Requirements Document

Taylor Yarn, Alex Upright

February 2019

1 Introduction

The application called 'TextTraQ' will be an application developed in Android Studio. Its general purpose will be to alert contacts about certain information that has been specified by the user via text-messages while they are driving.

2 High Level Description

TextTraQ itself will have five main different activity screens, the log in screen activity, the home screen activity, settings activity screen, contacts activity screen and the favorites activity. The app itself will also be able to generate contact settings screens whenever a contact is pressed on so that the user may change their settings for that user.

The log in screen will simply contain two text inputs, one that is used for a user name and one that is used for a password, a log in button, and finally a create an account button. If one inputs a valid user name and password they will be directed to the home screen, if one chooses to create an account a screen will pop up asking them for their desired user name and password, after which they will be asked to log in with those credentials. If they do not have an account they will be able to press a button labeled 'Create Account' that will bring them to a page where they can create an account to log in with

The Create Account page will have 3 text input boxes and 1 button. The 3 input text boxes will be labeled, 'username' 'password' 'confirm password' respectively. The button will be labeled 'Create Account'. If a user tries to create an account with a username that already exists an appropriate error message will be shown. Likewise if an account is attempted to be created without the two password text boxes matching an appropriate error message will also be shown.

The home screen will act as the first activity that the user sees and will interact with after logging into the application. The home screen will include buttons to navigate to the other three activities as well as a view that is connected to Google Maps to search locations and a button to start the overall application. The view that is connected to the Google Maps application will be meant for putting in locations, this will act as a place for the user to specify where they are travelling to and a location must be entered before pressing the start button. Below this view will be the button that starts the overall application. After a user has input their destination and pressed the start button, the application will start running in the background so that even if the user closes the application it will continue running. Starting the application will send text messages to the users specified contacts at an interval that the user has specified.

The settings screen will have multiple elements of which to select and change. This screen will act as a way to change the default settings of any contacts that are added into the list of people that the user wishes to contact when starting their journey. There will be a box specifying the default time between sending the automated text messages, a switch specify whether or not the text should include the users location, a switch specifying whether or not to send the users speed, a switch specifying whether or not to send the current estimated time of arrival of the user, and finally a custom message that can be sent with every text.

The contacts screen will contain a list of the contacts that have been added to the users list of people to contact when they start this application. Each contact will contain their picture which will be pulled from the phones contacts, their name, and their number in each view for the specific contact. The view containing the contact will also contain their personal settings, which are the same as the settings described above and can be changed to personalize what is sent to each contact. This screen will be scroll able downwards so that there can be a large number of contacts listed on this page, there will also be a small space towards the top of the page that contains a 'plus' symbol. When this symbol is pressed the user will be asked to add a contact from their list of contacts on their phone, to which they can either choose a contact or click out of the screen. If they choose to choose a contact, that contact from their phone will be added to the list of contacts on TextTraQ with the default settings. A contact can also be deleted or put into the favorites page from the contacts screen as well. If a contact is swiped to the left the contact will be deleted from this page, while if the contact is swiped to the right it will be added to the favorites page. Furthermore if a contact is pressed down upon for an extended period of time the user may move the order of that contact, possibly to the top or bottom of the page or in between other contacts.

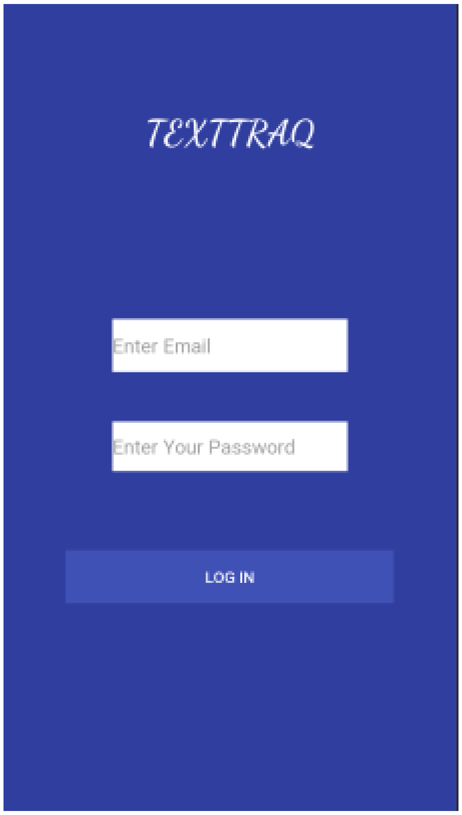
The favorites screen will contain the picture and name of a person you have added to your favorites from your contacts. Below the picture and name there are two buttons, one to add the selected favorite to an existing alarm, another to create a new alarm for the favorite. At the bottom of the the screen are two buttons that can be used to swap through the contacts you have added to the favorite menu, when you reach the end of the favorites, you will be brought back to the first contact in the favorites list. There will be a "Previous" button that takes you through the list in the reverse order.

