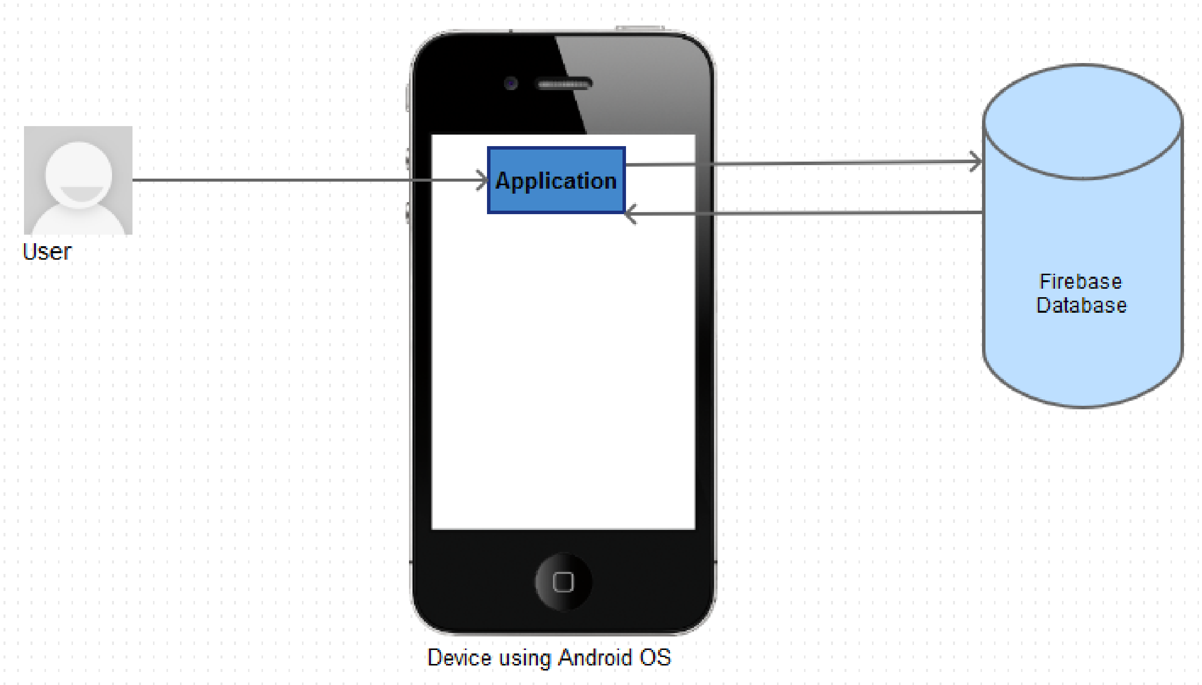
Users can report any bugs to developers allowing the issues to be addressed and fixed.

2.3.5 Extendibility Requirement

After the app is released it can still be updated with additional content. Development of the app will not cease after the initial release.

2.3.6 Design and Architecture

The following is the system architecture representation, the application itself will be hosted locally on a device that runs Android OS. The application will use Google Firebases infrastructure and implement Firebase code to use their secure database for storing user information and handling user messages. For a project of such a low scale compared to real life companies (Vodafone, 3, Meteor) this was the best approach to provide secure real-time messaging. I as Firebase admin to this project have sole access to user information, passwords will be encrypted and not visible to anyone including myself.

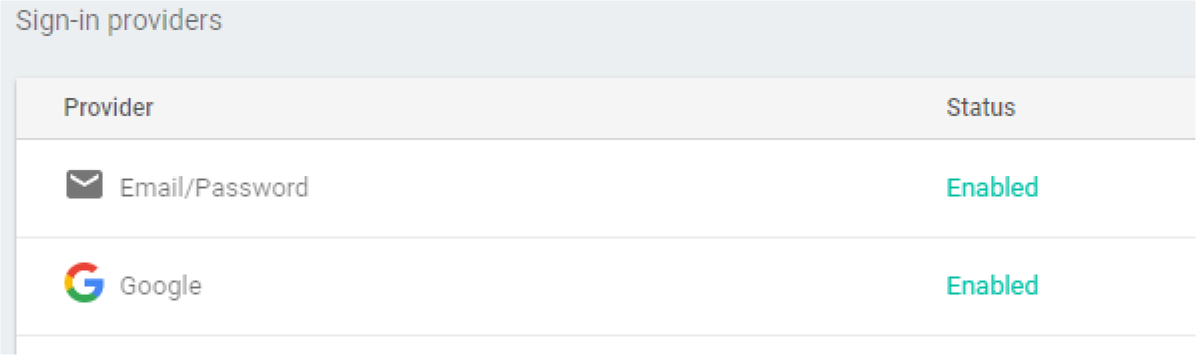


3. Implementation

Implementation languages that will be used in this project are mainly java which is the language used when developing in Android Studio. PHP was a consideration for implementing our database but now it looks as though we will be using Google Firebase to do the work for us. We can use Firebases authentication code to enable users to sign in using their Google account, to me this made the most sense because the app is Android specific. For a user to download an Android app they need to access the Google play store, for this, a user needs a Google account which would guarantee that if a user can download the app to a device then they can sign in. Anything else at this point is unforeseen but there may be other languages used for implementation of various aspects of the project that can be added in the later stages of development.

3.1 Firebase Authentication

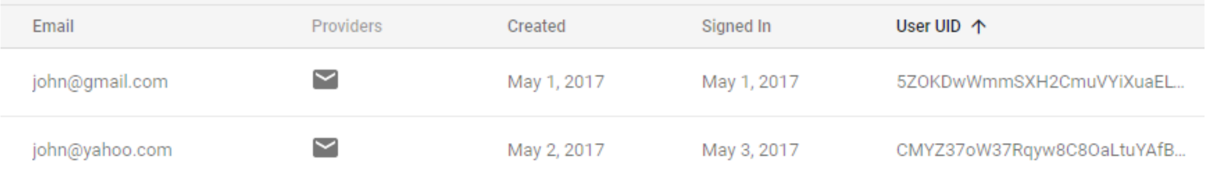
Firebase authentication uses SDK's and ready-made UI libraries to authenticate users to my app. The authentication process supports password authentication and integrates with popular federated identity providers such as Facebook, Twitter, for this app we will only use Google or email and password authentication.



Sign-in providers

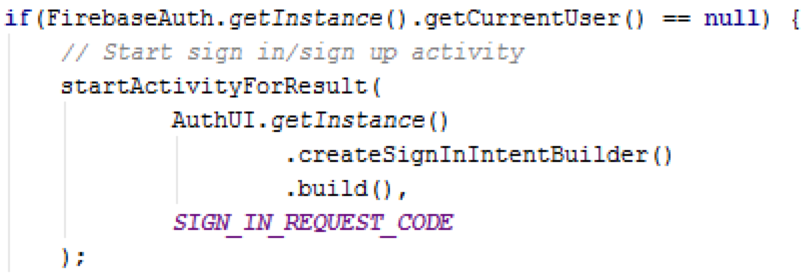
Provider	Status
Email/Password	Enabled
Google	Enabled

Example of authenticated user using email:



Email	Providers	Created	Signed In	User UID ↑
john@gmail.com	✉	May 1, 2017	May 1, 2017	5ZOKDwWmmSXH2CmuVYiXuaEL…
john@yahoo.com	✉	May 2, 2017	May 3, 2017	CMYZ37oW37Rqyw8C8OaLtuYAfB…

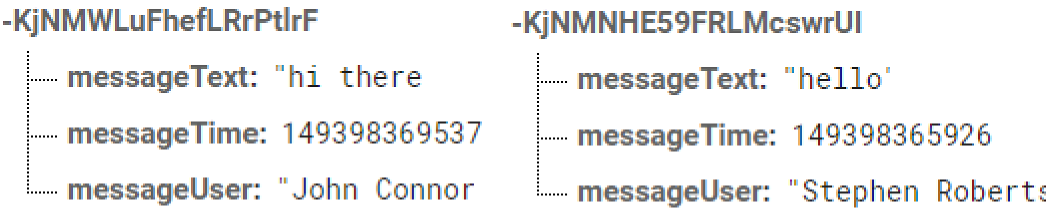
Sign in or Sign up code:



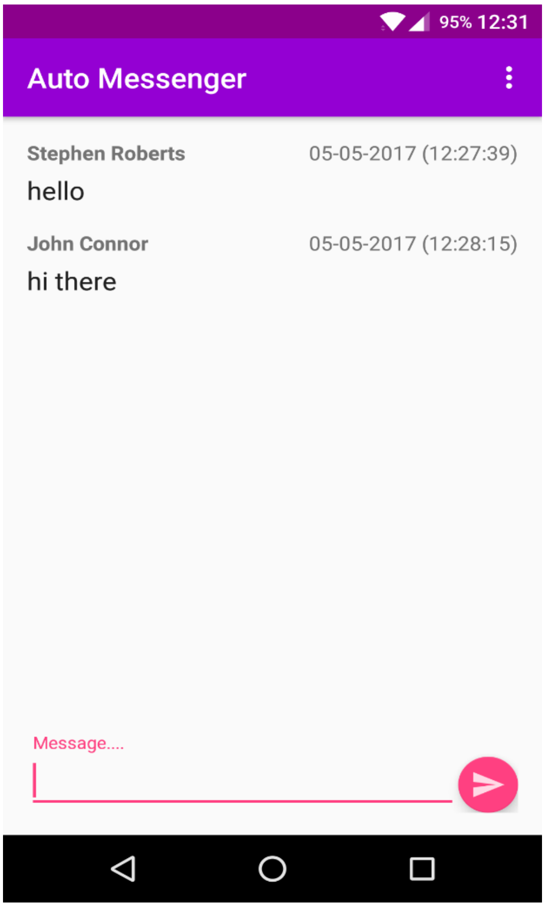
```
if(FirebaseAuth.getInstance().getCurrentUser() == null) {
    // Start sign in/sign up activity
    startActivityForResult(
            AuthUI.getInstance()
                    .createSignInIntentBuilder()
                    .build(),
            SIGN_IN_REQUEST_CODE
    );
```

3.2 Firebase Database

Not only is the database being used to store user information but it is also being used to store user text messages and send them back to the application for all users to see. Below is an example of the message and user structure as seen on Firebase and further below that is a screenshot of the same conversation within the app.



```
-KjNMWLuFhefLRrPtlrF
  messageText: "hi there
  messageTime: 149398369537
  messageUser: "John Connor

-KjNMNHE59FRLMcswrUI
  messageText: "hello'
  messageTime: 149398365926
  messageUser: "Stephen Roberts
```

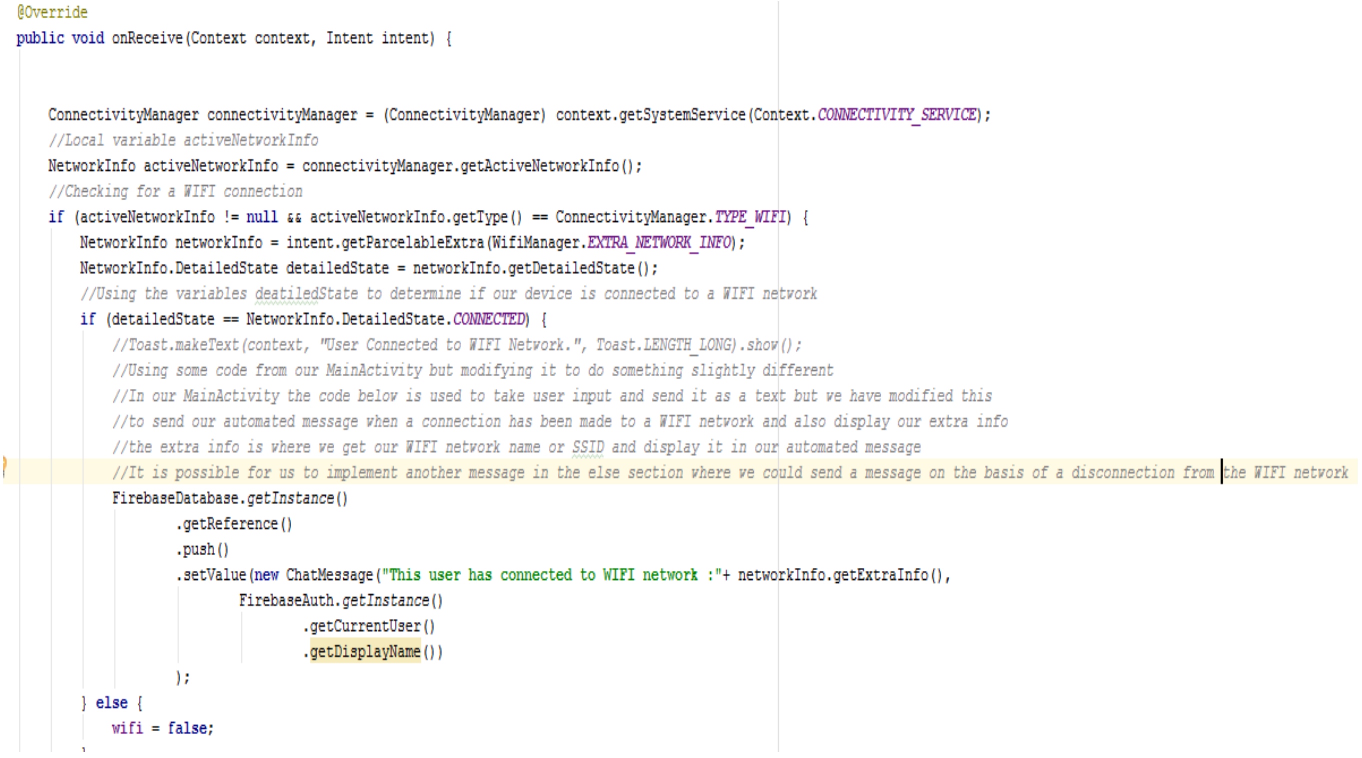


Stephen Roberts 05-05-2017 (12:27:39)
hello

John Connor 05-05-2017 (12:28:15)
hi there

3.3 Auto Messages Implementation

The auto messages are implemented using some modified Firebase code and broadcast receiver code where we ask the app to determine whether the device is connected to a WIFI network, if it is connected to a WIFI network we want some extra information. The extra information we seek is that of the name of the network or the SSID of the network. This extra information is then implemented into an automated message that will send automatically on successful connection of a device to a WIFI network. The following snippet of code displays the implementation and modification of some Firebase code that we are using to also send our manual text messages.



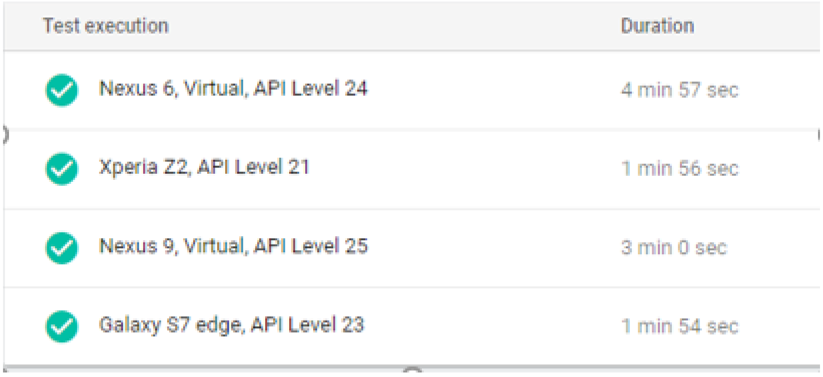
```
@Override
public void onReceive(Context context, Intent intent) {


    ConnectivityManager connectivityManager = (ConnectivityManager) context.getSystemService(Context.CONNECTIVITY_SERVICE);
    //Local variable activeNetworkInfo
    NetworkInfo activeNetworkInfo = connectivityManager.getActiveNetworkInfo();
    //Checking for a WIFI connection
    if (activeNetworkInfo != null && activeNetworkInfo.getType() == ConnectivityManager.TYPE_WIFI) {
        NetworkInfo networkInfo = intent.getParcelableExtra(WifiManager.EXTRA_NETWORK_INFO);
        NetworkInfo.DetailedState detailedState = networkInfo.getDetailedState();
        //Using the variables deatiledState to determine if our device is connected to a WIFI network
        if (detailedState == NetworkInfo.DetailedState.CONNECTED) {
            //Toast.makeText(context, "User Connected to WIFI Network.", Toast.LENGTH_LONG).show();
            //Using some code from our MainActivity but modifying it to do something slightly different
            //In our MainActivity the code below is used to take user input and send it as a text but we have modified this
            //to send our automated message when a connection has been made to a WIFI network and also display our extra info
            //the extra info is where we get our WIFI network name or SSID and display it in our automated message
            //It is possible for us to implement another message in the else section where we could send a message on the basis of a disconnection from the WIFI network
            FirebaseDatabase.getInstance()
                    .getReference()
                    .push()
                    .setValue(new ChatMessage("This user has connected to WIFI network :"+ networkInfo.getExtraInfo(),
                            FirebaseAuth.getInstance()
                                    .getCurrentUser()
                                    .getDisplayName())
                    );
        } else {
            wifi = false;
        }
```

4. Testing

4.1 Test Phase 1 – Firebase Robo Test

This testing phase is undertaken by the Firebase Robo test where I will build and upload the application APK file to Firebase. We will select several devices running on different API levels for the Robo test to automatically explore Auto Messenger to find any defects and report any crashes that may occur. This leads us to believe that installation and compatibility testing has been a success.

