An-Najah National University
Faculty Of Engineering
Department Of Computer Engineering

Software Graduation Project (Guardian - A)

Mohammed Domidi
Amjad Salhab

Supervisor : Dr. Mona Domidi

Presented in partial fulfillment of the requirements for Bachelor degree in computer engineering.

2020-2021

Acknowledgement

We would like to express our deepest appreciation to our families and friends for their huge support during the project.We would also like to extend our deepest gratitude to Guardian-A team for giving us the opportunity to work on such a great idea and for letting us to be guardians to help people to live a better life .Our success would not have been possible without the support of Dr. Mona Domidi , thanks wont be enough for her valuable advice and helpful contributions .

Yours sincerely
Mohammed ,
Amjad.

DISCLAIMER

This report was written by student(s) at the Computer Engineering Department, Faculty of Engineering, An-Najah National University. It has not been altered or corrected, other than editorial corrections, as a result of assessment and it may contain language as well as content errors. The views expressed in it together with any outcomes and recommendations are solely those of the student(s). An-Najah National University accepts no responsibility or liability for the consequences of this report being used for a purpose other than the purpose for which it was commissioned.

Contents

List of Figures

List of Tables

Abstract

This project is about the implementation of Guardian-A application for Guardian-A company which is located in the United Kingdom . Guardian-A is a mobile application that helps girls to live a better life , by providing features to prevent sexual harassment facing them in their daily life.

The implementation process of the project was challenging , starting from planning until delivering the final product . We managed to use **Ionic** which is a cross platform programming framework in order to implement the interface and handle some back-end functionality in addition to many other developing tools and API's such as **Google maps API** , **Firebase** and **Firestore**.

Finally , we were able to deliver a well structured , good looking mobile application prototype that meets all functional and non-functional requirements needed by Guardian-A team .

1. Introduction

Guardian-A is a mobile application that helps girls to live a better life , by providing features to prevent sexual harassment facing them in their daily life. Applications like Guardian-A can play a significance role in changing the life of girls living in dangerous areas or societies .

In order to be able to do the previous , the application must contain features such as maps to chose the safest path while traveling , reporting systems to make reporting much easier , chatting system with special notification feature and some other useful features .

The main objective of the project is facing a worldwide problem which is sexual harassment using technology by providing a mobile application to prevent , expose and report such incidents .

Unfortunately , this idea cant be applied to any country or region due to the lack of information about places and incidents related to sexual harassment . Luckily , we were able to apply our work inside the United Kingdom by using the help of google maps API and UK police API which we managed to integrate in our application .

This report is organized as follows, section two will discuss some of our constraints during the project , section three will discuss the related work to our main topic , section four will discuss Experimental procedure in our report , section five will discuss the results and analysis and section sex will talk about our conclusion .

2. Constraints,Standards/Codes and Earlier course work

2.1 Constraints

First of all , working on **Guardian-A** project - which is the winner of the most impactful idea from Global Innovation Catalyst at Stanford University - in general was challenging to us .We had to be ready to meet their demands and expectation during the whole process. On another hand , working with new frame works and API's was also challenging and interesting at the same time .

Figure 2.1: GIC most impactful idea award 2020

2.2 Standards/Codes

Regarding to software engineering standards used in this project , we managed to go with **Agile**[1] approach in order to keep the Guardian-A team involved with us during the whole process.Starting from structuring the application until the end phases which was very helpful for us to meet their expectations and to fast respond to changes.

2.3 Earlier course work

Preparing the project was one of the most interesting phases , we managed to look for new frameworks which can help us to accomplish our goal. This step helped us to save time learning these frame works and gave us a good idea about the available frame works and techniques used in mobile application development[2] .

3. Literature Review

Sexual harassment is one of the biggest problems facing people world wide .Efforts to stop sexual harassment never ends but unfortunately , as Lynn Bowes-Sperry and Anne M. O'Leary-Kelly mentioned in their article [3] that most targets do not report their experiences for many reasons related to culture and society , which encouraged us to come up with a simple way to report such incidents.

As Nasalski Ignacy mentioned in his article about sexual harassment in arab world , we can see that sexual harassment is rooted in arab society just like any other society in the world .Which shows the importance of applications like Guardian-A to be available for all people around the world.

According to Guardian-A team in UK , 85k women experience rape every year and 85% of rapes go unreported .which lead us to start the work in the UK to help preventing , reporting and exposing such incidents.

Similar projects have been done before , some of them still available today such as Life360 and Safe the city . Guardian-A application provides a combination between these two applications with much more helpful features added.

Features	Guardian-A	Life360	Safe & the city
Personalised group services	**yes**	**yes**	**yes**
Live geolocation safety features	**yes**	**yes**	**yes**
Crime reporting	**yes**	**no**	**no**
List of emergency phone numbers	**yes**	**no**	**yes**
Legal information pages	**yes**	**no**	**no**
NHS information pages	**yes**	**no**	**no**
user safety prioritised	**yes**	**no**	**no**
sexual safety oriented	**yes**	**no**	**yes**

Table 3.1: Guardian-A Vs similar projects.

4. Methodology

The purpose of this chapter is to summarize the developments that took place within the Guardian-A project and put them in a larger scientific and technological context.

4.1 Frameworks

Before diving into the details of our project , we must make it clear about frameworks and technologies used. Starting with **Ionic** which is a cross platform framework that we used to design our hybrid application using **Progressive Web Application** approach . This approach progressively enhance web applications to look and feel like native applications by providing reliability and offline functionality. Along side to **Ionic** , we managed to use **Cordova** building tool in order to build both ios and android native applications .