The contact settings screen will be very similar to the settings screen, however at the top of the screen it will contain the name and number of the contact whose settings are being changed.

3 Detailed Requirements

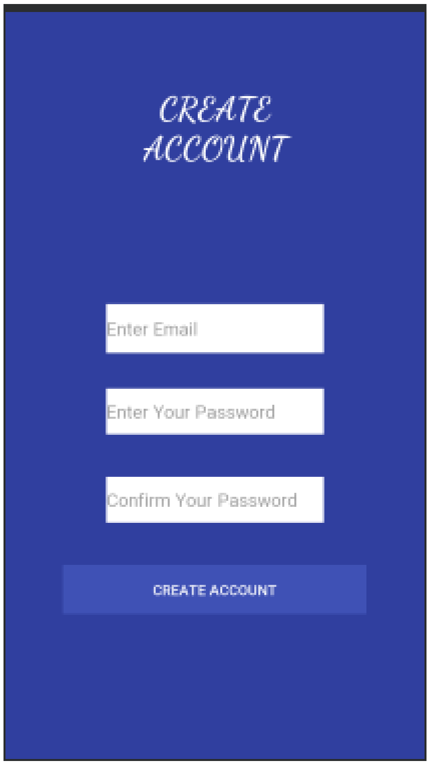
3.1 User

3.1.1 Log In Screen



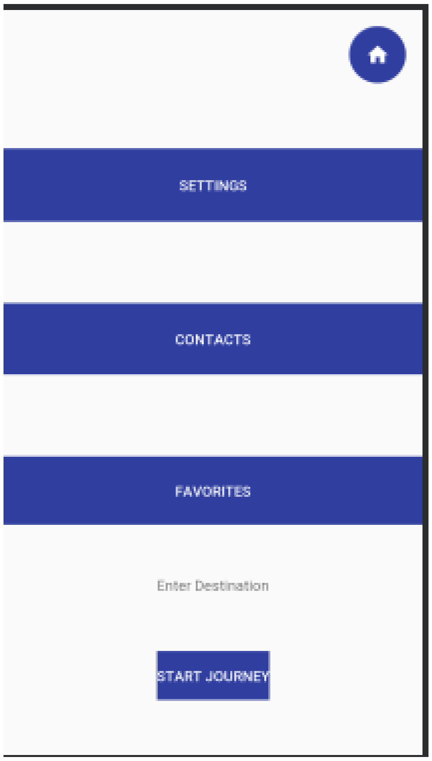
The log in screen as shown above has one text input, 1 password input, and a log in button. When a user inputs a valid email and password and presses the 'Log In' button the user will be brought to the home screen. If the email or password are incorrect an error message will appear and ask for the user to enter a valid email and password combination.