Device	API Level	Duration	Issues	Passed
Nexus 6	24	4 min 57 sec	0	Yes
Xperia Z2	21	1 min 56 sec	0	Yes
Nexus 9	25	3 min 0 sec	0	Yes
Galaxy S7 edge	23	1 min 54 sec	0	Yes



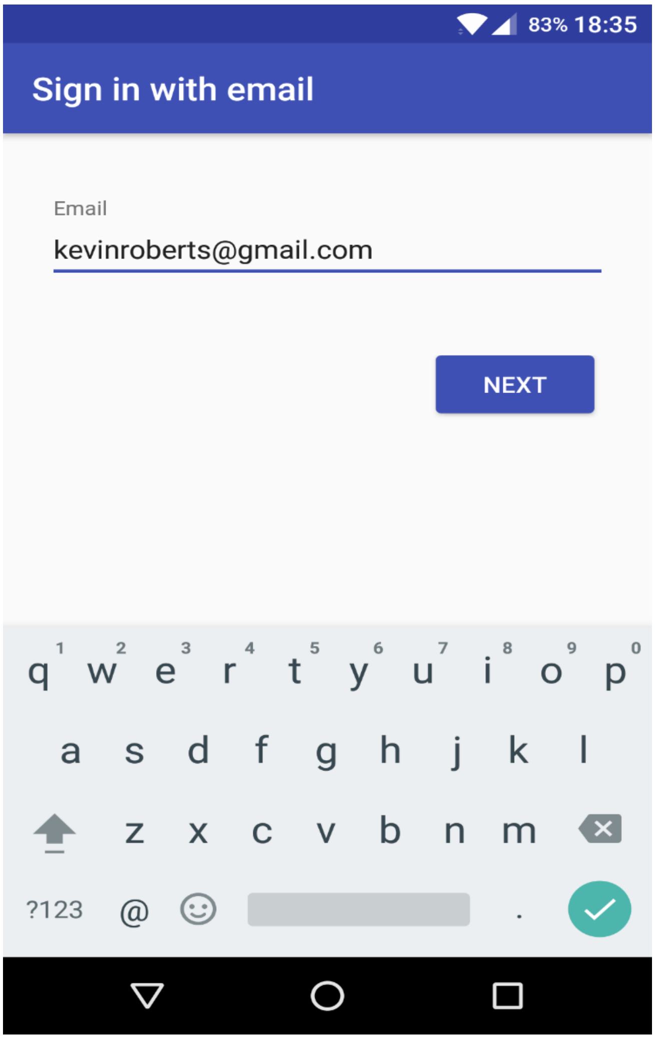
Test execution	Duration
✓ Nexus 6, Virtual, API Level 24	4 min 57 sec
✓ Xperia Z2, API Level 21	1 min 56 sec
✓ Nexus 9, Virtual, API Level 25	3 min 0 sec
✓ Galaxy S7 edge, API Level 23	1 min 54 sec

Screenshot from Actual Testing

4.2 Test Phase 2 – Create an Account

This test was used to monitor any implications that may arise during the creation of a new account on the app. The test was carried out by a volunteer tester and resulted in the successful completion of the create an account task. The app ran smoothly and worked as it should, the tester had no difficulty during the performance of this task. This leads us to believe that interface testing has been successful.

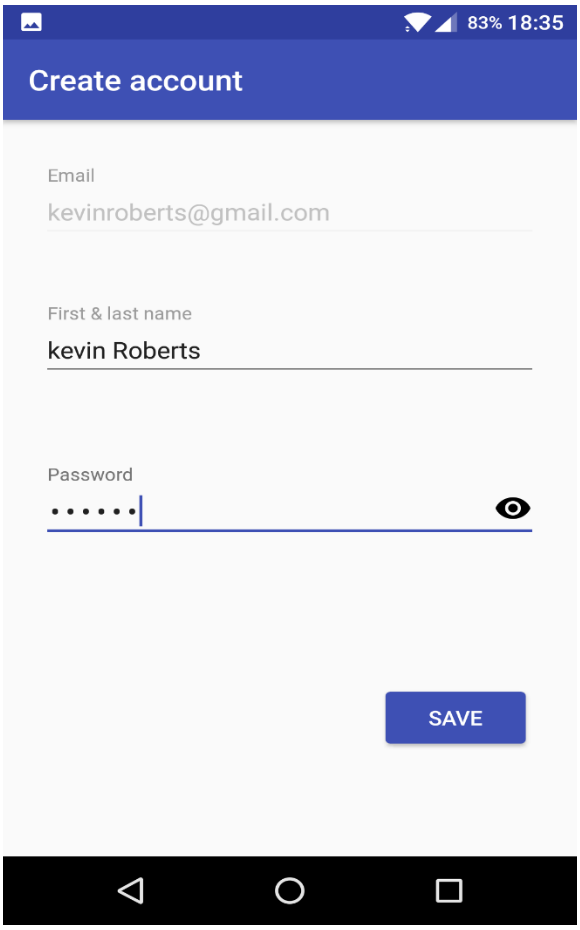
Step 1: Tester enters email address to be used in account setup, if the email is not recognised in the database, the tester is redirected to create an account.


Sign in with email

Email

kevinroberts@gmail.com

Step 2: Tester is redirected to create an account page as the email address is not recognised as an existing user.



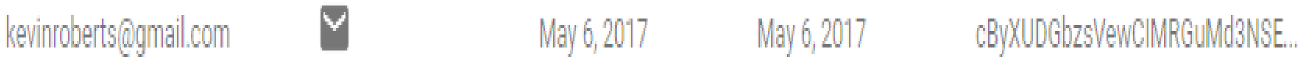
Create account

Email
kevinroberts@gmail.com

First & last name
kevin Roberts

Password
••••••

Creation of account has been successful:

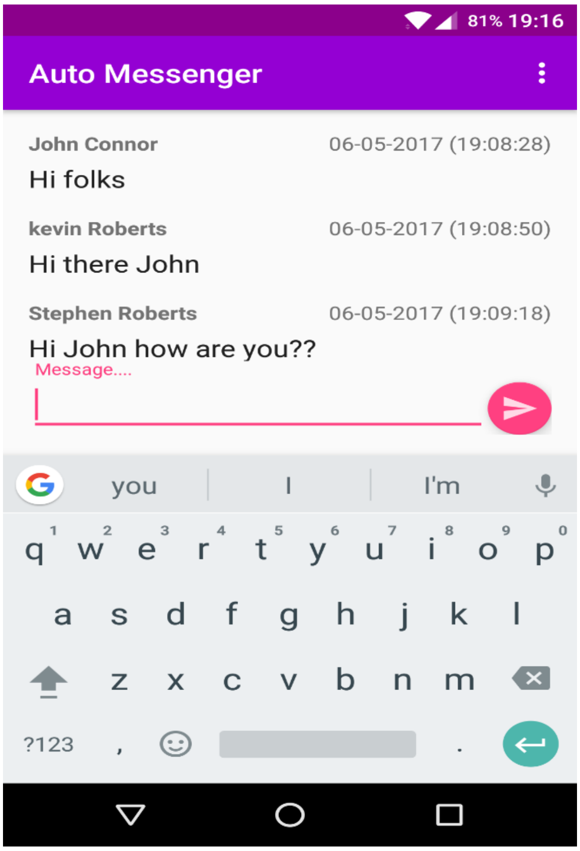


kevinroberts@gmail.com May 6, 2017 May 6, 2017 cByXUDGbzsVewCIMRGuMd3NSE...

4.3 Test Phase 3 – Sending & Receiving Messages

This test has been set up and implemented to display the sending and receiving of messages on three different devices by three different user accounts.

User	Device	Message	Successful
John Connor	Bush B2 Tablet	“Hi folks”	Yes
Kevin Roberts	OnePlus 2 Phone	“Hi there John”	Yes
Stephen Roberts	HTC M7 Phone	“Hi John how are you??”	Yes



John Connor 06-05-2017 (19:08:28)
Hi folks

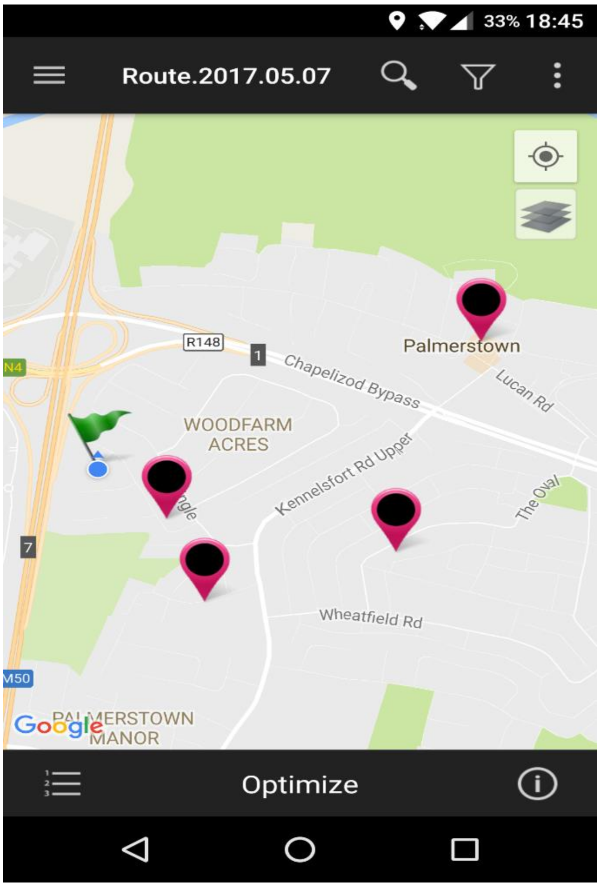
kevin Roberts 06-05-2017 (19:08:50)
Hi there John

Stephen Roberts 06-05-2017 (19:09:18)
Hi John how are you??

Screen Shot of conversation on OnePlus 2 Device

4.4 Test Phase 4 – Auto Messages

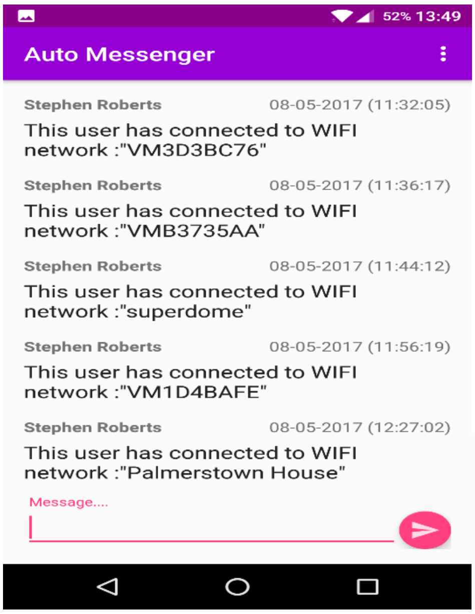
For this test, I have put together a list of destinations where I will physically visit one by one, these destinations will have a mixture of public and private WIFI networks that I have access to. When my device connects to these networks we should see a message sent out on the app to display that I have connected and print the SSID or name of the network along with time and date. Below is a map showing the route taken beginning and finishing at the green flag, followed by a table of results.



Location	Network Type	Network Name	Connection	Message Sent
1st (Green flag)	Private	VM3D3BC76	Successful	Yes
2nd	Private	VMB3735AA	Successful	Yes
3rd	Public	Superdome	Successful	Yes
4th	Private	VM1D4BAFE	Successful	Yes
5th	Public	Palmerstown House	Successful	Yes
6th (Green flag)	Private	VM3D3BC76	Successful	Yes

4.4.1 Test Phase 4 – Evaluation

This test was completed in full and the app performed as expected, connecting to the WIFI networks successfully and sending an automated message at each connection displaying the network name or SSID. A screenshot of the actual messages sent out can be seen in the image below, this is screenshotted from the device used in this testing phase.



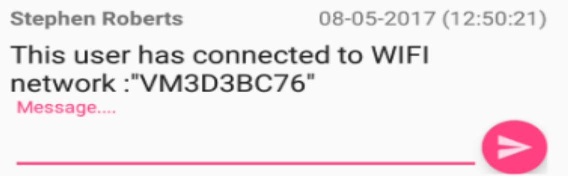
Stephen Roberts 08-05-2017 (11:32:05)
This user has connected to WIFI network :"VM3D3BC76"

Stephen Roberts 08-05-2017 (11:36:17)
This user has connected to WIFI network :"VMB3735AA"

Stephen Roberts 08-05-2017 (11:44:12)
This user has connected to WIFI network :"superdome"

Stephen Roberts 08-05-2017 (11:56:19)
This user has connected to WIFI network :"VM1D4BAFE"

Stephen Roberts 08-05-2017 (12:27:02)
This user has connected to WIFI network :"Palmerstown House"



Stephen Roberts 08-05-2017 (12:50:21)
This user has connected to WIFI network :"VM3D3BC76"

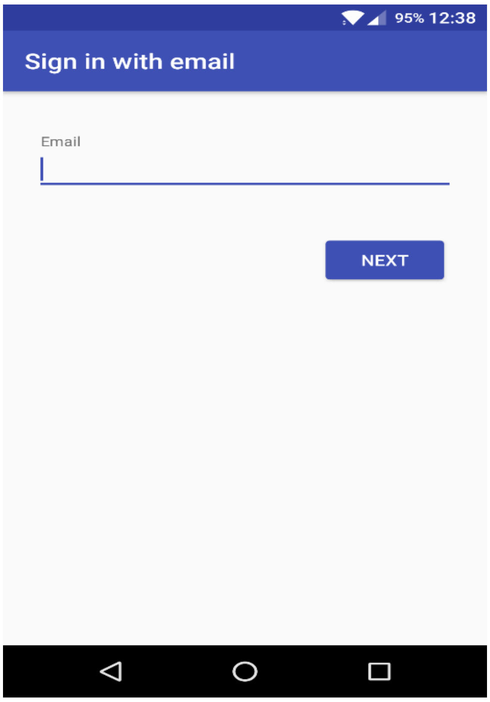
5. Graphical User Interface (GUI) Layout

The app will provide the user with 6 different GUI's:

1. Launch page, sign in with email.
2. Create an account.
3. Sign In.
4. Recover Password.
5. Main page, send text.
6. Main page, sign out.

