Anti-Harassment Alert

Vipanchi Chacham[#1], Mario Padilla[#2], Anh Nguyen[#3] , Efren Lopez[#4]

[#1-4]*Arizona State University*
Tempe, AZ, United States

[1]vchacham@asu.edu
[2]mgpadil1@asu.edu
[3]atnguy37@asu.edu
[4]edlopezm@asu.edu

***Abstract*— Harassment is one of the main crimes that keeps happening everyday. To address this issue, we have built an Anti Harassment Alert application. This aims at alerting the police and family members during an emergency by just a shake on the phone. This is customized for ASU campus as we show the Blue Safe light locations too along with public places like police stations or bus stops.. This application is context-aware and has the following features: Alert activation using voice recognition, button press and gesture recognition, real time broadcast of location, instant notification to all emergency contacts and safe zone locations around ASU campus.**

***Keywords*— Voice recognition, gesture recognition, GPS, context-aware,**

I. Introduction

Street harassment especially, harassing women is still a very prevalent problem around the world. A recent study in the US found that 65% of all women had experienced street harassment [1]. There are many kinds of street harassment ranging from catcalling and commenting to raping and attacking. In this paper we propose to use different context-aware functionalities made possible by Android enabled devices and their integrated sensors in order to empower women and anyone else facing harassment, fight it and help them be confident wherever they especially around the ASU Campus.

II. Project Setup

1. *Physical Android Device*

Manufacturer: Samsung Galaxy Note8
Model Number: SM-N950UI

2. *Android Virtual Device*

Manufacturer: Nexus 5
OS: Android 7.1.1

3. *Google Firebase Server*

Firebase is a database which can be integrated with mobile and web development platforms. Firebase Auth is a service that can authenticate users using only client-side code. It supports social login providers including Google. Additionally, it includes a user management system whereby developers can enable user authentication with email and password login stored with Firebase [2]. We used Firebase for storing the location history of the incidents, Usernames and account details, storing contact lists, etc

Project ID: cse535harassment-1563086466658
Google Cloud Platform: us-west2

III. Implementation

In this section we describe how we implemented all 20 discrete tasks implemented in the project:

1. Application Prototype

In order to have a good idea of how we wanted our GUI to look like, and how it would transition between screens, we designed a HiFi prototype (shown in Fig. 1) using Axure so that we could model the main Android application GUI in the AndroidStudio.

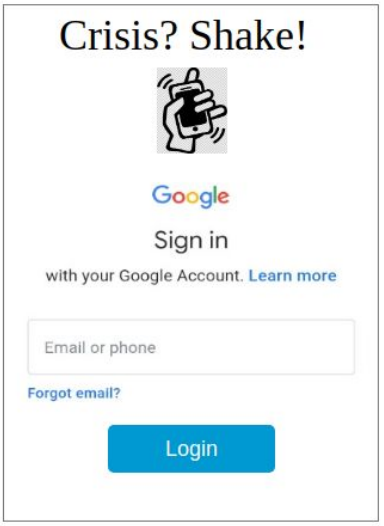

Figure 1: Application Prototype

2. GUI Development

After designing our prototype we went ahead and implemented the GUI to look as close as possible to the original design as possible.

3. Database setup

For this application we decided to use Google's Firebase.

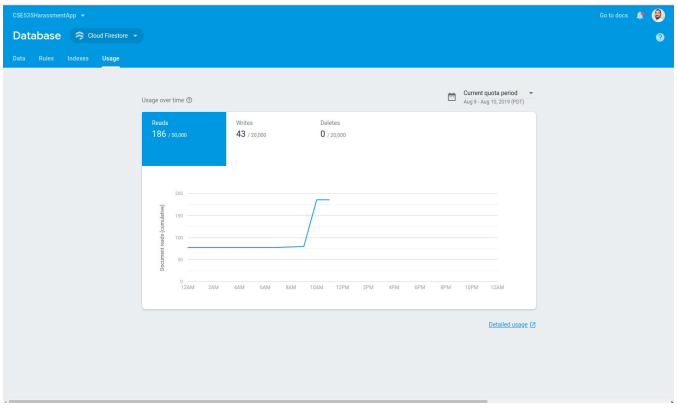

Figure 2: Firebase database console

4. Authentication

We also used Firebase authentication feature to enable Google authentication easily in our application.