4.2 Servers and API‘s

Regarding servers , we managed to use a **Server-less** architecture approach in our design and implementation in the back-end for many reasons. Where server-less mobile apps offer the same benefits as building a typical web application with a server-less back-end which are Scalability , Less overhead , Quick updates and the ability to run code anywhere.

In order to insure a fast and reliable application , we managed to use **Firebase** and **Firestore** services. Firebase helped us to design and implement real time dynamic data systems such as chatting system and real time notifications. Firestore helped us to store different resources using online storage.

Regarding API‘s , we managed to work with **Google Maps Java script API** which is considered the best navigation service available . Documentation provided by that API was great , it was well organized and easy to use. Next , we managed to integrate **UK police API** in order to work on the safest path algorithm.

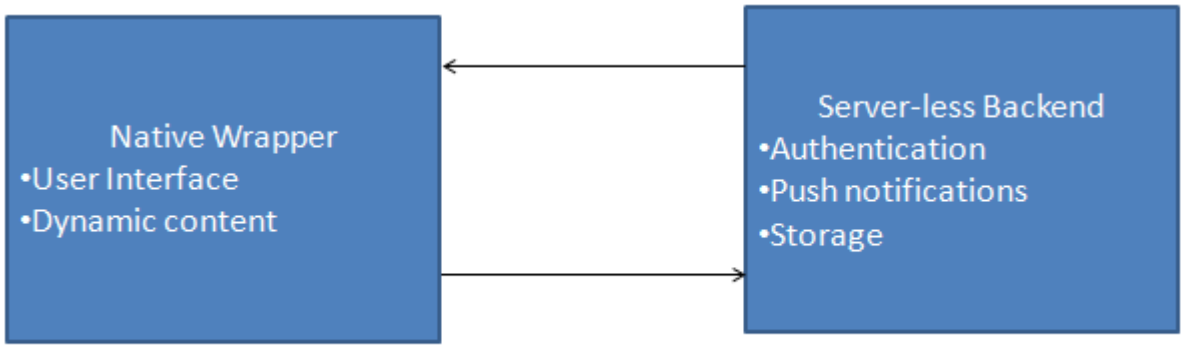

Figure 4.1: Server-less Hybrid application architecture .

4.3 Project Structure

At the first stage of work , we started planning our application structure by deciding pages and components needed to complete each section.The following figure shows the project plan that we started working with.

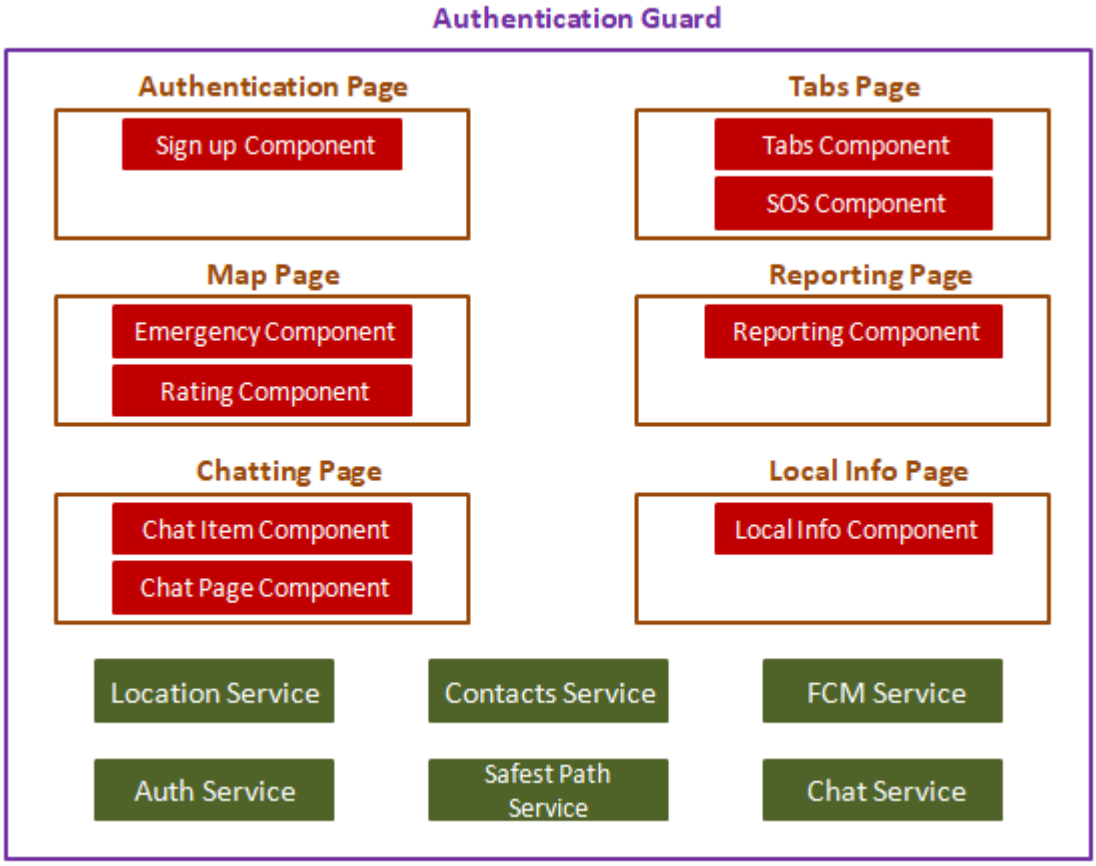

Figure 4.2: Guardian-A project structure .