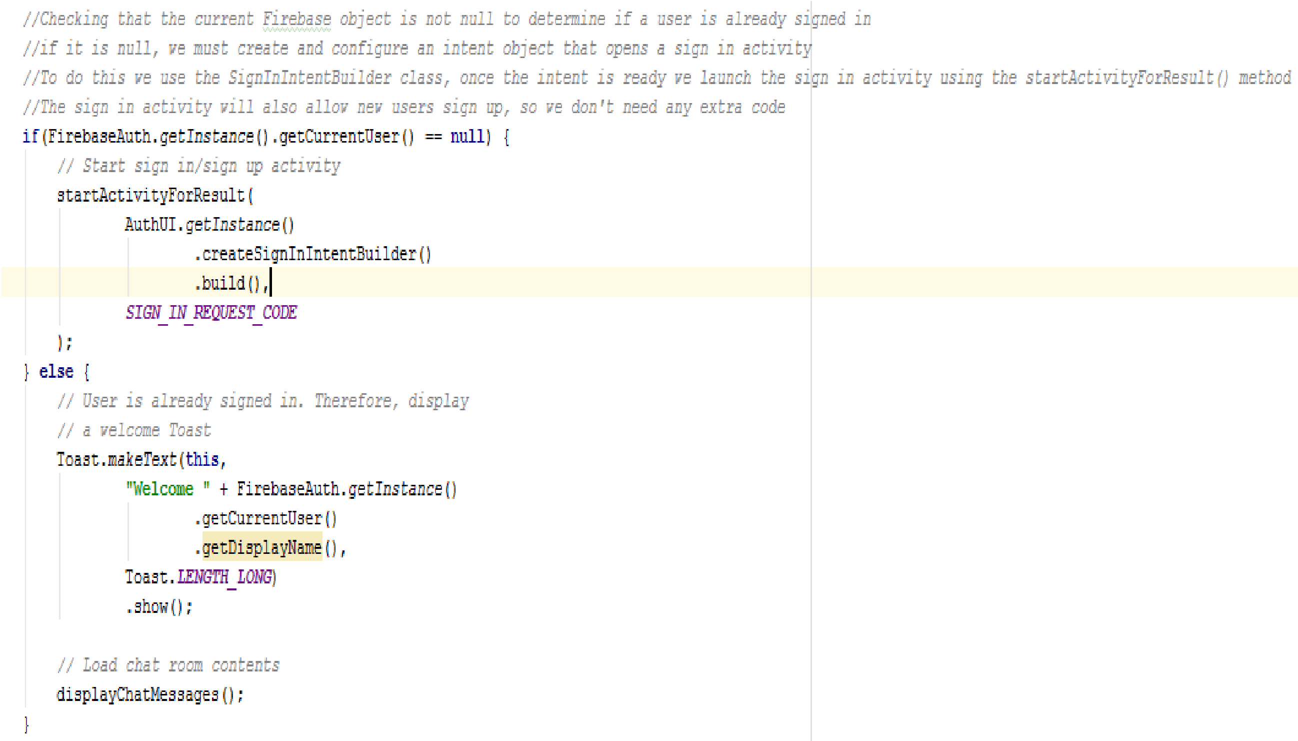
5.1 Launch Page

Launch page or the sign in with email page is where users can sign in using their email address, users will land on this page when the application is launched if they do not have an account or have signed out of their last visit. At this point users can sign in, sign up, or exit the application.



Launch Page

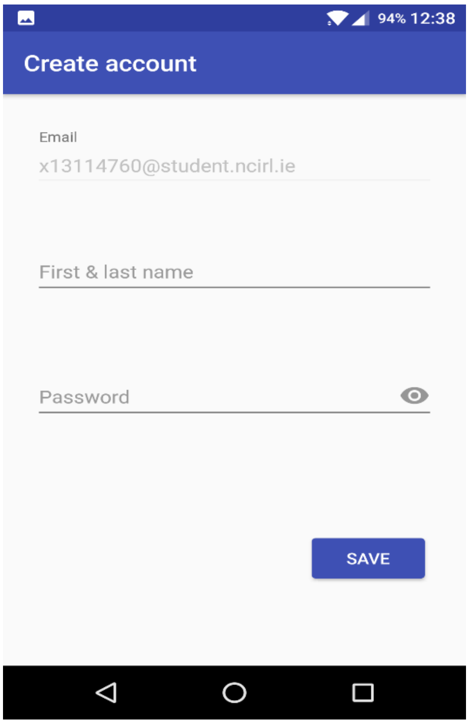
The following code depicts the sign in or sign up process that will have registered users emails recognised and non-recognised emails will be sent to our create an account activity which is next on our GUI list.



```
//Checking that the current Firebase object is not null to determine if a user is already signed in
//if it is null, we must create and configure an intent object that opens a sign in activity
//To do this we use the SignInIntentBuilder class, once the intent is ready we launch the sign in activity using the startActivityForResult() method
//The sign in activity will also allow new users sign up, so we don't need any extra code
if(FirebaseAuth.getInstance().getCurrentUser() == null) {
    // Start sign in/sign up activity
    startActivityForResult(
            AuthUI.getInstance()
                    .createSignInIntentBuilder()
                    .build(),
            SIGN_IN_REQUEST_CODE
    );
} else {
    // User is already signed in. Therefore, display
    // a welcome Toast
    Toast.makeText(this,
            "Welcome " + FirebaseAuth.getInstance()
                    .getCurrentUser()
                    .getDisplayName(),
            Toast.LENGTH_LONG)
            .show();

    // Load chat room contents
    displayChatMessages();
}
```

5.2 Create an Account

Once a user has entered and email account on the launch page Firebase will send the user to the create an account page if the email is not authenticated in the Firebase database. On this page users are prompted to enter their name and password they want associated with the corresponding email address entered. Users can then press save and their account setup will be complete.



94% 12:38

Create account

Email

x13114760@student.ncirl.ie

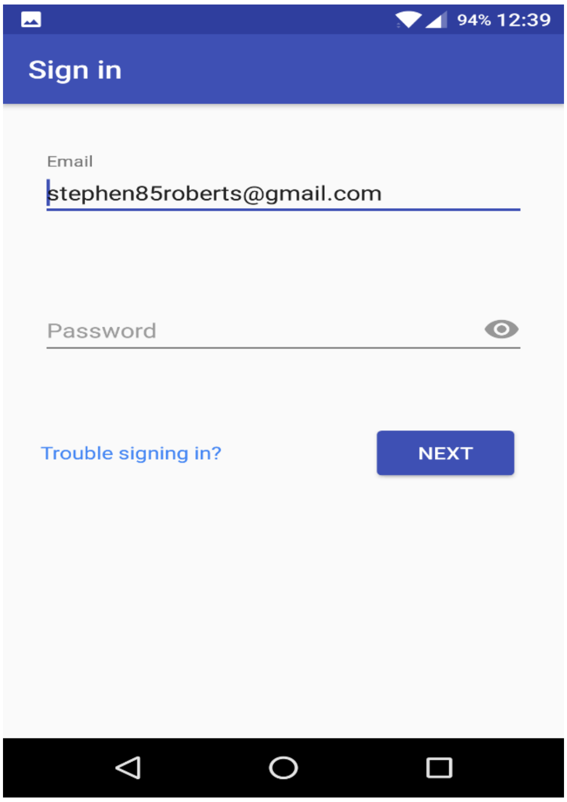
First & last name

Password

Create an Account Page

5.3 Sign In

Once a user has created an account they can use the sign in page to enter their password and gain access to the application. The sign in page also offers an option for users that are having trouble remembering their password, these users will be emailed a link to change their password. The link will only be sent to the email address that was used to create an account with.



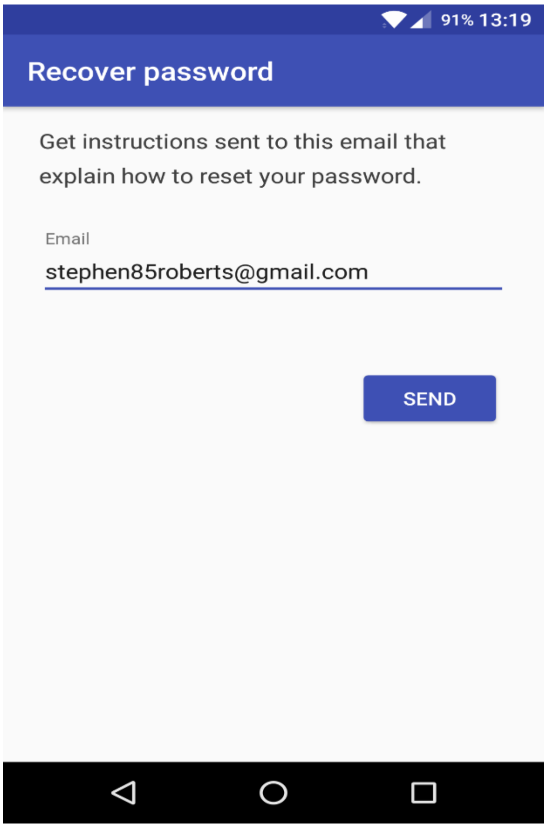
Email: stephen85roberts@gmail.com

Password

Sign In

5.4 Recover Password

This page will be initialised if the user click on the “trouble signing in” link on the sign in page, users will be emailed instructions with a link to set a new password.



Recover password

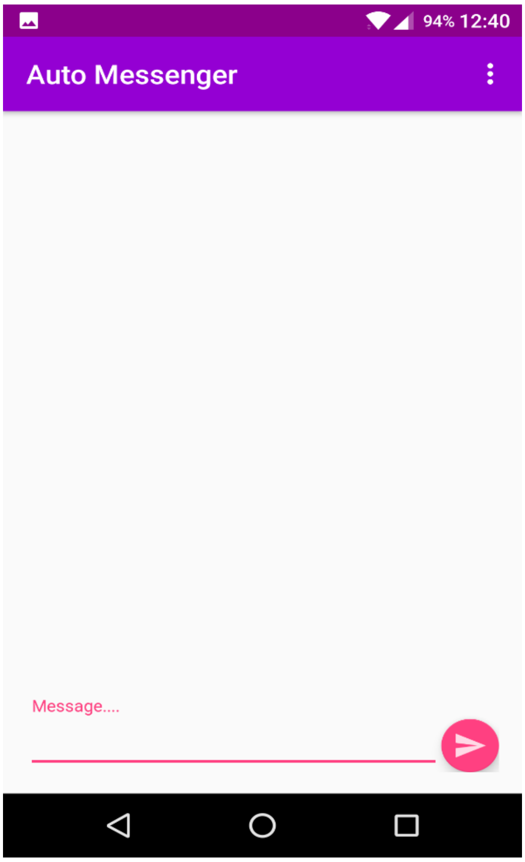
Get instructions sent to this email that explain how to reset your password.

Email
stephen85roberts@gmail.com

Recover Password

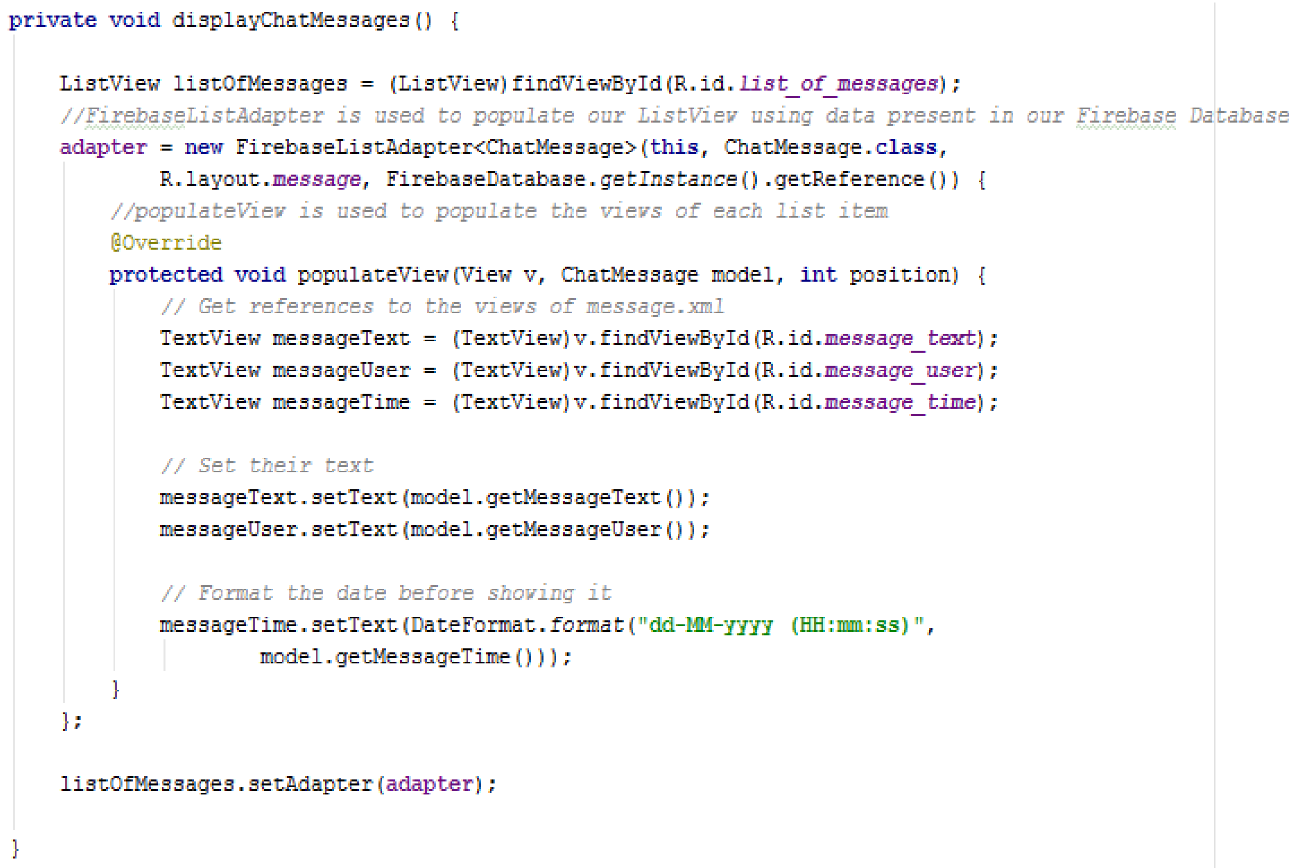
5.5 Main Page Send Text

On this page users can write, send, receive, and read all text messages from other users. This is also where users can read auto messages displaying information on other users WIFI network connections. This is the main feature that the app was built for, a templated message will appear on this page with the WIFI network SSID that a user has connected to.



Main Page, Send Text

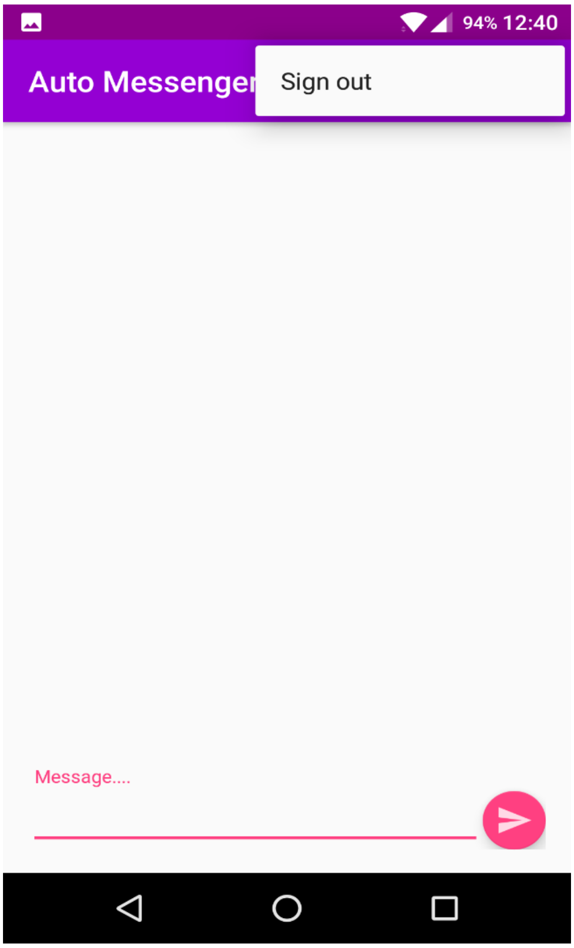
The following code is the structure for our text messages where users names, text, time, and date will be formatted to be displayed onscreen within the app.



```
private void displayChatMessages() {

    ListView listOfMessages = (ListView)findViewById(R.id.list_of_messages);
    //FirebaseListAdapter is used to populate our ListView using data present in our Firebase Database
    adapter = new FirebaseListAdapter<ChatMessage>(this, ChatMessage.class,
            R.layout.message, FirebaseDatabase.getInstance().getReference()) {
        //populateView is used to populate the views of each list item
        @Override
        protected void populateView(View v, ChatMessage model, int position) {
            // Get references to the views of message.xml
            TextView messageText = (TextView)v.findViewById(R.id.message_text);
            TextView messageUser = (TextView)v.findViewById(R.id.message_user);
            TextView messageTime = (TextView)v.findViewById(R.id.message_time);

            // Set their text
            messageText.setText(model.getMessageText());
            messageUser.setText(model.getMessageUser());

            // Format the date before showing it
            messageTime.setText(DateFormat.format("dd-MM-yyyy (HH:mm:ss)",
                    model.getMessageTime()));
        }
    };

    listOfMessages.setAdapter(adapter);

}
```