3.2 Create New Account Screen



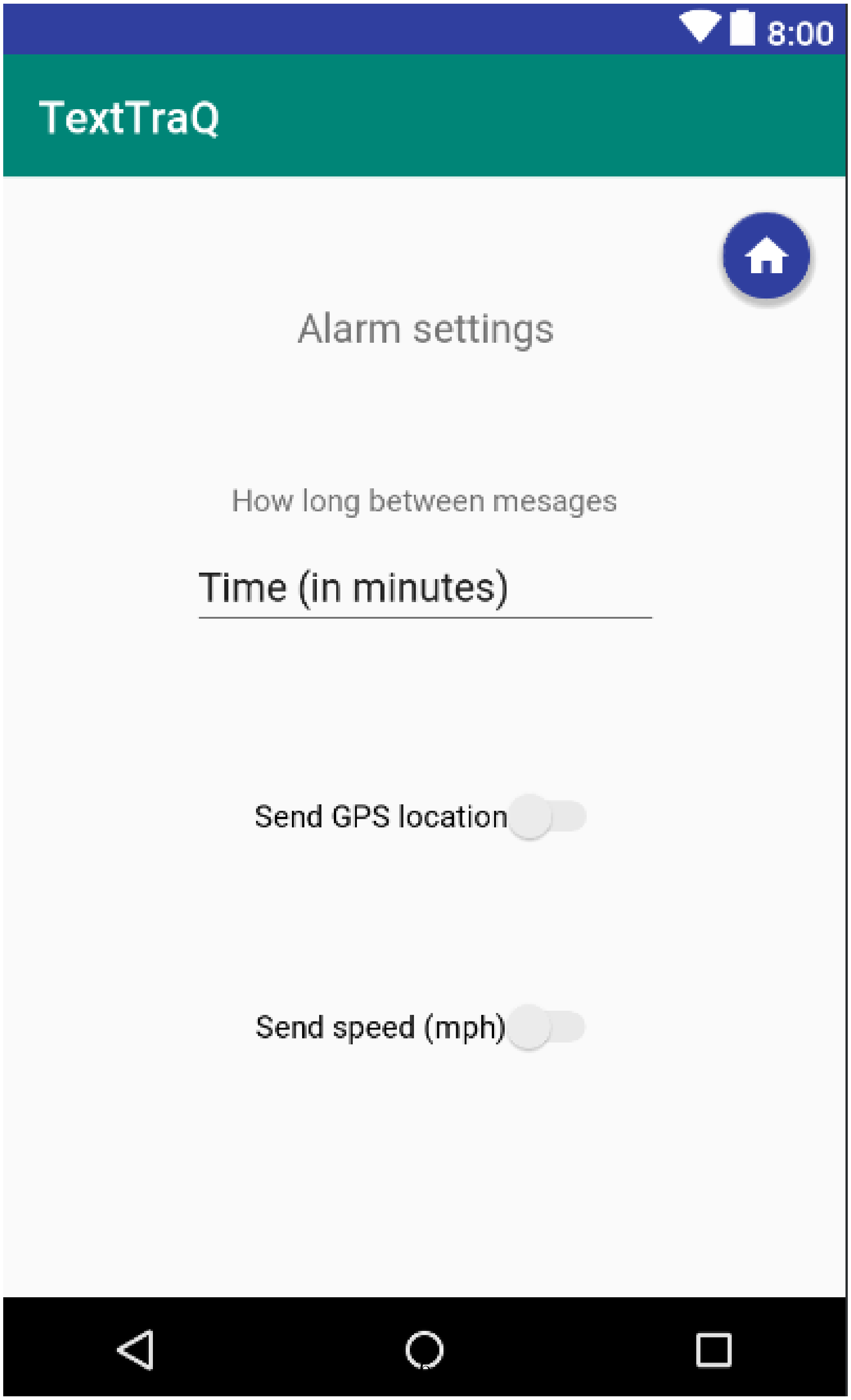
The Create New Account Screen as shown above has 1 text input for the username, and 2 password inputs for the user to enter and confirm their password. Once all 3 have been filled and the Create Account button has been pressed the database will be queried to see if the username exists already, if it does an error will be shown, otherwise the account will be created.

3.2.1 Home Screen



As shown above the home screen will have multiple buttons, there will be a button in the top left corner that will reload the home page, it will be present on every page. There is a settings button that will bring you to the settings page when pressed, likewise for the contacts and favorites screen. There is a text box that will be connected to Google Maps as explained above, and finally there is the start journey button. The start journey button will start the activity in the background so that even if it is closed the automated text messages to your contacts will continue.