4.4 Nonfunctional Requirements

The main non-functional requirements of our Application:

1. ***Availability :*** *Guardian-A can be Downloaded and used by anyone at any time .*
2. ***Acceptability :*** *Guardian-A is as easy and clear as possible to use. Also, we worked on the interface in a way that it would be simple, comfortable for the eye, without much extra-details in a page to avoid confusions .*
3. ***Security :*** *we made sure that all passwords are encrypted,taking all advantages of security features provided by* ***Firebase*** *authentication services .*
4. ***Portability :*** *we ensured that Guardian-A can work on all platforms without causing any troubles .*

4.5 Functional Requirements

The following table shows some of the main functional requirements of Guardian-A :

Tab	Feature
Sign In	**Sign up (First name, last name, phone number)**
	Sign in using phone number
	Sign in validation (wrong number, wrong passord .. etc)
	Reset password and recieve a random verification code via sms
	Logout from gaurdian-A mobile app
Reporting Tab	**Submit reports using an in app browser which is going to host a link to the police online reporting website**
Local Info Tab	**See details about police, hospitals, medical centers and NGOs near once the user click on the local info tab. The search results appears in a list of places which appear as a slider at the bottom of the page. Each place will be displayed with its own icon , name and location. After clicking on a place, it will place a marker on the map. A small toggle button is placed on the top right corner of the map , after clicking on it the searching mode will be switched from geolocation search into a search bar search , the same functionality described above will be applied to both modes.**
Map Tab	**Apply Guradian- A template to the map in terms of colors**
	Intergrate Google API
	Integrate Police Api (https://data.police.uk/docs/)
	Implement Safest Route Algorithm
	Implement Rating Module in which users could rate a specific location on the map and add the crime types.
Chat Tab	**Users can send and receive messages such as texts , photos , files , audios and videos**
	Users recieve realtime notification
	The app will ask the user to sync contacts from the phone in order to get access to users who have a Guardian-A account
	User can create a group of contacts
	User can add settings to each chat section.
	User can mute/ unmute chat group or contact
SOS button	**Notify by a text and immediately share location with emergency contacts which are set in the setting feature. Emergency contacts will receive a text sms and an app notification.**

Table 4.1: Guardian-A Functional Requirements

4.6 Class Diagram and GRASP principles

In the following section were going to show some basic functionalities in our application using ***Class Diagram*** *.After that , were going to discuss some* ***GRASP*** *principles applied to our project.*