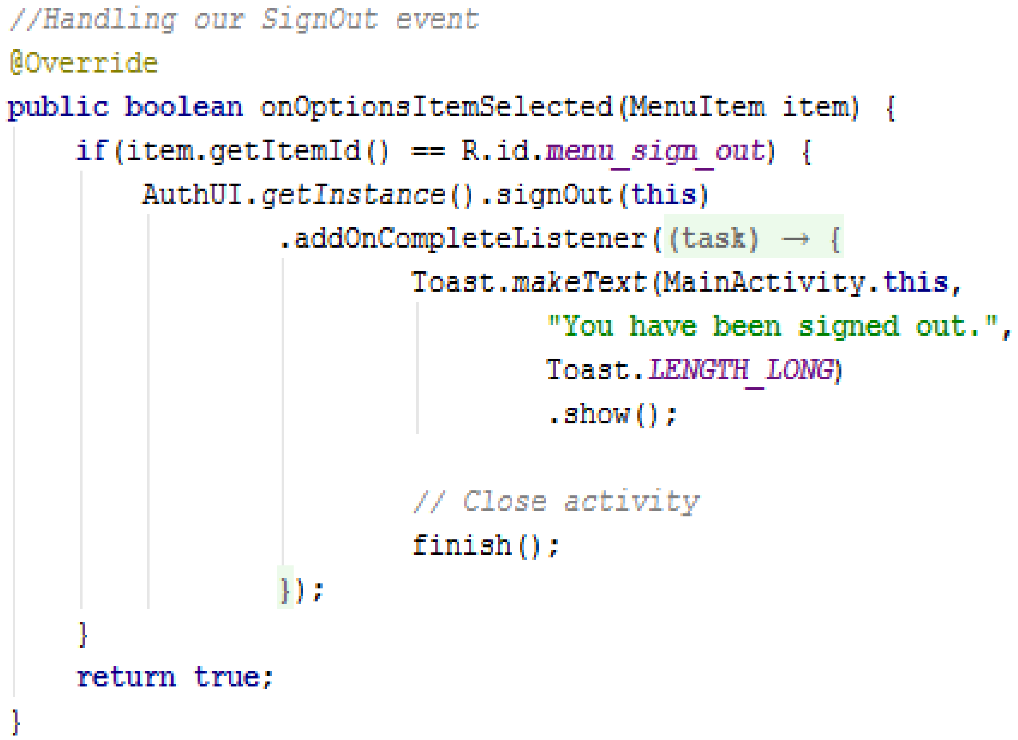
5.6 Main Page Sign Out

This is basically the same page as the main page but the sign out option has been selected, this can be used by all users when they want to fully sign out of the app.



Main Page Sign Out

This is the code to handle our sign out event when a user decides they want to sign out from the app, this activity will cease with the closure of the app and a toast message to display to the user that they have indeed been signed out.



```
//Handling our SignOut event
@Override
public boolean onOptionsItemSelected(MenuItem item) {
    if(item.getItemId() == R.id.menu_sign_out) {
        AuthUI.getInstance().signOut(this)
                .addOnCompleteListener((task) → {
                        Toast.makeText(MainActivity.this,
                                "You have been signed out.",
                                Toast.LENGTH_LONG)
                                .show();

                        // Close activity
                        finish();
                });
    }
    return true;
}
```

5.7 Evaluation and User Feedback

The evaluation of this application was carried out on several Android operated mobile phones several Android operated tablets. Mobile phones will make up a very high percentage of devices that will utilize this application but the app is fully responsive to tablet sized devices and functions in the same way as it does on the smaller mobile devices. So as of now no changes need to be applied to the application to improve cross platform performance.

Here is some general feedback and comments from friends and family who have tried the app:

> "It's a really great concept and I'd be happy to use this application"
>
> "Good app, easy to use and the auto message is a great idea. I for one am always forgetting to text people when I arrive home after a night out"
>
> "Good app, auto message is a good concept, but the app needs further development to progress to a higher standard!

More comments can be found in the survey located in the appendix.

6. Conclusions

The advantages of the app itself are nothing new, anybody that uses a smart device most likely has one or more messenger apps on that device. Messenger apps allow users to use free messaging and calling services once the user can connect to a wireless network, who would turn down such a service. The advantage of this idea is not in the app itself, the advantage is having the Auto Message feature, something that none of the big messenger apps have as a feature. Their disadvantage of not having a feature like this implemented is ultimately my ideas advantage, but them having amounted massive popularity over the years is my apps disadvantage. Do I see an app like mine being able to overthrow a big messenger like WhatsApp or Facebook Messenger?? Absolutely not! but my app does hold what I believe to be a key feature that would only make the already immense popularity of those messenger apps grow even further, while helping to keep people safe and that's the ultimate goal, user safety and security.

7. Further Development

The system is open to evolve, over time extra features may be added to the system such as introducing an auto emailing feature as suggested by one of our survey takers. End to end encryption of messages should be added in the future to keep user conversations safe and secured. A function for mapping the most recent route of a user where their device made several Wi-Fi connections could be added to the application to cater for certain users, such as those who are of special needs but like to maintain some independence. The only limitations of this project right now are my lack of experience in the field and my limited knowledge of the overall area of the project.

8. References

www.idc.com. (2016). *IDC: Smartphone OS Market Share*. [online] Available at: http://www.idc.com/prodserv/smartphone-os-market-share.jsp [Accessed 18 Nov. 2016].

Hathibelagal, A. (2017). *How to Create an Android Chat App Using Firebase*. [online] Code Envato Tuts+. Available at: https://code.tutsplus.com/tutorials/how-to-create-an-android-chat-app-using-firebase--cms-27397 [Accessed 24 Jan. 2017].

www.tutorialspoint.com. (2017). *Android Broadcast Receivers*. [online] Available at: https://www.tutorialspoint.com//android/android_broadcast_receivers.htm [Accessed 10 Feb. 2017].

Developer.android.com. (2017). *ScanResult | Android Developers*. [online] Available at: https://developer.android.com/reference/android/net/wifi/ScanResult.html [Accessed 20 Mar. 2017].

Programcreek.com. (2017). *Java Code Example android.net.wifi.ScanResult*. [online] Available at: http://www.programcreek.com/java-api-examples/index.php?api=android.net.wifi.ScanResult [Accessed 25 Apr. 2017].

Developer.android.com. (2017). *Android Developers*. [online] Available at: https://developer.android.com/index.html [Accessed 16 Oct. 2016].

YouTube. (2017). *TVAC Studio*. [online] Available at: https://www.youtube.com/user/akshayejh [Accessed 6 Nov. 2016].

Firebase. (2017). Firebase. [online] Available at: https://firebase.google.com/?utm_source=google&utm_medium=cpc&utm_campaign=1001467%20%7C%20Firebase*%20Brand%20GENERIC%20%7C%20Global%20%7C%20en%20%7C%20Desk%2BTab%2BMobile%20%7C%20Text%20%7C%20BKWS%20%5B2017%5D&utm_term=%7Bkeyword%7D&gclid=Cj0KEQjw6LXIBRCUqIjXmdKBxZUBEiQA_f50PrOVefzowWBr8SRznX9QE3wpJL8tgbxuNFj7lVVg5igaAvVx8P8HAQ [Accessed 30 Oct. 2016].

9. Appendix

9.1 Project Proposal

Auto Messenger

Stephen Roberts, x13114760, x13114760@student.ncirl.ie

BSc (Hons) in Computing 4th Year

Specialization: Gaming and Multimedia

October 21st, 2016

9.1.1 Objectives

The objective of this project is to succeed in creating and developing a fully functional mobile application that will provide the user all the features of any messenger app on the market at the moment, but with added features and functionality that are not currently available on any of the most popular messenger apps. The app will be developed develop using Android Studio with the intention of making the app available on the Google Play Store for download and installation, this will users of android smart devices download and use the app.

9.1.2 Background

The app idea actually came to me during the summer of 2015, I hadn't been consciously thinking of ideas of things that I could create or develop. I knew coming into 4th year what was going to be expected of the students with regards the final year project and that we should really have our own idea, but I felt at that stage in the summer of 2015 that I had plenty of time to figure out an idea. It's quite strange how the app sprung into my head so quickly and it made sense, it was relevant, and made sense with regards to modern society and the world we live in. I was just sitting around reading some articles online when I came across one and the headline obviously caught my attention, I can't remember what that headline was though. The article was about a young girl who had come up with the idea of nail polish that would change in colour if the person wearing it had been spiked, straight away I thought it was a fantastic idea. Around the time of me reading the article there was an ongoing case of an Irish girl that had been spiked and murdered in Scotland and I thought to myself, god that nail polish had the potential to have saved that girls life and a lot of heartache and pain. So, at this stage I've read the article and was really impressed by the idea and I pondered to myself, "how could an idea like that be implemented from a tech point of view." Ok what do the majority of people have on their person on nights out "mobile phone" I thought, ah that was easy!! How can we use peoples mobile phones to make them in some way safer on nights out, it's got to be something that users don't consciously have to using for it to function correctly, something they can have set before they go on their night out and not have to worry about remembering while out. This is where the idea really started to flourish for me and my brain just ran with it, I was actually surprised one of the big messenger apps hadn't got this feature implemented seen as how they are the most popular downloaded and installed apps. WhatsApp and Facebook Messenger on android alone have over a billion downloads each. The idea is for my messenger app to have a feature where the user can allow for and set a location to specific WI-FI network, for example a user's home WI-FI network can be set to location "home." Then when the user's phone automatically connects to an already activated WI-FI network the app will send out a templated message to chosen groups or individuals.

The message will be based on what locations have been set for specific networks and what network the phone has connected to (“work” “home” “college” “Johns house”) the templated message could look similar to this (“user-name” has arrived at “set location”). If this can be achieved I feel it’s a feature well worth having on a very popular type of app and will hopefully help keep people safe and eliminate elements of worry.

9.1.3 Technical Approach

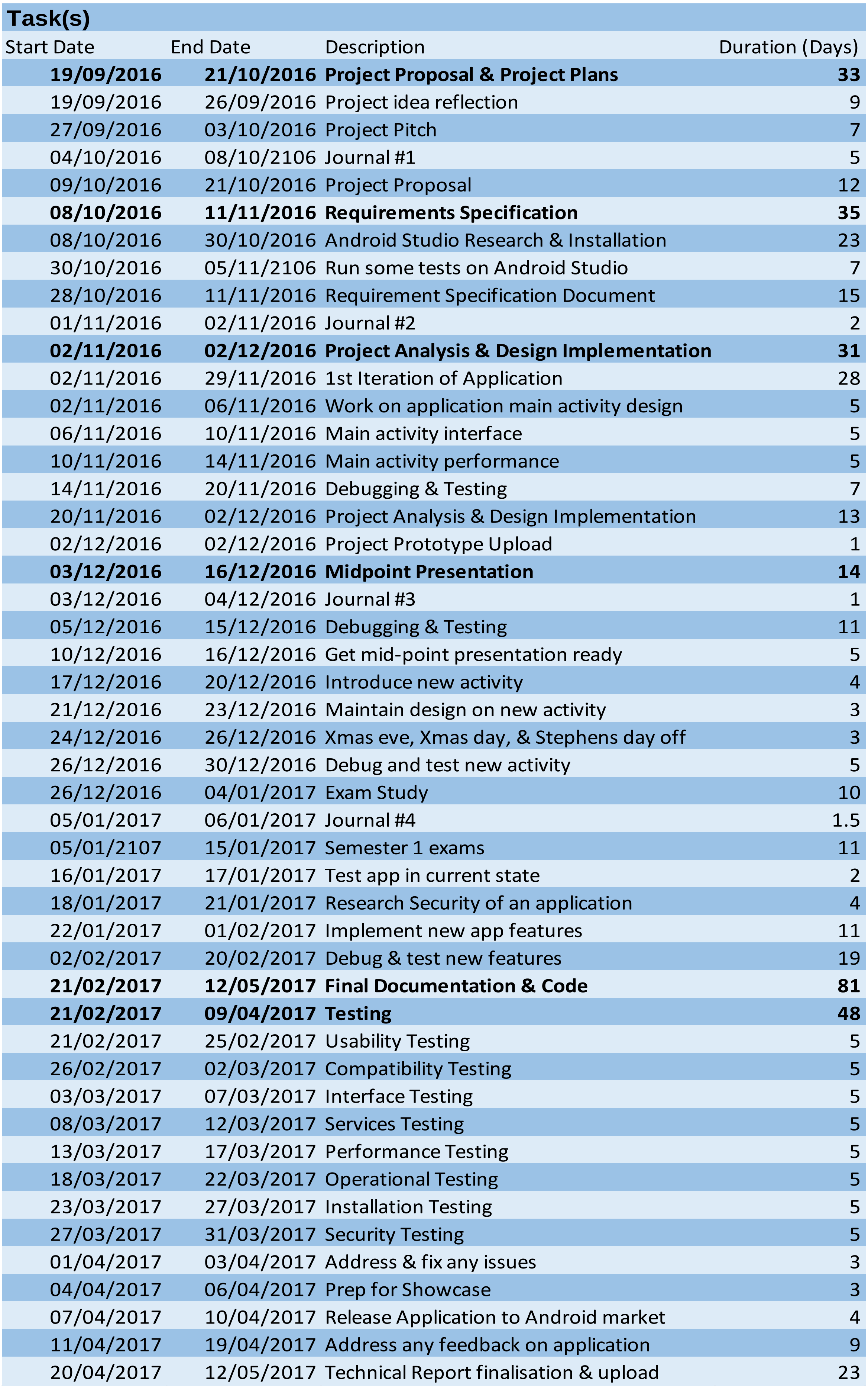
As of right now I’m not really sure how the project should be approached, research is going to play a big part in the project because I’m attempting something I really don’t know how to implement or achieve and I’m wondering if I’m biting off more than I can chew. The main reason I’m sticking with the idea is because I feel it’s something worth creating, I just hope it can be achieved because there’s a lot resting on the outcome being a good one. The way I’ve approached projects in the past is to try being as agile as possible and do what it takes to get the work done so I guess that same approach will be taken this time around as I probably know no better. At the end of the day I’m willing to do whatever it takes to get this project completed but do I have the technical knowledge or knowhow is another question. As part of my research I will be studying and reviewing material that’s offered on the Android Studio website free of charge and try and take advantage of any online tutorials available. I have already contacted Signal Messenger App Developers Open Whisper Systems to try and obtain any advice, tips, or steps to be taken when setting out to develop a messenger app but it was to no avail. Their code is open source so maybe having a look through that might send me in the right direction.

9.1.4 Special Resources Required

Android studio is completely free to download and use which is a massive bonus when trying to develop an android app. Tutorials on Android Studio and android development can also be found free of charge on the Android Studio website and alike on YouTube, a book called Android Studio Development Essentials written by Neil Smyth can be purchased at a cost of $28 and then the only other cost will come when the app needs to be published on the Google Play Store where there is a charge of $25 for developers to publish their apps. Any other special resources that may be required have not been foreseen at this point in time. So, as we stand now the total cost of the project from a monetary perspective is $53 and from any other perspective it may cost me my sanity and a few more grey hairs.