3.2.2 Setting Screen



Alarm settings

How long between mesages

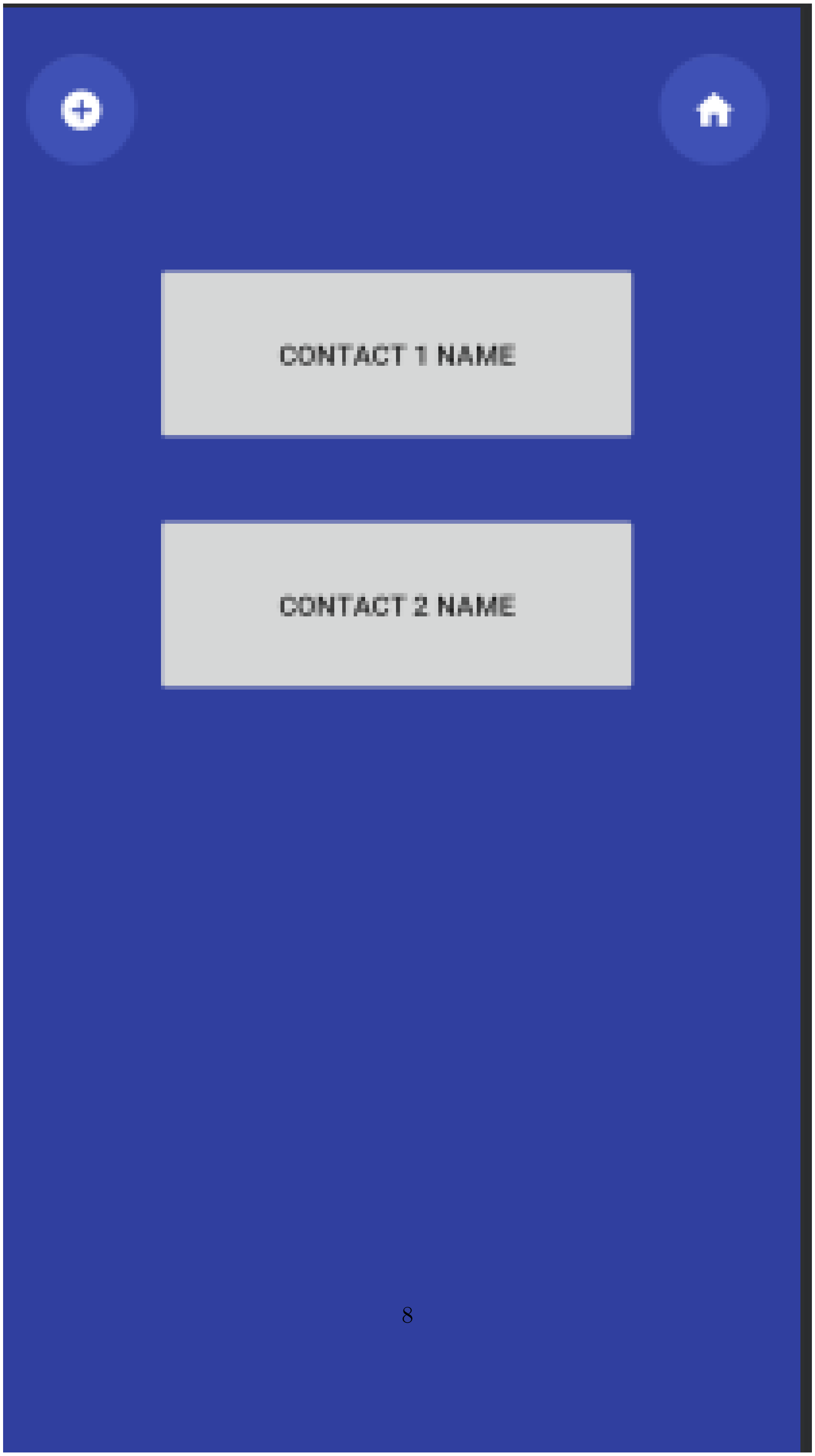
Time (in minutes)

Send GPS location

Send speed (mph)