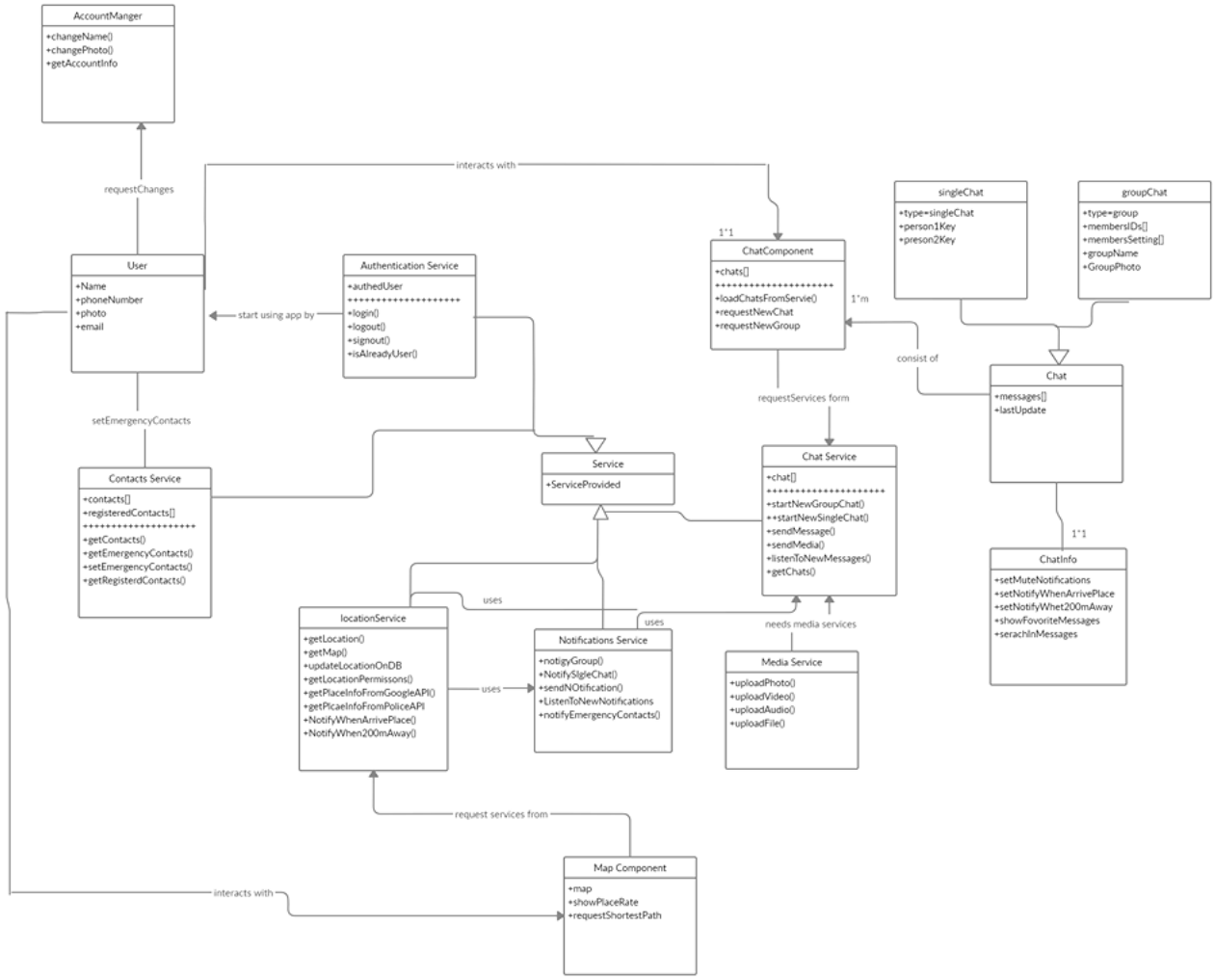

Figure 4.3: System class diagram .

1. **Information expert :** For local info class that has the information and settings of the chat ,as the class has this information it can modify settings and information for the chat.

2. **Creator :** For chat service as it contains array of chats ,so it should be the creator of new chats.

3. **Controller :** Both chat,map components considered as controllers because they capture the interaction of users and delegate the events to the services classes.

4. **Polymorphism :** Services divided into several types based on the type of type that can provide,Chat also divided into two types :single and group chat to achieve high cohesion and avoid checking the type of each chat .

5. **Pure Fabrication :** Instead of assigning the chat service the responsibility of saving media to the storage ,a new service added (media service)to be used to save media to storage ,this will increase the cohesion and re-usability.

6. **Indirection :** Most services act as intermediate classes between classes and workers,like notifications it behaves like intermediate service between other services and the system of sending notifications.

7. **Protected variation :** we have two external system on our app:google maps API,police API, So to protect our system from effects of variation of this API's the logic of reacting with this two systems located on single separate service and every class need service will call the separate service and not use the API directly .

5. Results and Discussion

In this chapter , were going to show and discuss the final results of our project. This chapter will be divided into sections were each section will discuss a feature or more provided by Guardian-A application.

5.1 Log in / Sign up page

This page is one of the first pages designed in our application . We managed to stick to Guardian-A brand and logo in order to keep the simple and good looking style for the interface. This page was designed using **Ionic** components in addition to some pure html and css codes .

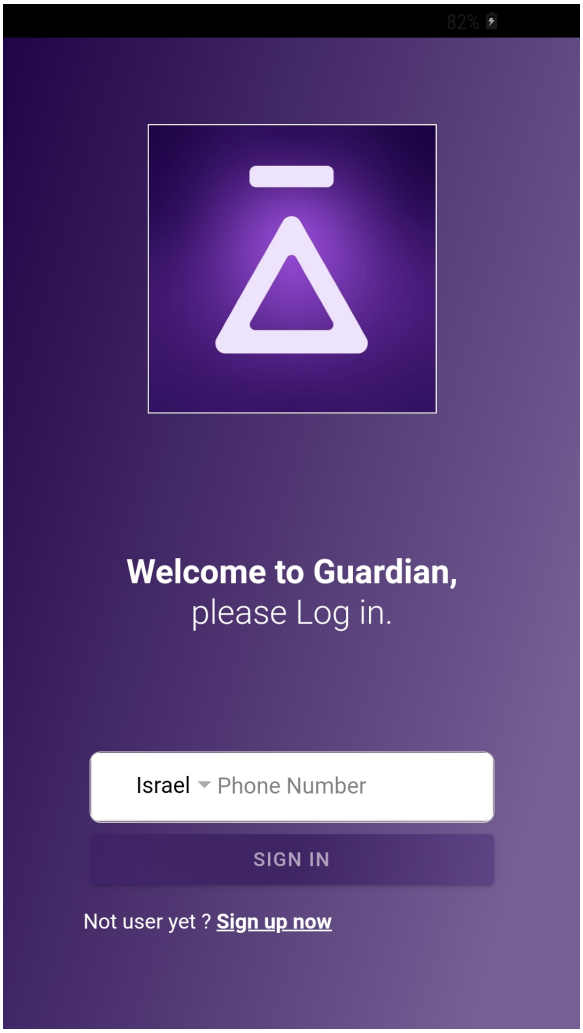

Figure 5.1: Guardian-A Log in page .

As we can see in order to log in, user must enter his phone number and wait to receive a random code in order to sign in into his account .This feature was implemented using Firebase authentication service which uses both FCM and Location services provided in our application .

Next we have the sign up functionality , which we implemented as a modal with multi stage process , each stage is stored separately until the final stage when the user is asked to enter and verify a valid phone number to link it to his account .