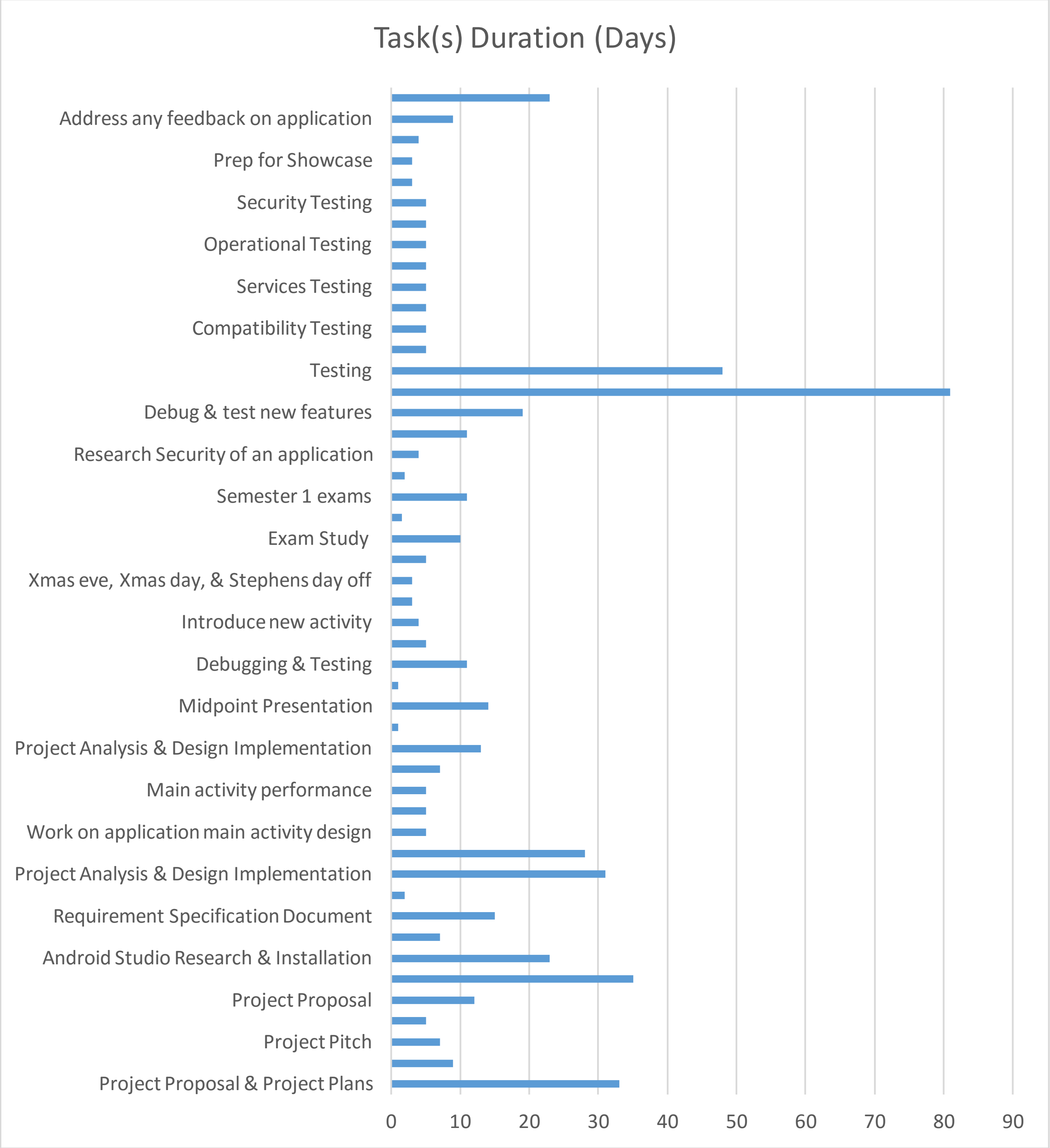
9.1.5 Project Plan

Please find project plan table completed and shown on next page, this will be uploaded in an excel document accompanied with a chart displaying the information in the table.



Task(s)

Start Date	End Date	Description	Duration (Days)
19/09/2016	**21/10/2016**	**Project Proposal & Project Plans**	**33**
19/09/2016	26/09/2016	Project idea reflection	9
27/09/2016	03/10/2016	Project Pitch	7
04/10/2016	08/10/2106	Journal #1	5
09/10/2016	21/10/2016	Project Proposal	12
08/10/2016	**11/11/2016**	**Requirements Specification**	**35**
08/10/2016	30/10/2016	Android Studio Research & Installation	23
30/10/2016	05/11/2106	Run some tests on Android Studio	7
28/10/2016	11/11/2016	Requirement Specification Document	15
01/11/2016	02/11/2016	Journal #2	2
02/11/2016	**02/12/2016**	**Project Analysis & Design Implementation**	**31**
02/11/2016	29/11/2016	1st Iteration of Application	28
02/11/2016	06/11/2016	Work on application main activity design	5
06/11/2016	10/11/2016	Main activity interface	5
10/11/2016	14/11/2016	Main activity performance	5
14/11/2016	20/11/2016	Debugging & Testing	7
20/11/2016	02/12/2016	Project Analysis & Design Implementation	13
02/12/2016	02/12/2016	Project Prototype Upload	1
03/12/2016	**16/12/2016**	**Midpoint Presentation**	**14**
03/12/2016	04/12/2016	Journal #3	1
05/12/2016	15/12/2016	Debugging & Testing	11
10/12/2016	16/12/2016	Get mid-point presentation ready	5
17/12/2016	20/12/2016	Introduce new activity	4
21/12/2016	23/12/2016	Maintain design on new activity	3
24/12/2016	26/12/2016	Xmas eve, Xmas day, & Stephens day off	3
26/12/2016	30/12/2016	Debug and test new activity	5
26/12/2016	04/01/2017	Exam Study	10
05/01/2017	06/01/2017	Journal #4	1.5
05/01/2107	15/01/2017	Semester 1 exams	11
16/01/2017	17/01/2017	Test app in current state	2
18/01/2017	21/01/2017	Research Security of an application	4
22/01/2017	01/02/2017	Implement new app features	11
02/02/2017	20/02/2017	Debug & test new features	19
21/02/2017	**12/05/2017**	**Final Documentation & Code**	**81**
21/02/2017	**09/04/2017**	**Testing**	**48**
21/02/2017	25/02/2017	Usability Testing	5
26/02/2017	02/03/2017	Compatibility Testing	5
03/03/2017	07/03/2017	Interface Testing	5
08/03/2017	12/03/2017	Services Testing	5
13/03/2017	17/03/2017	Performance Testing	5
18/03/2017	22/03/2017	Operational Testing	5
23/03/2017	27/03/2017	Installation Testing	5
27/03/2017	31/03/2017	Security Testing	5
01/04/2017	03/04/2017	Address & fix any issues	3
04/04/2017	06/04/2017	Prep for Showcase	3
07/04/2017	10/04/2017	Release Application to Android market	4
11/04/2017	19/04/2017	Address any feedback on application	9
20/04/2017	12/05/2017	Technical Report finalisation & upload	23


Task(s) Duration (Days)
Address any feedback on application
Prep for Showcase
Security Testing
Operational Testing
Services Testing
Compatibility Testing
Testing
Debug & test new features
Research Security of an application
Semester 1 exams
Exam Study
Xmas eve, Xmas day, & Stephens day off
Introduce new activity
Debugging & Testing
Midpoint Presentation
Project Analysis & Design Implementation
Main activity performance
Work on application main activity design
Project Analysis & Design Implementation
Requirement Specification Document
Android Studio Research & Installation
Project Proposal
Project Pitch
Project Proposal & Project Plans
0
10
20
30
40
50
60
70
80
90

9.1.6 Technical Details

Implementation languages that will be used in this project are mainly java which is the language used when developing in Android Studio and most likely PHP for interacting with the database, anything else at this point is unforeseen but there may be other languages used for implementation of various aspects of the project that can be added in the later stages of development.

9.1.7 Evaluation

The download of Android Studio must be carried out along with the download of the latest jdk files and android SDK to get started. When developing an app using Android Studio the testing and evaluation of the app can be done using any android operated phone as long as the phone supports the version of the android OS that the app is built on. I will start by turning on the ‘Developer Option’ on my personal android device to insure my device can be used for testing the app. The device needs to be connected to the computer being used for development and steps for setting up the device to run the app on must be followed in Android Studio. Once the app is installed on a device, functional test cases may be carried out to determine what functionality passes and what fails, what should be kept and what should be gotten rid of. Google Firebase Android Cloud can be used during testing and this will smoke test the app for us by using robots to do so. These test cases can be kept track of using a spreadsheet. The final system will need to be evaluated by at minimum two end users, the users will need to download and install the app, set up a connection between each user device (have each other’s phone number stored in their contacts) the devices will need to be able to connect to a WIFI network then we can test how the automatic templated messaging service will work between the devices when connected to an established WIFI network. Below is a list of types of testing to be carried out over the course of the project development.

Testing to be carried out:

- **Usability Testing** – This testing will be carried out to ensure that the app easy to use and provides a satisfactory user experience.
- **Compatibility Testing** – The testing of the application to ensure compatibility across different devices, screen sizes, and OS versions.
- **Interface Testing** – The testing of menu options, buttons, navigation, and flow of the application between activities.
- **Services Testing** – Testing the services of the app online and offline.
- **Performance Testing** – Testing the performance of the app over different connections and testing battery usage.
- **Operational Testing** – Testing backup and recovery plans if the battery were to go down or data loss while upgrading the application.
- **Installation Testing** – Carried out to confirm validation of the application by installing and uninstalling the app on different devices.
- **Security Testing** – This testing will be carried out to insure the app is secure and user details are safely stored.

9.2 Monthly Journals

9.2.1 September 2016

Month: September

My name is Stephen Roberts and as I set off on the task of beginning this 4th year project journal I am 31 years of age. My mum and I live together in Palmerstown west Dublin and I work part-time as a mechanic in Superdome bowling alley Palmerstown as I have done throughout my time in college. Currently I am in my second week of my 4th and final year of my degree at The National College of Ireland (October 2016).

I decided at 27 that I would return to education on the advice and encouragement of some good friends. That was in the January of 2013 and I can still remember how daunting the whole idea of it was, I can remember being called for an interview with the then course director I think he was?? Stefan. After talking for about 5-10mins I remember Stefan pulling out of his pocket some connect four blocks and proceeded to have me do this little task that I completed within 20 seconds, I already felt like a winner (laughs). So anyway, I was offered a place on the HCC course with the intention of progressing into the BSHC course, as I had discussed with Stefan who I remember laughing at my eagerness and saying, "one step at a time." I accepted my place immediately and commenced into HCC 1 in the September of 2013 and I had turned 28 the March previous, so it's safe to say I wasn't the spring chicken of the class but I was by no means the eldest which gave me great hope and inspiration for being able to give this a damn good go.

The whole reason for me wanting to return to college and get a third level education was mainly down to my situation, I had lost my full-time job during the economic downturn and ended up working a 30-hour week in another job that hadn't much to offer me in terms of progression. I felt like I was missing something in life and I figured out what I was missing was a sense of achievement, the feeling that I had worked hard for something and been rewarded at the end of it. That's where college came into the fray, it was my way of doing something and getting that sense of achievement, and trying to better myself to make for a better future.

So, it's the middle of September 2016 and I'm back in college for my fourth and final year, it's crazy how fast the years have gone by when I stop and think about it.

Now it's time to get super serious about the work that needs to be done over the next few months. Need to get good grades to finish with a good degree with the hope of going on and finding some meaningful work that's enjoyable and it's something that I want to do and be involved in (with a bit of luck it pays well too).

First class back in college is for our software project, no beating around the bush. We've got two weeks to get our idea together and be ready to pitch it to three lecturers who in turn will decide whether the project idea is good enough to proceed with. This sets in motion endless thinking of ideas that might be good enough to pursue but also relevant to modern society's needs and expectations. An idea with an element of something that hasn't been done before, an idea that could possibly be commercially viable, something simple that might take on a snowball effect and evolve over time.

I do have an idea in mind but I'm not really sure how or if I can achieve it to the level that is expected. I'm by no means a fantastic programmer although it's something I really enjoy, I love seeing the outcome of running a complete program and what they can do.

Anyway, I've got a little bit of time to do some more pondering and consider other available options before I must commit to and pitch my idea if I do decide to go ahead with that particular one.

There is always the option to take an idea provided by lecturers so if all else fails we're not totally out of options. We'll see!!

9.2.2 October 2016

So, we're into the beginning of October now and things are starting to pick up now, the work load is adding up and sometimes very hard to juggle and produce what's expected. Our project pitch is coming up in a few days and I must put together a couple of slides about my idea, this is so lecturers can have a brief indication of the idea before I go in and pitch the idea to three lecturers to see if I can continue with it. My pitch is scheduled for 3:30pm on Wednesday the 5th and the lecturers I am pitching my idea to are Francis Sheridan, Dominic Carr, and Mohammad Iqbal. My idea and the background as to how I came up with it is outlined below, and this is what I will be going through during my pitch to the lecturers.

The app idea came to me during the summer of 2015, I hadn't been consciously thinking of ideas of things that I could create or develop. I knew coming into 4th year what was going to be expected of the students with regards the final year project and that we should really have our own idea, but I felt at that stage in the summer of 2015 that I had plenty of time to figure out an idea. It's quite strange how the app sprung into my head so quickly and it made sense, it was relevant, and made sense with regards to modern society and the world we live in. I was just sitting around reading some articles online when I came across one and the headline obviously caught my attention, I can't remember what that headline was though. The article was about a young girl who had come up with the idea of nail polish that would change in colour if the person wearing it had been spiked, straight away I thought it was a fantastic idea. Around the time of me reading the article there was an ongoing case of an Irish girl that had been spiked and murdered in Scotland and I thought to myself, god that nail polish had the potential to have saved that girls life and a lot of heartache and pain. So, at this stage I've read the article and was really impressed by the idea and I pondered to myself, "how could an idea like that be implemented from a tech point of view." Ok what do most people have on their person on nights out "mobile phone" I thought, ah that was easy!!

Auto Messenger Technical Report

Student no. X13114760

How can we use people's mobile phones to make them in some way safer on nights out, it's got to be something that users don't consciously have to be using for it to function correctly, something they can have set before they go on their night out and not have to worry about remembering while already out. This is where the idea started to flourish for me and my brain just ran with it, I was surprised one of the big messenger apps hadn't got this feature implemented seen as how messenger apps are the most popular downloaded and installed apps.

WhatsApp and Facebook Messenger on android alone have over a billion downloads each. The idea is for my messenger app to have a feature where the user can allow for and set a location to a specific WI-FI network, for example a user's home WI-FI network can be set to location "home." Then when the user's phone automatically connects to an already activated WI-FI network the app will send out a templated message to chosen groups or individuals. The message will be based on what locations have been set for specific networks and what network the phone has connected to ("work" "home" "college" "Johns house") the templated message could look similar to this ("user-name" has arrived at "set location"). If this can be achieved I feel it's a feature well worth having on a very popular type of app and will hopefully help keep people safe and eliminate elements of worry.

Wednesday 05/10/2016: I had my pitch today and it seemed to go well, Francis and Mohammad liked the idea but Dominic was unsure as he explained he couldn't see the complexity of the idea, and was afraid that the idea wouldn't help me achieve good marks as per the marking scheme. Francis helped me realise that the app had even more potential and could be expanded to use GPS/hotspots for Alzheimer's sufferers who are in the initial stages and would still like to be able to keep some of their independence. The automated messaging feature in the app could be used to send out messages of the location of the device at that given time, to family members and friends. For example, if the user of the device (Alzheimer's sufferer) wants to go about a regular routine (go to the butchers, then on to the super market, stop off for a coffee, then home) messages for each location will be sent out letting the recipients know the device is at that location. This feature will surely alleviate some of the stress and worry that family members and friends will be feeling.

Monday 10/10/2016: I found out that my project idea was accepted and I can go ahead and start working on it, I got a yes from two lecturers out of three as I kind of expected I would. I can honestly say that at this point I have no idea how to even get this started, but I'm not going to worry about that just yet (I know I should.) I just know it's a clever idea and has the potential to make a positive difference to people's lives.

Friday 14/10/2016: I have started working on my project proposal now that I have confirmation of what I'm doing, I have until the 21st of October to upload it.

Friday 28/10/2016: I found out who my project supervisor is, Keith Maykock. I had Keith as a lecturer in 2nd year I think it was, for Computer Architecture and I also now have Keith as a lecturer this year for Artificial Intelligence. Probably one of the most knowledgeable people I have ever met, his intelligence is extremely enviable. We had a meeting today in a part of the college I have never been, up on the fourth floor in an office surrounded by glass and containing nice comfortable leather chairs. There were Keith and all the students he will be supervising, that were in the room for the meeting. Each student took a turn to explain their idea to Keith, then receive feedback and some suggestions on expanding certain parts of the idea. We established that we will be having this meeting every Monday afternoon from today on.

9.2.3 November 2016

It's the start of November now and my project proposal has been uploaded and I have begun working on my app. The work load has kept piling on and it's getting even harder to try and keep up with everything, in hind sight I probably should have tried to opt for a project that would have been easier to implement and just get me the marks I need, because I am dropping them elsewhere and the only suggested option for help was to defer some modules and do them next year.