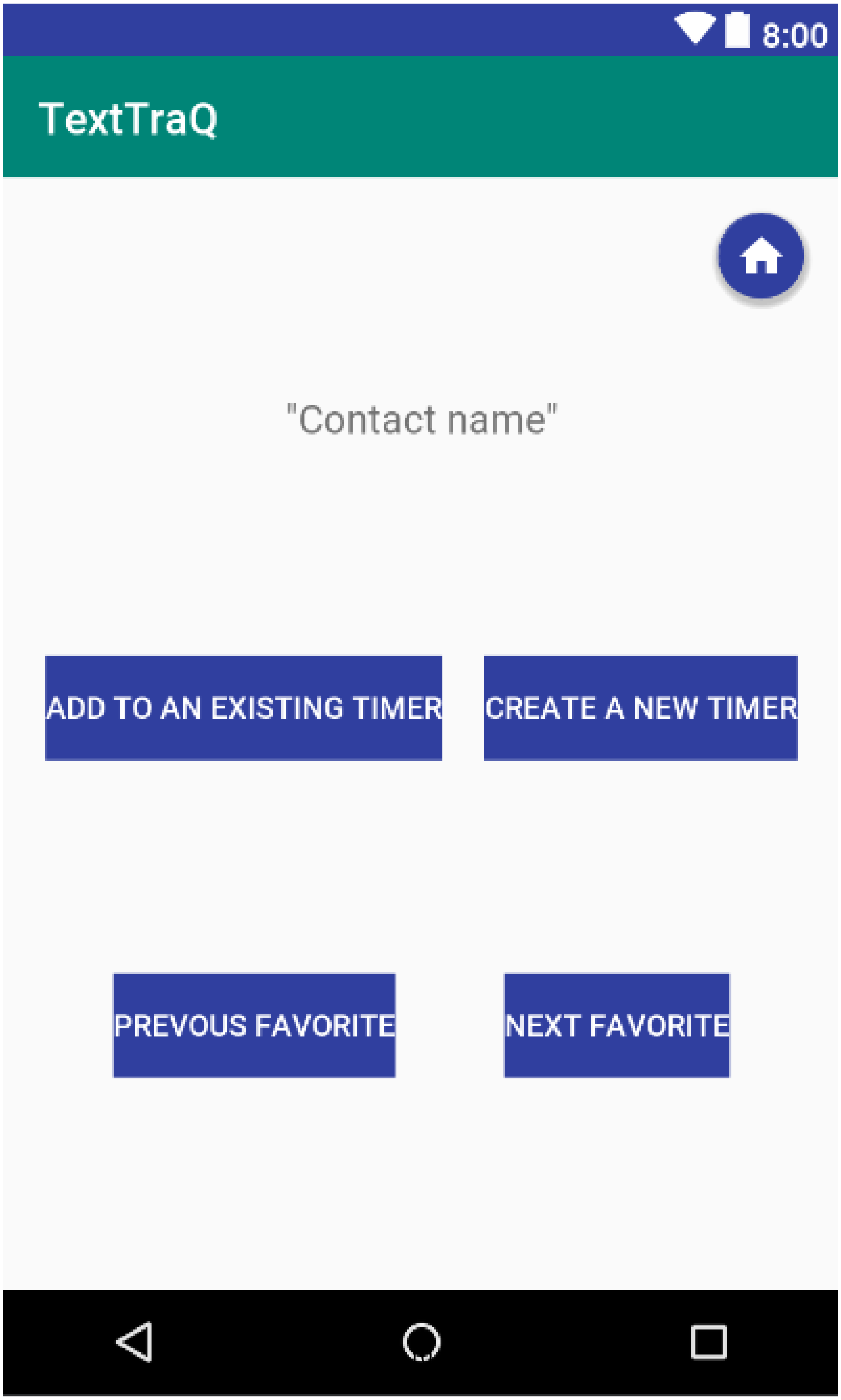
As displayed above, the settings screen will have a box allowing you to set the time between sending the automated text messages, a switch specify whether or not the text should include the users location, a switch specifying whether or not to send the users speed, a switch specifying whether or not to send the current estimated time of arrival of the user, and finally a custom message that can be sent with every text.