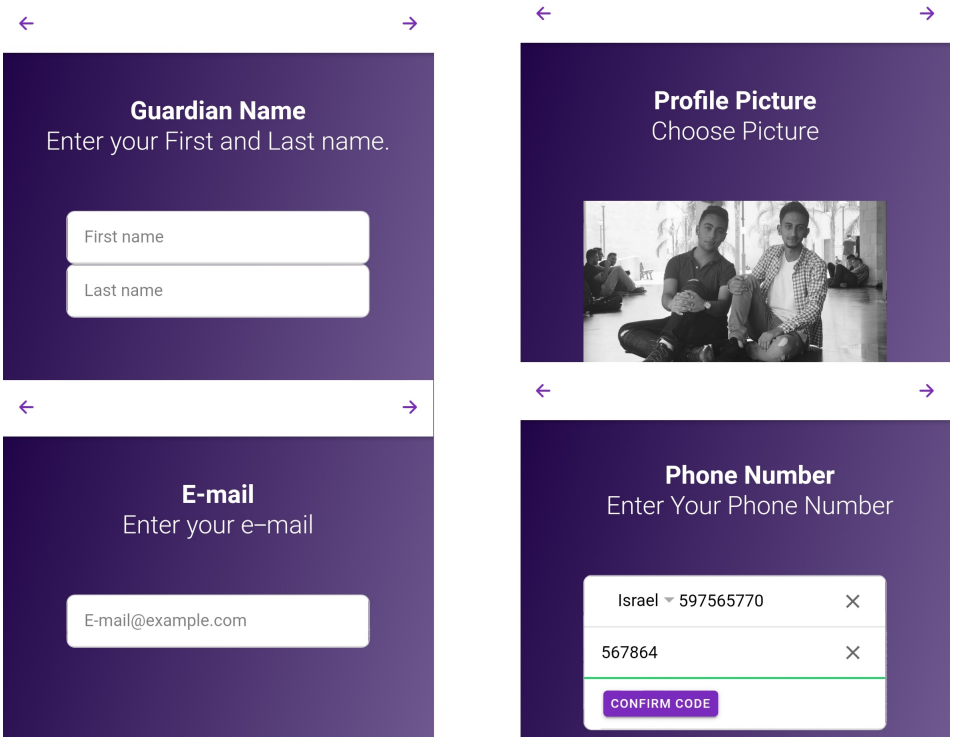

Figure 5.2: Guardian-A Sign up page .

5.2 Reporting page

This is one of the most important pages in our application , which allows users to report any incident as a witness or a victim easily by using an in app browser that holds Metropolitan Police online reporting option .

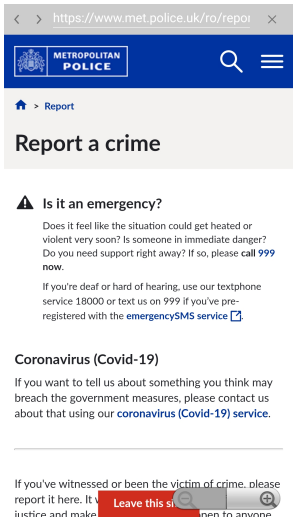

Figure 5.3: Guardian-A Reporting page .

5.3 Local info Page

The following page provides one of the most useful features in our application , users can use this page to search for every hospital, clinic and police station within the UK. Users should get a list of these that are nearest to him, with the top one being the closest, the second the second-closest etc.There should also be a search bar at the top of that screen where user would be able to input an address that is different to where he currently is so that he can search for local info at a different location. All these features were implemented using Google map Places and Location services .

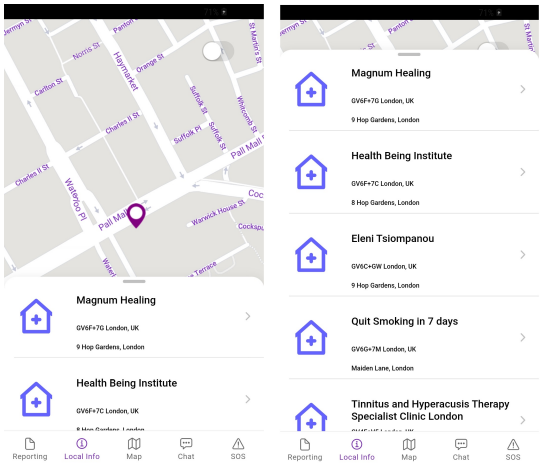

Figure 5.4: Guardian-A Local info page using Geo-location .

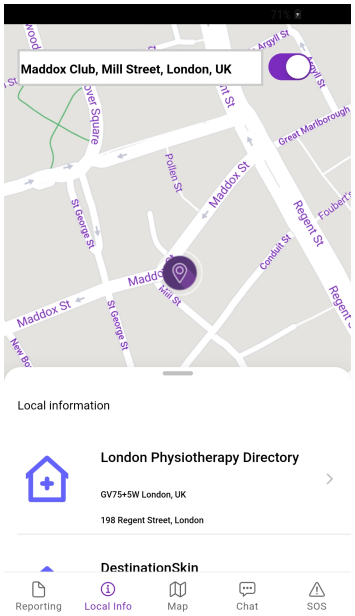

Figure 5.5: Guardian-A Local info page using Input address .

5.4 Map page

This page is considered the main one in our application , after a user log in he will be directed into the map page as a starting page , which will hold a map covering the screen with a marker to point to the user Geo-location . We can also see a search bar and an avatar at the top part of the map , this search bar is used to search for a destination which is restricted to the UK and biased to his location , an auto complete service is attached to this search bar. The avatar shown next to it is considered as the trigger to the user menu which holds the account settings , emergency numbers and log out options. A toggle button is used to switch the searching mode into source and destination mode and a navigation button is used to relocate user to his Geo-location when pressed .