Friday 04/11/2016: I've started working on the UI for the login and registration of my app, I'm trying to draw on limited experience of app development and use whatever resources I can, to make a user friendly and visually pleasing interface. My plan is to get as much done as possible this month in preparation for the midpoint presentation in mid-December and leave myself plenty of study time in December.

Friday 11/11/2016: After working on the user interface I am now moving onto adding the functionality behind the login and register. I am going to try use some code that I have used before to create similar functionality on an older app that I built. I done this using PHP, SQL, Workbench, and hosting on Gearhost.

Friday 18/11/2016: After plenty of testing and troubleshooting, I have been unable to make my PHP code work correctly and I cannot figure out why. So now I need to find an alternative way of doing what I need to do to complete the functionality that I'm after.

Friday 25/11/2016: Having not been able to create the functionality for my login and registration for the app using PHP, I have had to find an alternative way of doing this. After some research, I have decided to use Google Firebase. This will allow me to add some libraries and code to my app from Firebase. Firebase will look after the app database and store new user information including, name, email, and password.

9.2.4 December 2016

December already and I have started the login for my app using Google Firebase. Firebase will take care of login validation and database information.

Saturday 03/12/2016: I've been following some great tutorials on setting up Firebase and writing the code in Android Studio to link the app login with Firebase. I need to get this finished as soon as possible as I have my mid-point presentation in a little over a week.

Saturday 10/12/2016: The login and register UI are now fully functional, linked with Firebase, and ready to display as part of my mid-point presentation. Now I need to put everything together and get some slides ready for the presentation.

Thursday 15/12/2016: Had my mid-point presentation today with Keith Maycock and Manuel Tova-Izquierdo, apart from me being terrible at presentations it seemed to go very well. Manuel seemed to really like the idea and both lecturers gave very constructive feedback and even offered some ideas of how the idea could be improved.

9.2.5 January 2017

Christmas has been celebrated and has now passed, I'll be taking most this month to study for and complete my first semester exams.

Tuesday 03/01/2017: It has been a few weeks since I had my project open so I'm going to open Android studio and complete any updates needed. Updates downloaded and installed successfully.

Monday 16/01/2017: The exams are now over and the studying is paused for now, time for a short break before semester two begins and we start all over again.

Friday 20/01/2017: Getting ready now to begin semester two of my last year, have also been trying to research how I can complete what my app needs to do but I have been unsuccessful thus far. The functions that the app needs to perform are going to be tricky as I have been unable to find any example of what it will take to bring the functions to life.

9.2.6 February 2017

Semester two is now in full swing, there hasn't been any progress on the functionality of the project as I'm finding it extremely hard to figure out how to write the code for the app functionality.

Monday 06/02/2017: There haven't been any supervisor meetings set up now for a couple of weeks, we were originally meant to have them every Monday afternoon so I need to try set one up to see how progress can be made on the project.

Friday 10/02/2017: There have been a few issues with my supervisor and he has been absent from college so no meetings have been set up yet so I have been focusing on other module projects and CA's. The project seems like it is suffering at this stage but the college and out of college workload is unsustainable with very little in the form of assistance offered.

Monday 20/02/2017: It turns out my original project supervisor will not be back to the college this semester so this morning I was told I would be assigned a new project supervisor and we will continue from there.

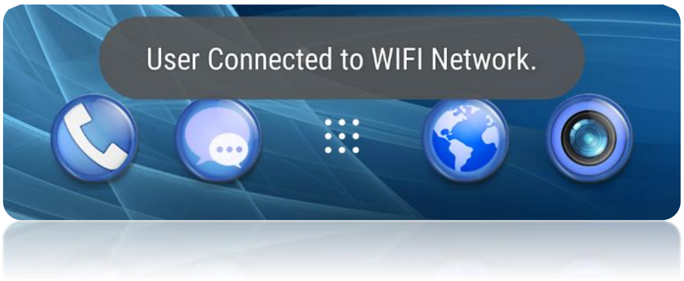
Monday 27/02/2017: I had my first meeting with my new supervisor today and explained what my project was and voiced my concerns about not knowing which direction to turn with regards of making progress with the project. One of the lads in the group made some suggestions as to what to look at, so that's the next step now amongst the fifty other things that needs to be done.

9.2.7 March 2017

March already and not long to go now, after my first meeting with new project supervisor I came away with some possible routes to research regarding moving the project forward.

Saturday 11/03/2017: I have been searching through Vogella, Tutorials Point, and Android Developers to try find information and determine what can be done to make the next step in the project. I found some information on broadcast receivers that I might be able to use to broadcast a state change in a device. I will continue to look at this and try to implement some code to test.

Monday 13/03/2017: I have been able to use a combination of code that I found in a couple of online tutorials to create a broadcast receiver that listens for a state of change in the devices WIFI connection and then broadcasts a TOAST message on that device to announce the state change. This is displayed in the figure below this text. I now need to find out if I can have a second device listen for the state change on the first device and receive a notification of the state change on the first device.



User Connected to WIFI Network.

Monday 20/03/2017: I had a meeting with my supervisor Paul Stynes today where we discussed any developments within the project and what further developments can be made. I have some ideas on what to do since I was able to implement code for TOAST notifications based on WIFI connection.

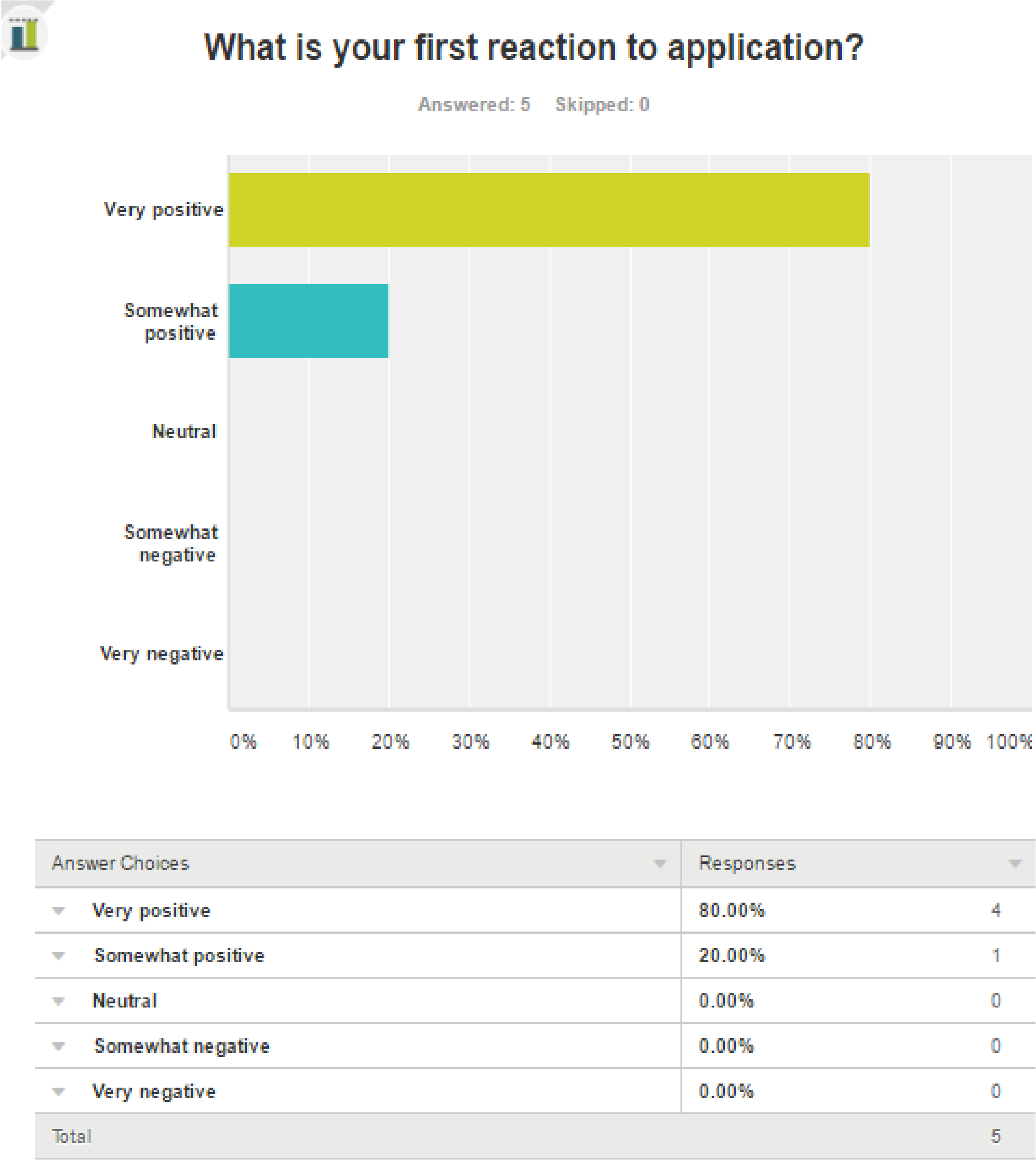
Saturday 25/03/2017: Things are finally starting to come together and the project seems to be falling into place, I have been working away on a new login and register system with Firebase, it seems a lot cleaner than the one I was using before and it's leading towards me being able to develop the actual messaging system of the app. I'm hoping once the messaging system is setup I will be able to modify and implement code to provide the core functionality of the app, which is the automated messages based on WIFI connections. We shall see!!

Thursday 30/03/2017: So, we're coming towards the end of the semester now, it has been such a busy year and now I'm off to study for exams then back to project work for the big finale.

9.3 Survey Results

These are the results of a survey consisting of ten questions aimed at five users who have prior knowledge of the scope of this application before taking this survey.

Question 1:

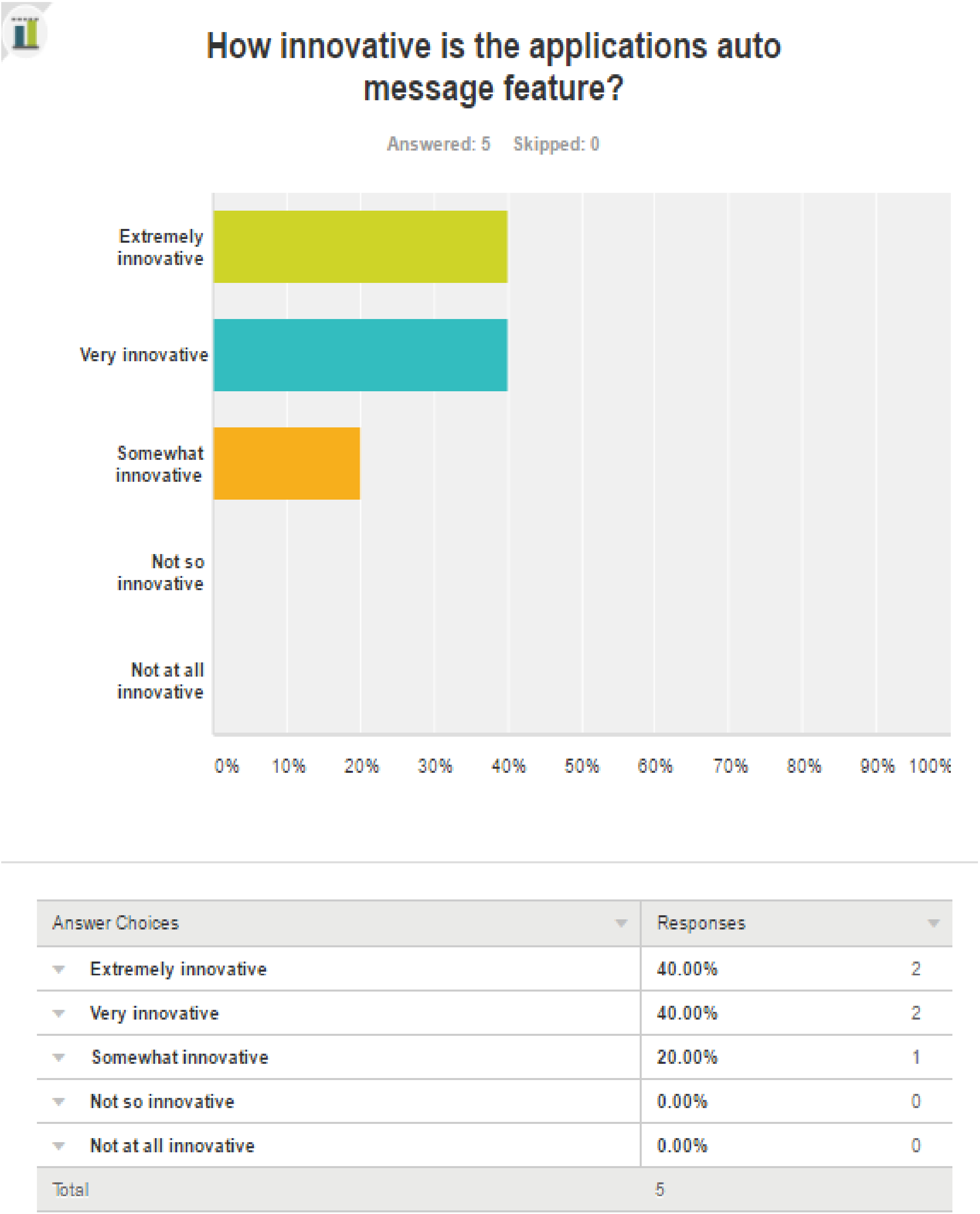


What is your first reaction to application?

Answered: 5   Skipped: 0

Answer Choices	Responses	
Very positive	80.00%	4
Somewhat positive	20.00%	1
Neutral	0.00%	0
Somewhat negative	0.00%	0
Very negative	0.00%	0
Total		5

Question 2:

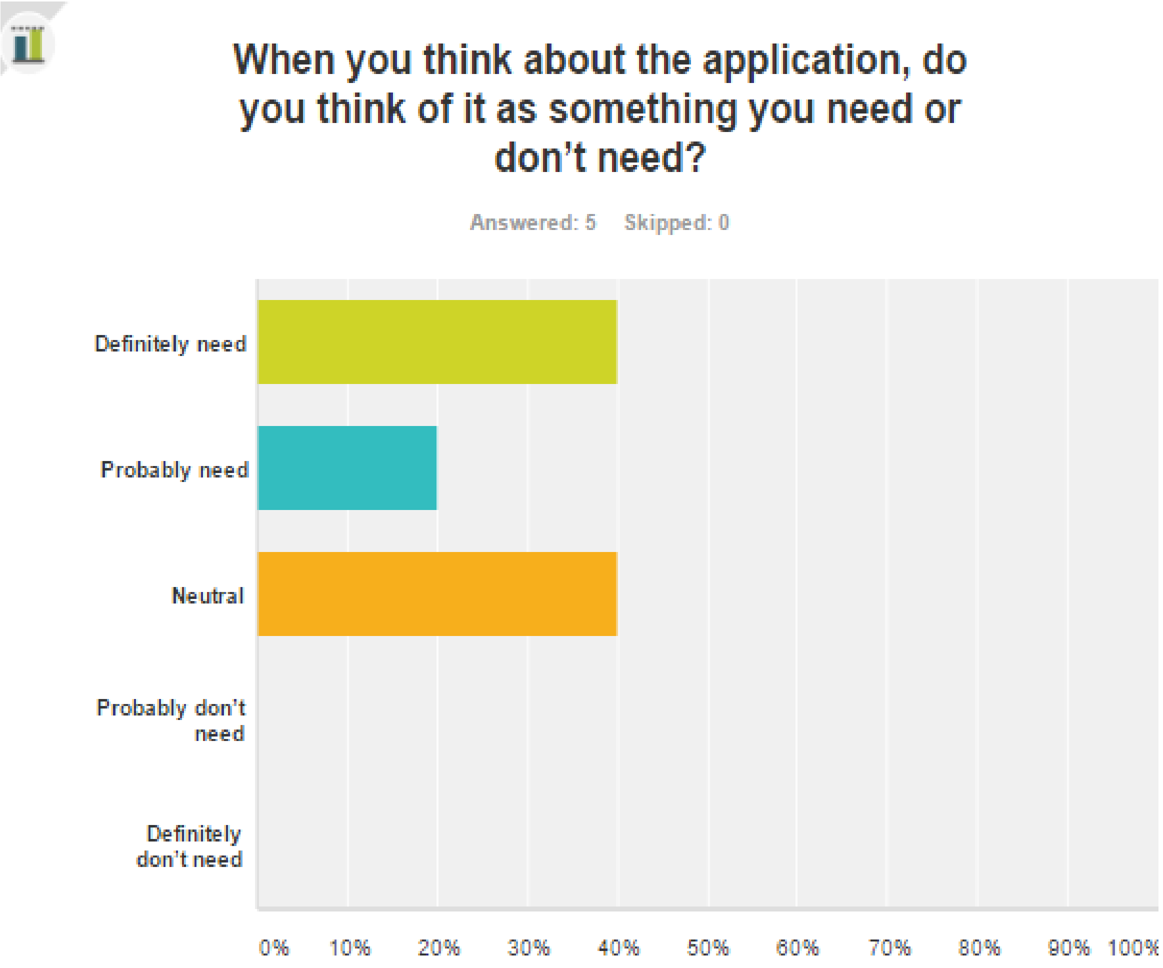


How innovative is the applications auto message feature?