3.2.3 Contacts Screen



As shown above, the contacts page will contain two floating buttons at the top, and any number of buttons after that. The home button in the top right of the screen will bring the user back to the home screen of the application. The second floating action button on the top left of the screen will prompt the user to add another contact from their phones list of contacts to this applications list of contacts. After this there will be buttons for each contact added, shown are two buttons who hold the names of 'Contact 1 Name' and 'Contact 2 Name', if another contact, say 'Contact 3 Name' was added a button would appear with that name shown, this view will be scroll able so if there are many contacts added you will be able to see them all. If a contacts button is pressed you will be brought to their personal settings page which will be shown below.

3.2.4 Favorites Screen



The favorites screen displays, as shown above, the name of the contact with four buttons, two to each line. The first line of buttons relate to timers, the left most adding the contact to a timer that you have already made, the right most takes you to the settings where you can select new settings for the contact. At the bottom there are two buttons, the left takes you to the previous favorite while the right button takes you to the next favorite.

3.2.5 Contact Settings Screen

3.3 System

This application requires an Android Operating system that has an API level of 15, more specifically this app requires that the device be running on Android 4.0.3(IceCreamSandwhich) or higher. This application requires permissions to access the users Contacts, Location, SMS and the application Google Maps.

3.4 Hardware

This application requires that the phones internet connection work as the application will occasionally be accessing a server side database. It also requires that the phones location services work as well as it is able to send SMS's

3.5 Network

This application will be using TCP to send any packets over the network it is connected to.

4 Data Format

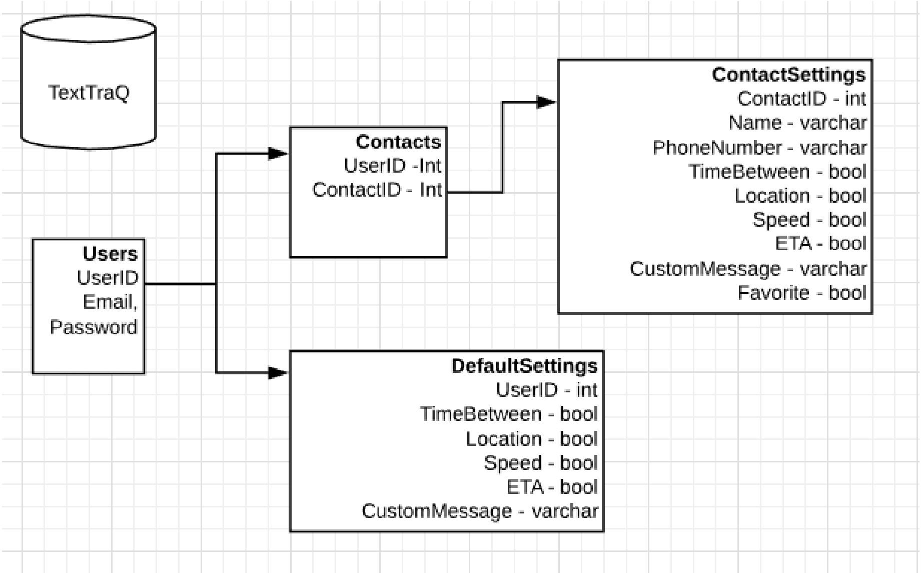
Attributes that will be stored in the database:

UserID, Email, Password, ContactID, Name, PhoneNumber, TimeBetween, Location(bool), Speed(bool), ETA(bool), CustomMessage, DefaultID.

Functional Dependencies:

UserID – Email, Password, ContactID, DefaultTimeBetween, DefaultLocation, DefaultSpeed, DefaultETA, DefaultCustomMessage

ContactID – Name, PhoneNumber, TimeBetween, Location, Speed, ETA, CustomMessage, Favorite



5 Constraints

5.1 Performance

5.2 Reliability

5.3 Security

5.4 Time

5.5 Resources

6 Functional Requirements

6.1 Business Flow

Upon opining the app for the first time the user will first see the log in screen, after successfully entering in your email and password you will be taken to the home screen. While on the home screen the user will be able to navigate, using buttons to the settings, contacts, favorites, and they will be able to start their journey. If the user presses the settings button they will be taken to the setting screen, while in the settings screen they will be able to press the home button in the upper right hand corner of the screen to be brought back to the home screen. The contacts screen will also have a home button in the upper right hand corner of the screen is addition to having an option to add a contact to the favorites menu. In the Contacts screen the user will have the choice of pressing the home button in the upper right hand corner to be brought back to the home screen.