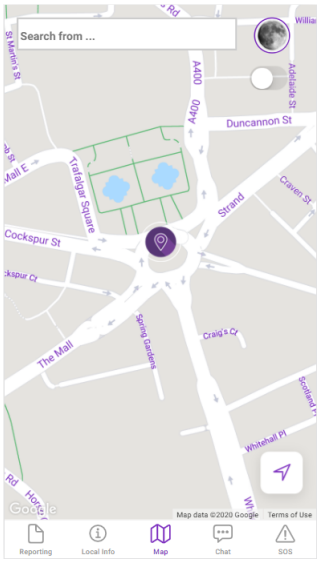

Figure 5.6: Guardian-A Map page .

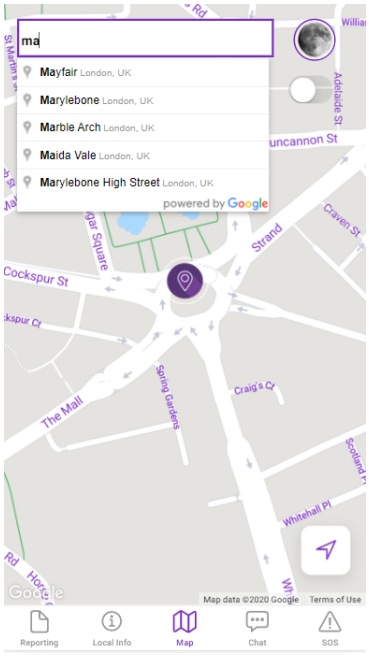

Figure 5.7: Guardian-A Map page Search bar auto complete.

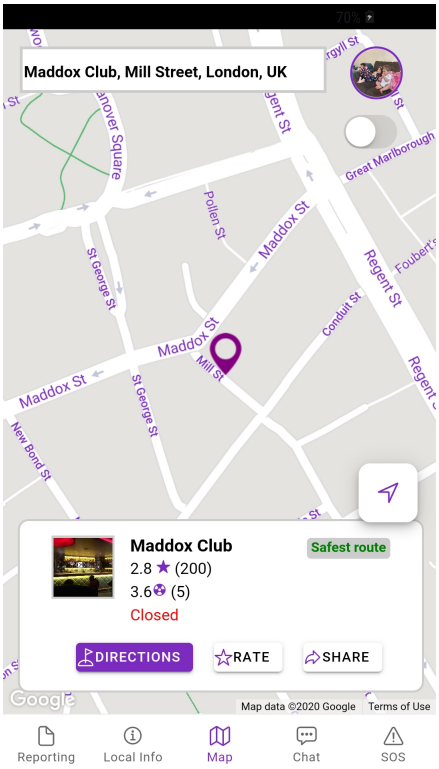

Figure 5.8: Guardian-A Map page Search process .

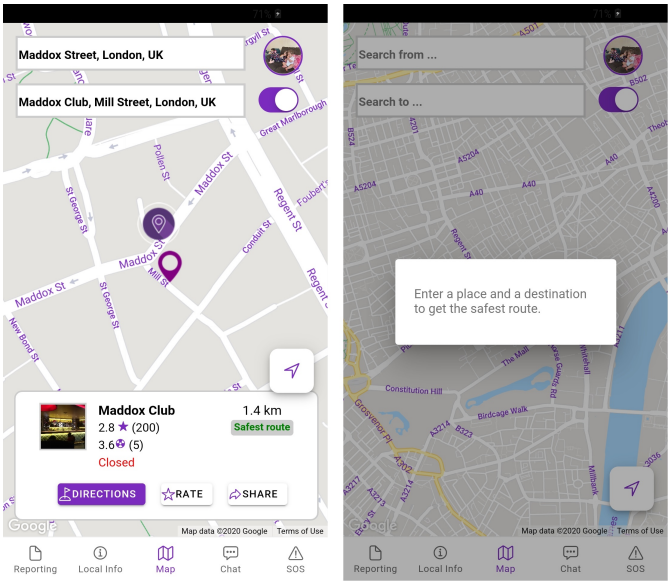

Figure 5.9: Guardian-A Map page source and destination mode.

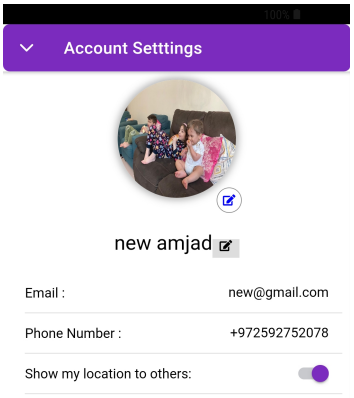

Figure 5.10: Guardian-A Account settings modal .

After searching for a place , a chard will pop up at the bottom of the screen , holding place info with three options button , the first button will calculate the safest route between source and destination the second one will open a rating module to rate the place and the third one will share the place into a group chat .

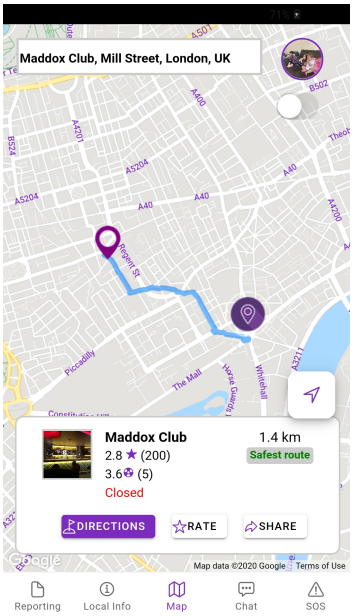

Figure 5.11: Guardian-A After clicking direction button .