Answered: 5 Skipped: 0

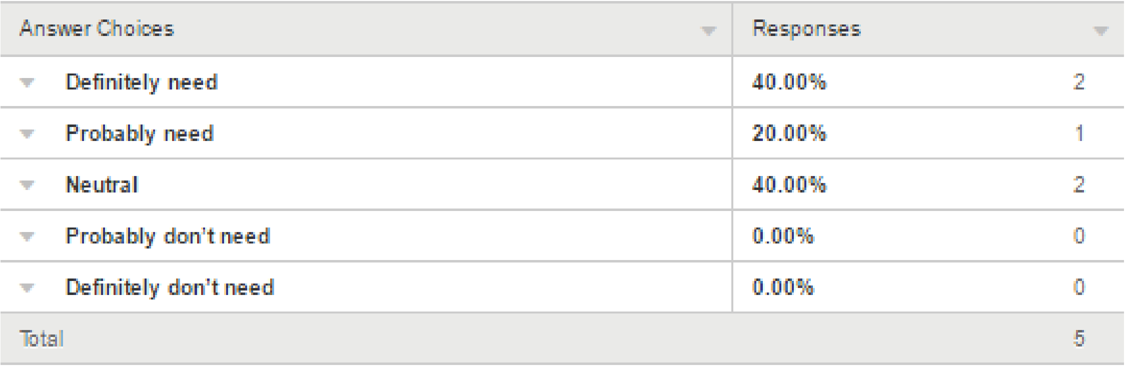
Answer Choices	Responses	
Extremely innovative	40.00%	2
Very innovative	40.00%	2
Somewhat innovative	20.00%	1
Not so innovative	0.00%	0
Not at all innovative	0.00%	0
Total	5	

Question 3:



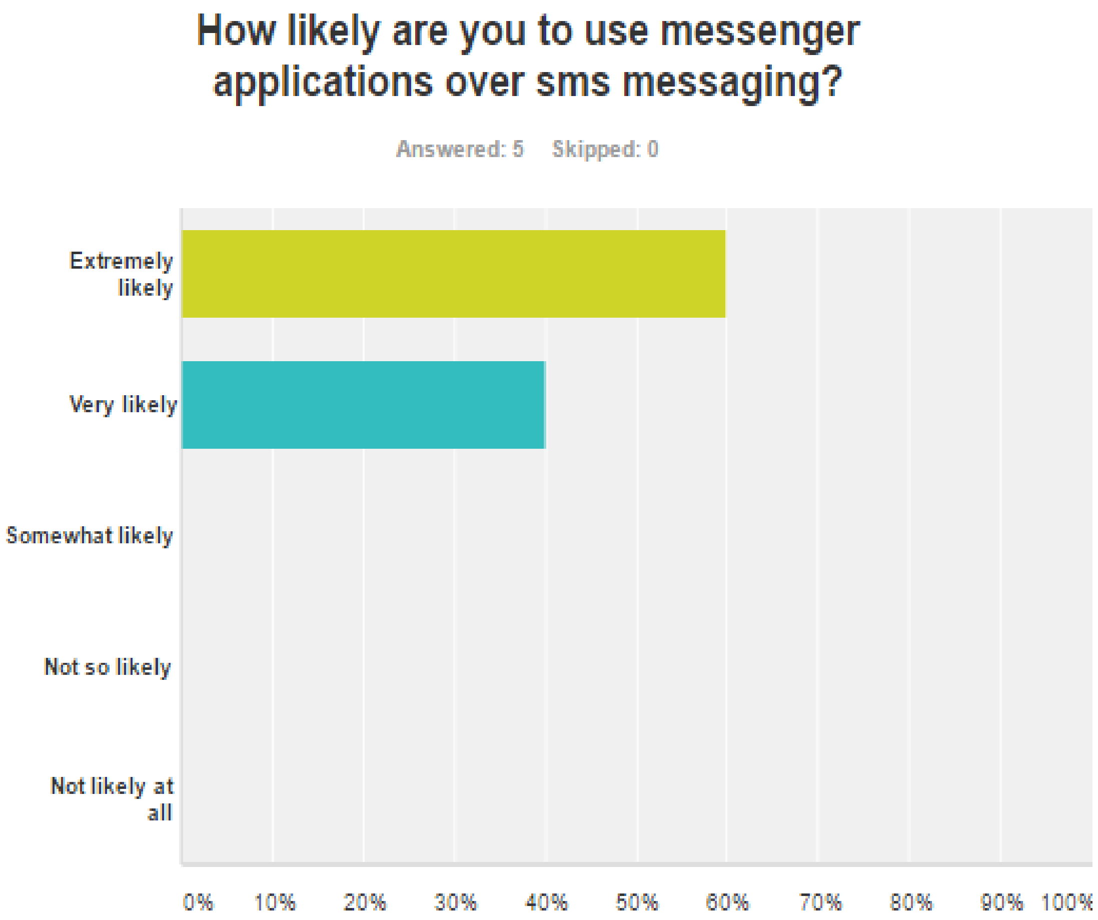
When you think about the application, do you think of it as something you need or don't need?

Answered: 5 Skipped: 0



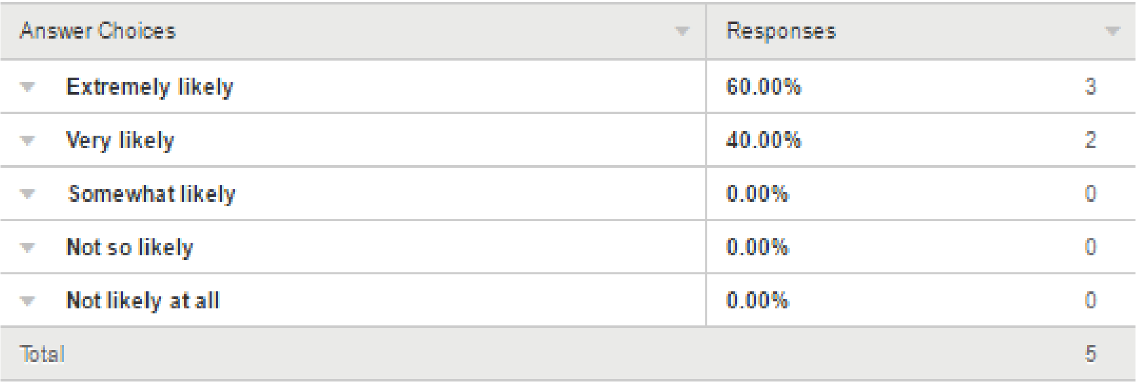
Answer Choices	Responses	
Definitely need	40.00%	2
Probably need	20.00%	1
Neutral	40.00%	2
Probably don't need	0.00%	0
Definitely don't need	0.00%	0
Total		5

Question 4:



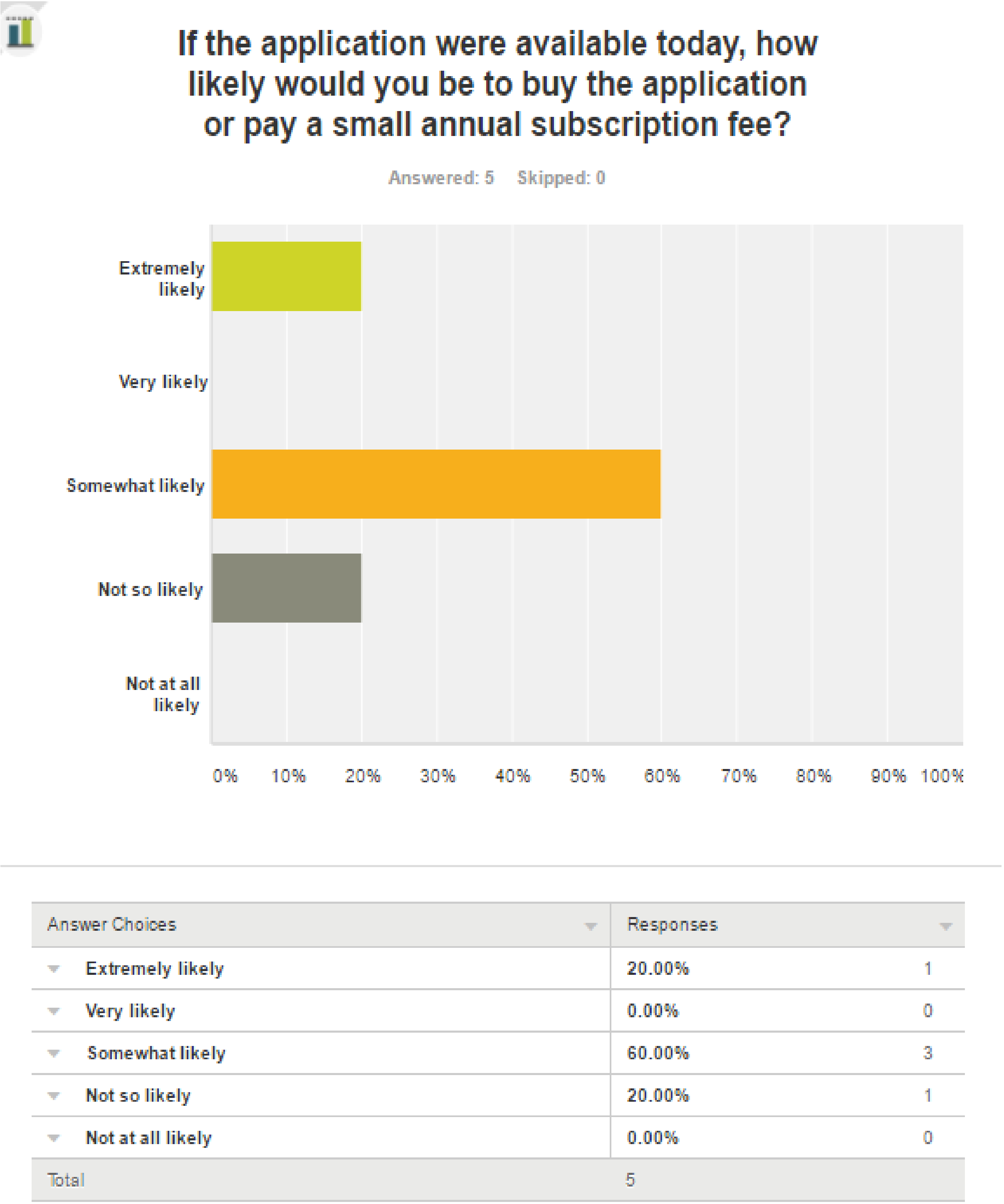
How likely are you to use messenger applications over sms messaging?

Answered: 5 Skipped: 0



Answer Choices	Responses	
Extremely likely	60.00%	3
Very likely	40.00%	2
Somewhat likely	0.00%	0
Not so likely	0.00%	0
Not likely at all	0.00%	0
Total		5

Question 5:

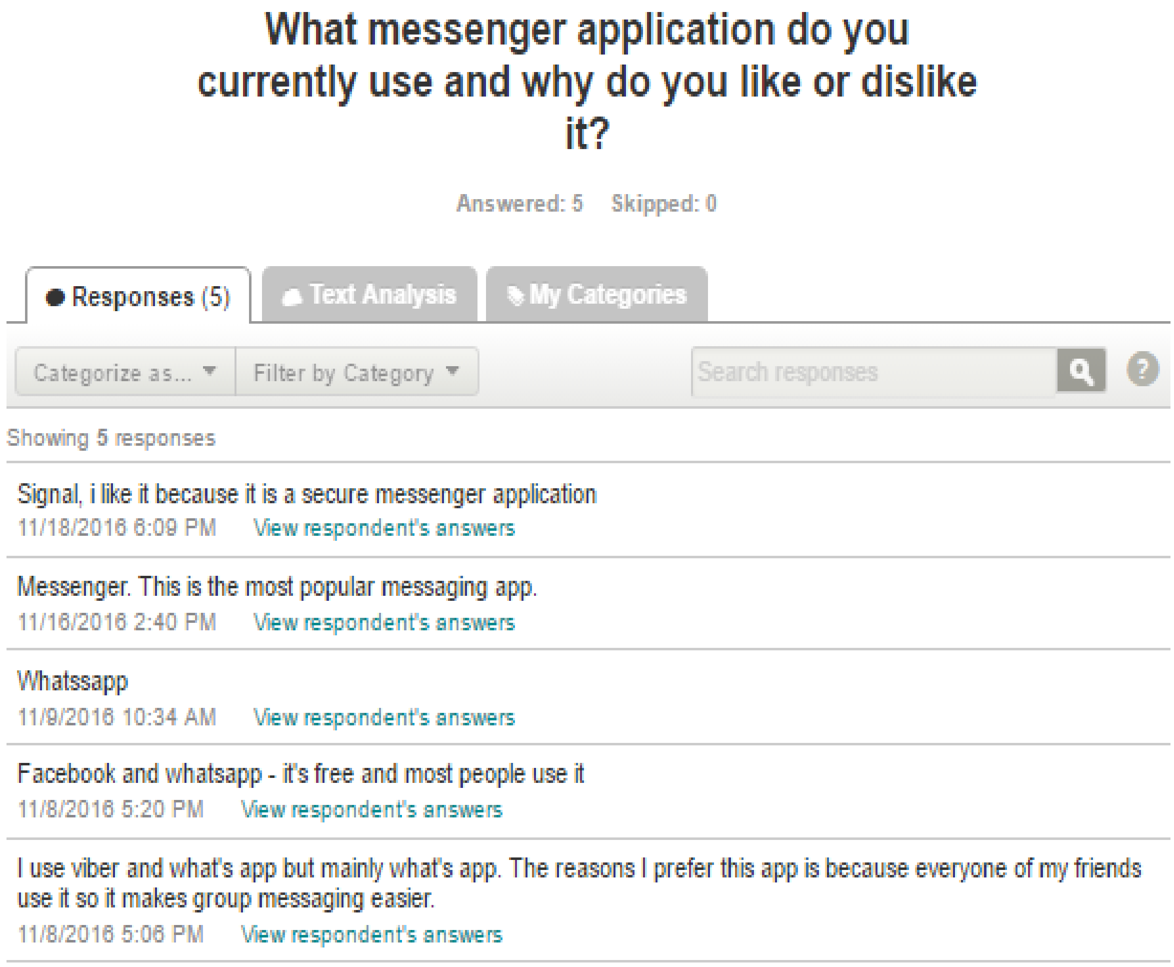


If the application were available today, how likely would you be to buy the application or pay a small annual subscription fee?

Answered: 5 Skipped: 0

Answer Choices	Responses	
Extremely likely	20.00%	1
Very likely	0.00%	0
Somewhat likely	60.00%	3
Not so likely	20.00%	1
Not at all likely	0.00%	0
Total	5	

Question 6:



What messenger application do you currently use and why do you like or dislike it?