6.2 Use Cases

Case 1: If a user opens the application, the log in screen will be the first thing that shows up, if the user then enters the correct log in information that is contained in our database, the user will be redirected to the home page. If the user enters the incorrect log in information the user will be given an error. This is to test that the log in page works correctly.

Case 2: If a user is on the home screen, and the user presses the button with the name 'Settings', the user will be brought to the settings page, likewise with the 'Contacts' and 'Favorites buttons. This is to test that the buttons bring the user to the correct pages

Case 3: While the user is on the home page, if he starts typing into the text box, Google maps will prompt him with places similar to the text the user is typing in giving suggestions on his actual destination. This is to test that Google Maps is integrated within the application correctly.

Case 4: If the user has not entered a location into the text box, and presses the begin journey button, he will be given an error message. If the user has entered a valid location and presses the begin journey button, the application will start running in the background and sending messages to each contact correctly so even if the user gets off the application it will still perform its function . This is to test that the general application works correctly in the background

Case 5: If the user is on the settings page, and changes the default time to say 10 minutes from the previous 5 minute default, and the user then goes to the contacts page and adds a contact, that contact will show up with a default contact time of 10 minutes in between each automated message. This is to show that the correct data is being pushed to the database when we change the settings page and that the correct data is being grabbed from the database afterwards.

Case 6: If the user is on the Contacts screen and presses add contact, a pop up list of contacts with a search bar from the users contacts on their phone will pop up and the user can choose one, after choosing that contact will then show up in the contacts page on the application.

Case 7: If the user is on the contacts page and swipes right on a contact, that contact will remain on the contacts page, but if you go to the favorites page that contact will also be available on the favorites page. This is to test that our swiping right function works.

Case 8: If the user is on the contacts page and swipes left, that contact will be deleted from the contacts page. This is to test that our swiping left function works correctly.

Case 9: if the user holds on the contact photo in favorites, then then cycles through the favorites list, the contact will no longer be visible. This is to test that the hold function works.

Case 10: If the user enters an invalid time into the interval section, the app will display an error message asking the user to use a valid time.

Case 11: If the user is on the favorites screen and presses the next button, they will be shown the next contact that is a favorite if there is one, otherwise nothing will happen, likewise if they press the previous button they will be shown the previous favorite contact if there is one.

Case 12: If the user is on the create account screen and enters a username that is already being used, they will be shown an error message.

Case 13: If the user is on the create account screen and enters a valid username but two different passwords, they will be shwon an appropriate error message.

Case 14: If the user is on the create account screen and enters a valid username and passwords that match, the users account will be created, the user will then be able to log in with that account.

7 Schedule

March 14th DataBase Completed, Log In Page is functional, i.e if you log in with correct credentials you will be redirected to the home page, else if you put in the incorrect credentials you will be shown an error message. Create New Account page functional, i.e if you create a new account with valid credentials it will be put in the database correctly, if you try and create an account with a username that is already there you will be given an error.

March 21st Home Page is complete and will bring you to the correct pages depending on which button that you click, Text Box connecting to Google Maps when looking up location is complete

March 28th Settings Page is complete and will change the default settings in the database correctly.

April 4th Contacts Page is complete, you will be able to add contacts to this page from your contacts on your phone as well as if you click on a contact it will bring you to their personal settings page where if you change their settings their settings in the database will change.

April 11th Favorites Page is complete(Alex Describe)

April 18th When the Start Journey button is pressed it will start the application in the background and send out messages to contacts specified every one of their time intervals

8 Grading Rubric

Comments: 20 % Functions and overall Code is well commented. Each function should have Pre and Post conditions explaining what the function does. There should also be comments throughout the code explaining import aspects or making the code more readable.

Functionality: 30 % Code works correctly without errors

Style: 10% The page of the application looks as it was intended too.

Test Cases/Use Cases 15% The test cases/use cases for each deadline test all aspects of the application to make sure that they work correctly.

DataBase: 15% The page of the application works correctly with the database

On Time: 10% The application page or code was completed by the time signified on this document.