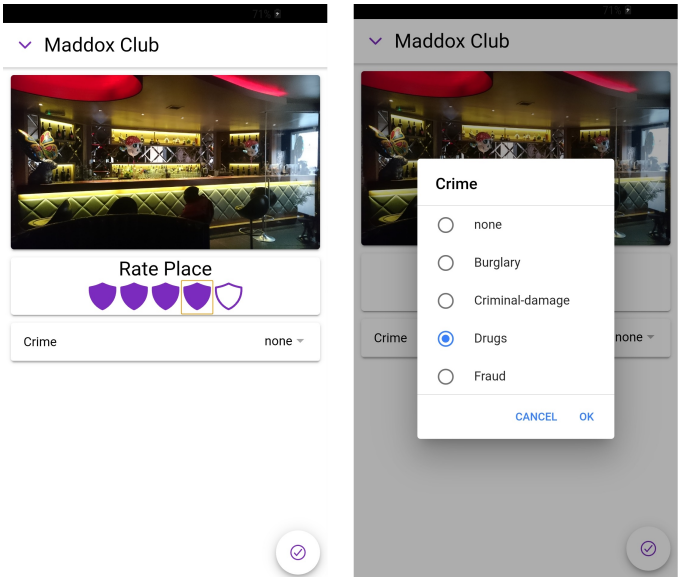

Figure 5.12: Guardian-A After clicking rating button .

5.5 Chat page

This page contains our chatting module which we built from scratch using real time Firebase services . This module contains all basic functionalities provided by any chatting system . Users can start chatting by searching to any contact on their mobile . Guardian-A provides more interesting features such as notification system which can be used to notify a chat when user reaches a specific location or when he is away from his group members by a certain distance .

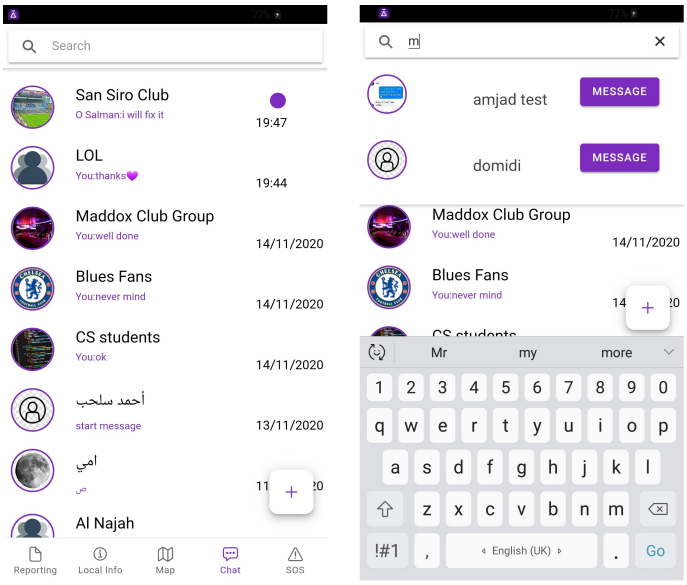

Figure 5.13: Guardian-A Chats page .

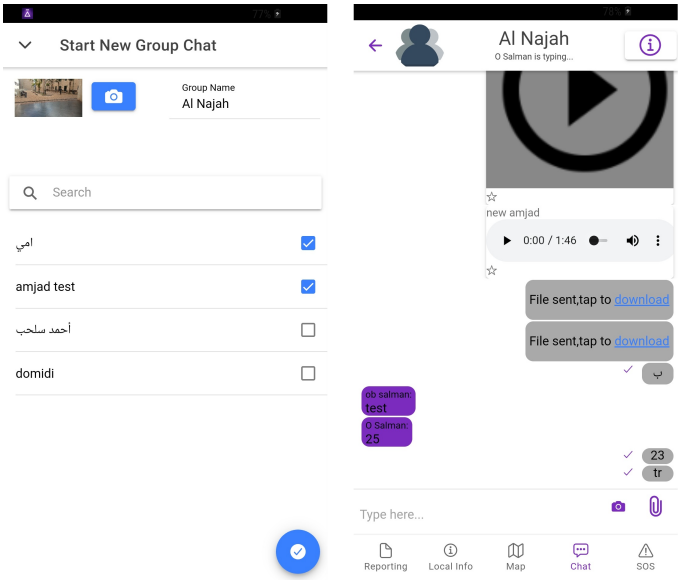

Figure 5.14: Guardian-A Starting a new group and chat page .

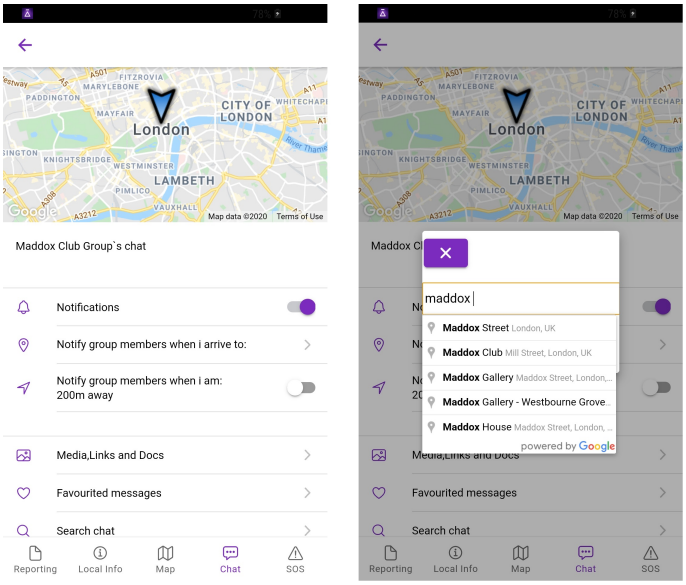

Figure 5.15: Guardian-A Chat setting page .