Answered: 5 Skipped: 0

Showing 5 responses

Signal, i like it because it is a secure messenger application
11/18/2016 6:09 PM View respondent's answers

Messenger. This is the most popular messaging app.
11/16/2016 2:40 PM View respondent's answers

Whatssapp
11/9/2016 10:34 AM View respondent's answers

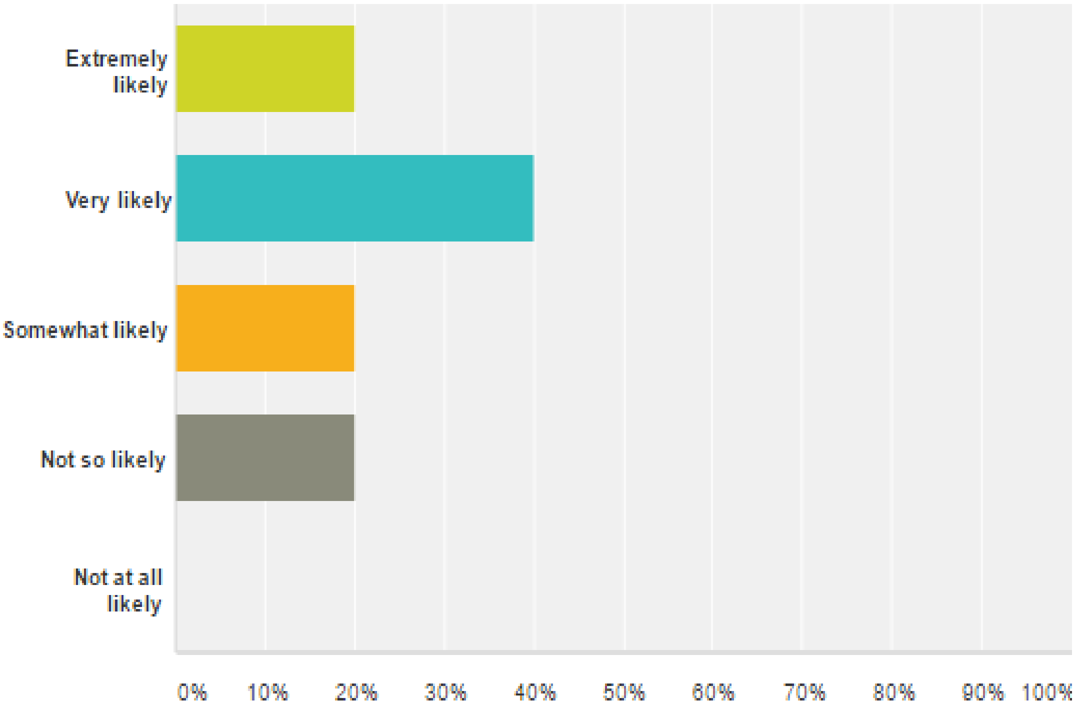
Facebook and whatsapp - it's free and most people use it
11/8/2016 5:20 PM View respondent's answers

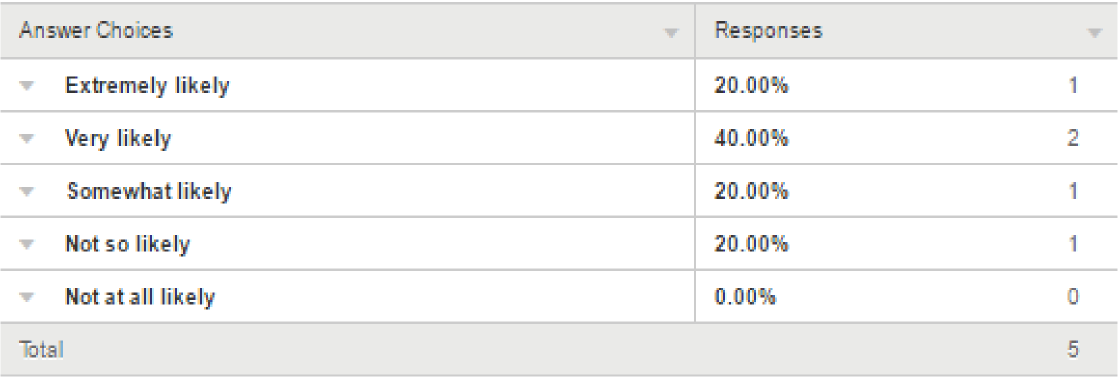
I use viber and what's app but mainly what's app. The reasons I prefer this app is because everyone of my friends use it so it makes group messaging easier.
11/8/2016 5:06 PM View respondent's answers

Question 7:

How likely are you to replace your current messenger application with this application?

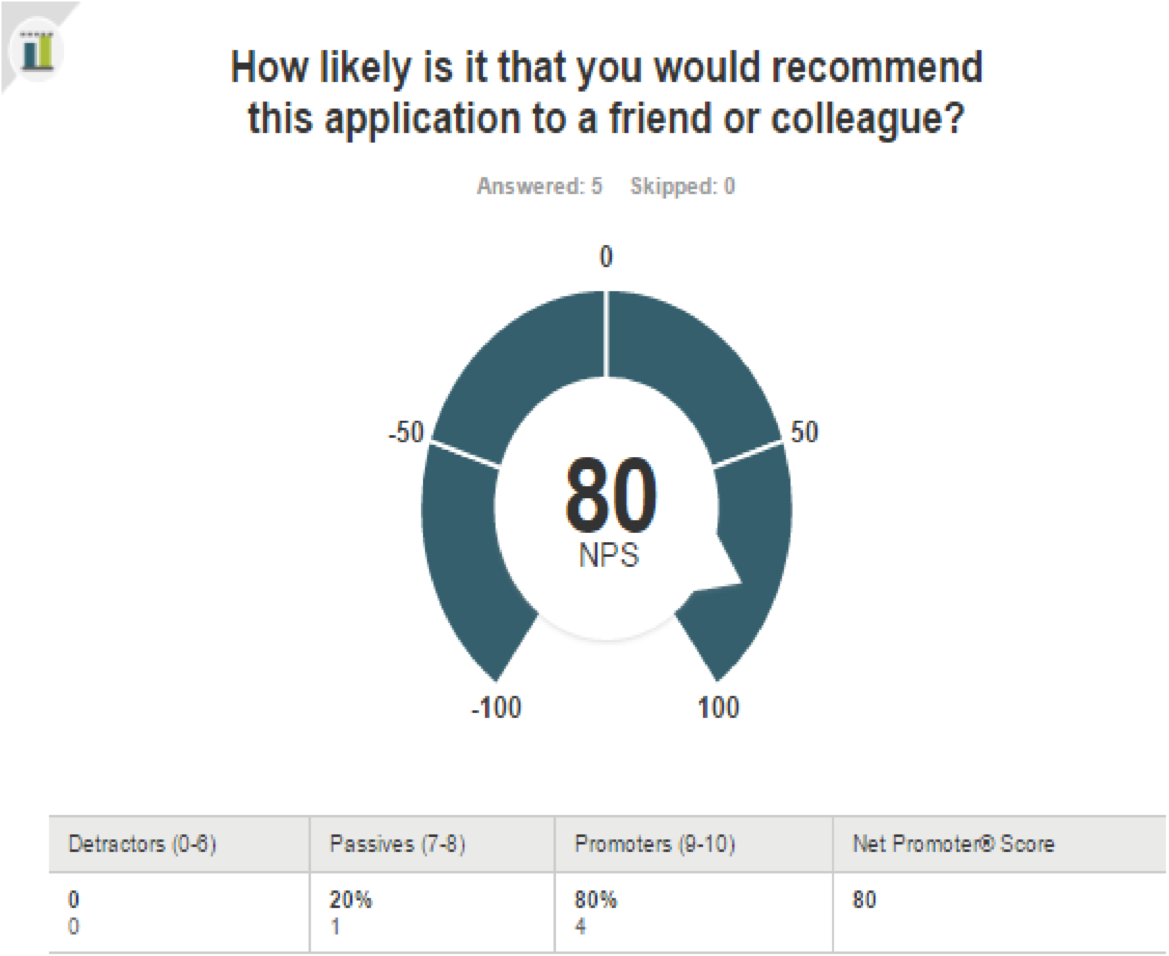
Answered: 5   Skipped: 0





Answer Choices	Responses	
Extremely likely	20.00%	1
Very likely	40.00%	2
Somewhat likely	20.00%	1
Not so likely	20.00%	1
Not at all likely	0.00%	0
Total		5

Question 8:

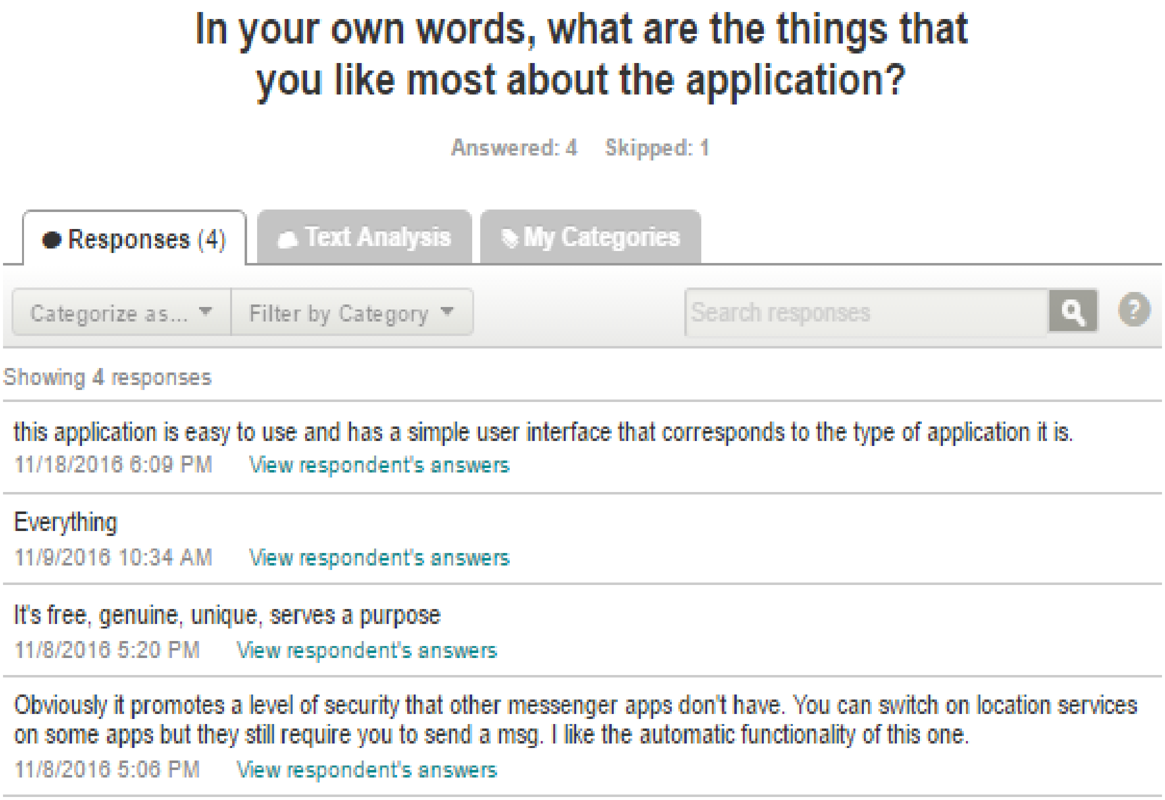


How likely is it that you would recommend this application to a friend or colleague?

Answered: 5 Skipped: 0

Detractors (0-6)	Passives (7-8)	Promoters (9-10)	Net Promoter® Score
0 0	20% 1	80% 4	80

Question 9:



In your own words, what are the things that you like most about the application?

Answered: 4 Skipped: 1

Showing 4 responses

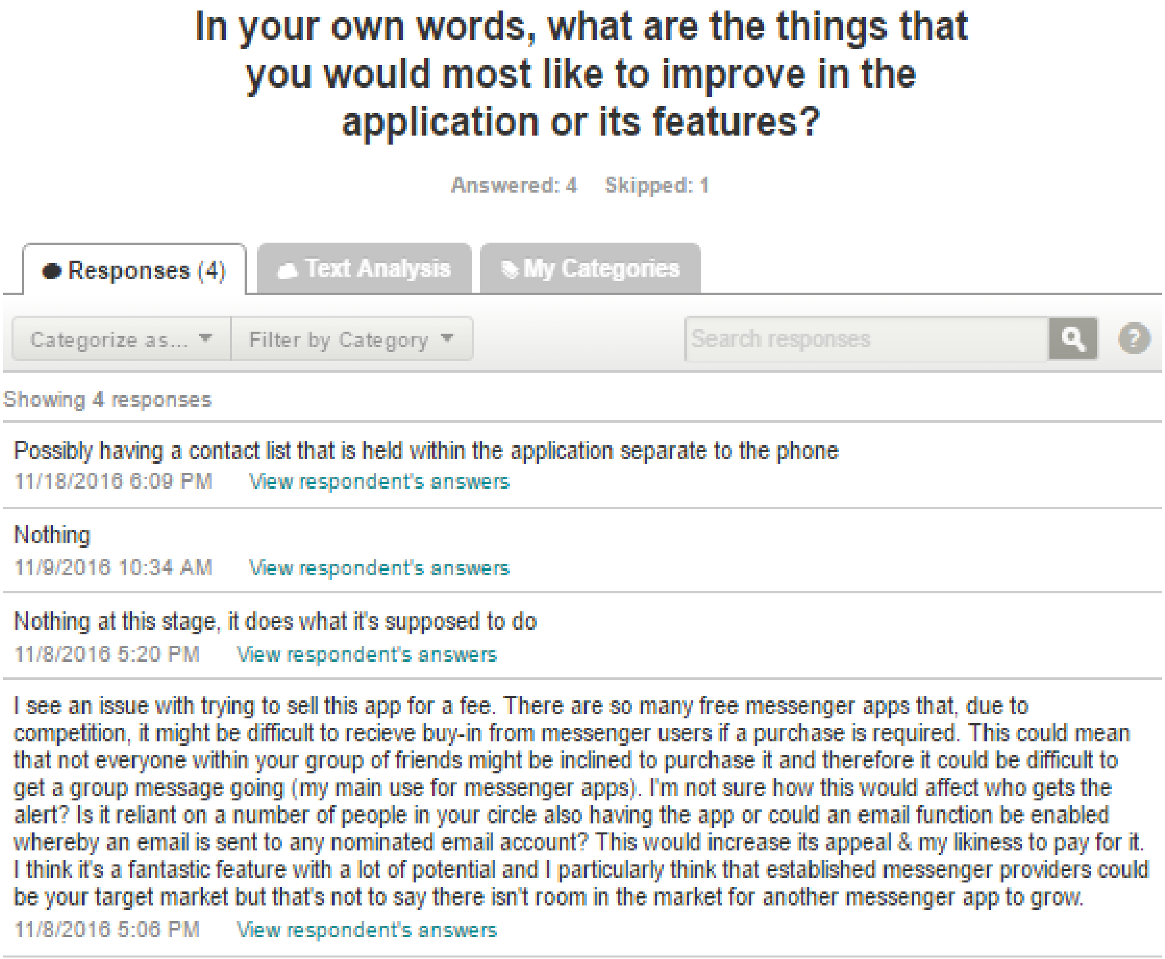
this application is easy to use and has a simple user interface that corresponds to the type of application it is.
11/18/2016 6:09 PM View respondent's answers

Everything
11/9/2016 10:34 AM View respondent's answers

It's free, genuine, unique, serves a purpose
11/8/2016 5:20 PM View respondent's answers

Obviously it promotes a level of security that other messenger apps don't have. You can switch on location services on some apps but they still require you to send a msg. I like the automatic functionality of this one.
11/8/2016 5:06 PM View respondent's answers

Question 10:



In your own words, what are the things that you would most like to improve in the application or its features?

Answered: 4 Skipped: 1

Showing 4 responses

Possibly having a contact list that is held within the application separate to the phone
11/18/2016 6:09 PM View respondent's answers

Nothing
11/9/2016 10:34 AM View respondent's answers

Nothing at this stage, it does what it's supposed to do
11/8/2016 5:20 PM View respondent's answers

I see an issue with trying to sell this app for a fee. There are so many free messenger apps that, due to competition, it might be difficult to recieve buy-in from messenger users if a purchase is required. This could mean that not everyone within your group of friends might be inclined to purchase it and therefore it could be difficult to get a group message going (my main use for messenger apps). I'm not sure how this would affect who gets the alert? Is it reliant on a number of people in your circle also having the app or could an email function be enabled whereby an email is sent to any nominated email account? This would increase its appeal & my likiness to pay for it. I think it's a fantastic feature with a lot of potential and I particularly think that established messenger providers could be your target market but that's not to say there isn't room in the market for another messenger app to grow.
11/8/2016 5:06 PM View respondent's answers