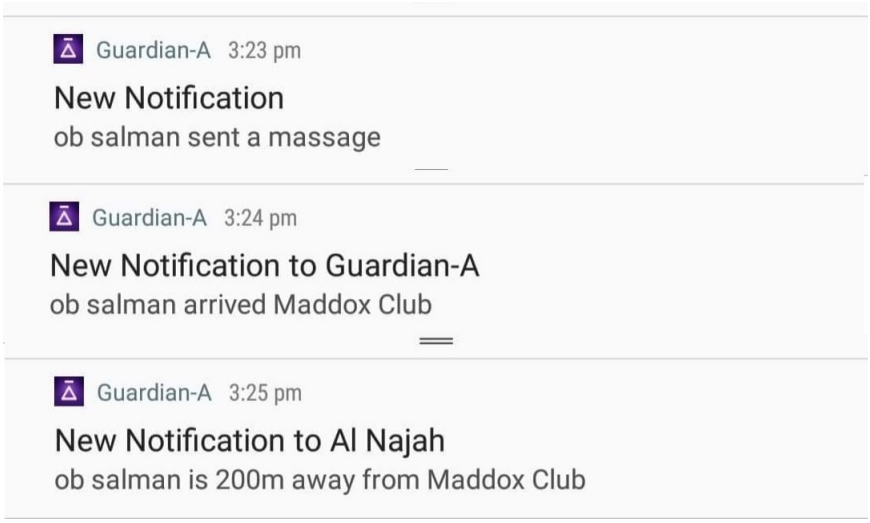

Figure 5.16: Guardian-A Notifications .

5.6 SOS button

This button is placed next to the tabs at the bottom of the screen to be accessible from any where inside the application . This button will trigger and alarm in addition to send a sms message which contains the user location to pre defined emergency contacts which user can select from his account settings menu . The button should also call the nearest police force to the user Geo-location . Also contacts who have Guardian-A account will be notified too .

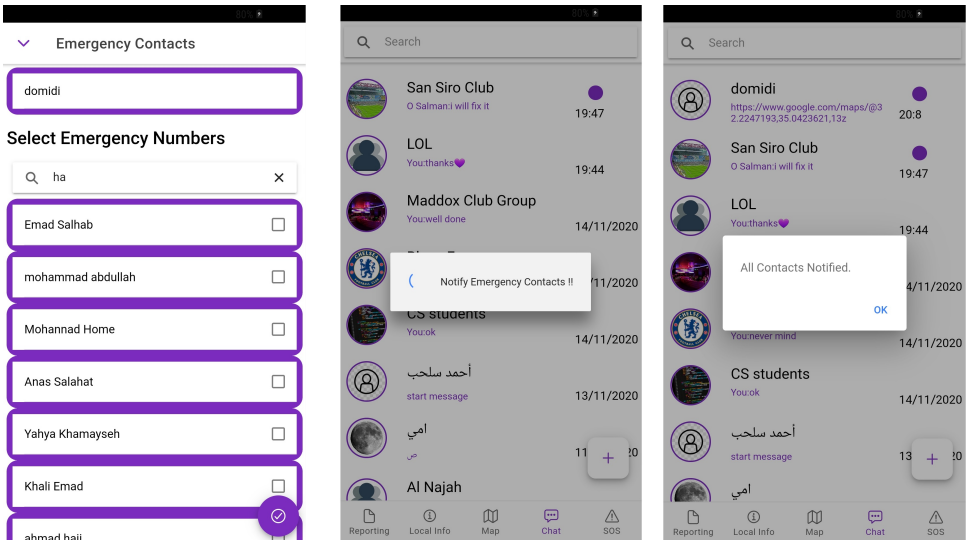

Figure 5.17: Guardian-A SOS process .

6. Conclusions and Recommendation

In this section , were going to show the most important results we have achieved in our project .

6.1 Summary

Depending on the results shown in the previous sections , we can conclude that the idea of our application is acceptable and reliable for the users . Using the previously mentioned frameworks and API‘s is recommended when dealing with similar projects. However, our project is indicated by one major limitation , which is the availability of the data and resources to work with in some other countries and regions .

6.2 Future work

For the short-term future we aim to continue our work by adding more feature and enhancing or fixing any bugs found in our implementation . On the other hand , for the long-term future we aim to release Guardian-A alpha version in order to work on it and study it from the customers point of view .

References

[1] Martin Fowler, Jim Highsmith, et al. The agile manifesto. *Software Development*, 9(8):28–35, 2001.

[2] Maximilian Schwarzmüller. Ionic - build ios, android web apps with ionic angular, 2020.

[3] Lynn Bowes-Sperry and Anne M O'Leary-Kelly. To act or not to act: The dilemma faced by sexual harassment observers. *Academy of Management Review*, 30(2):288–306, 2005.