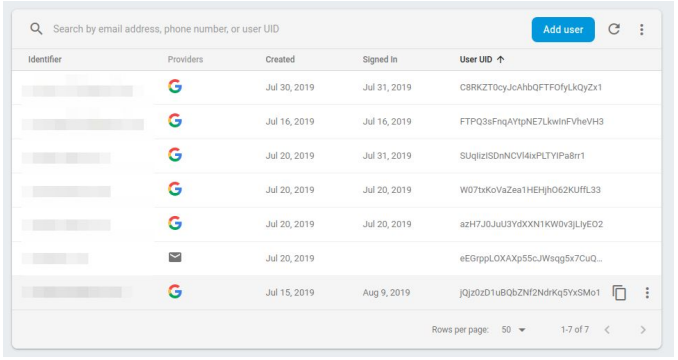

Figure 3: Authentication tokens

Using the Firebase server we can use our Google accounts to log into the application; there is no need to create new accounts.

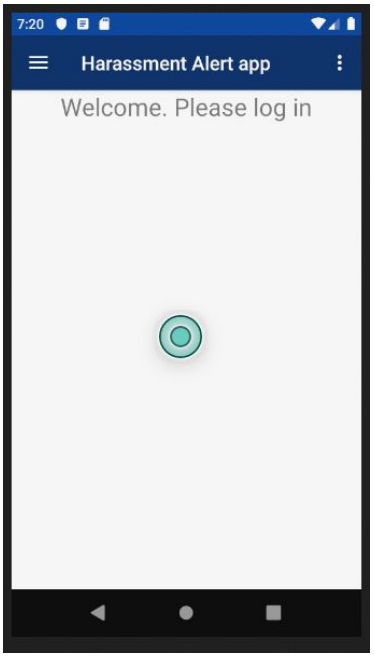

Figure 4: Screen before authentication

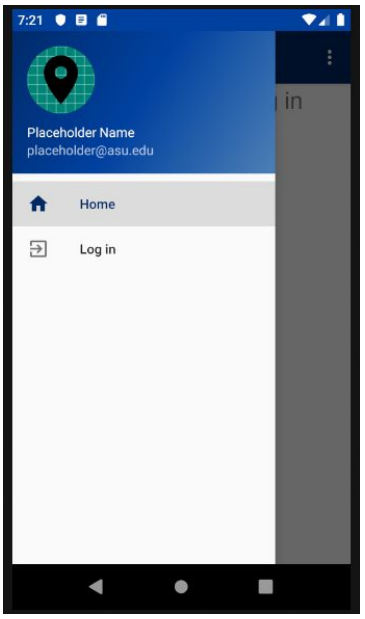

Figure 5: Login button

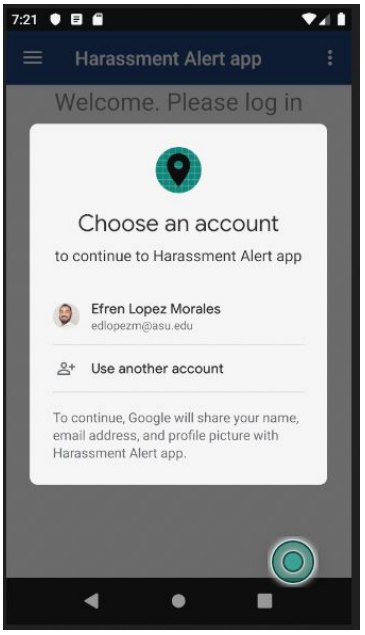

Figure 6: Choose Google account

5. Live location

We implemented live location broadcast of the user when the alarm is activated. So this live location is sent to the Police and the family.

6. Safe Zones

We implemented Safe zone locations using Google Maps API to let users know what the closest safe zones are such as the police station or the bus stops. It also shows the blue beacons for the ASU campus.

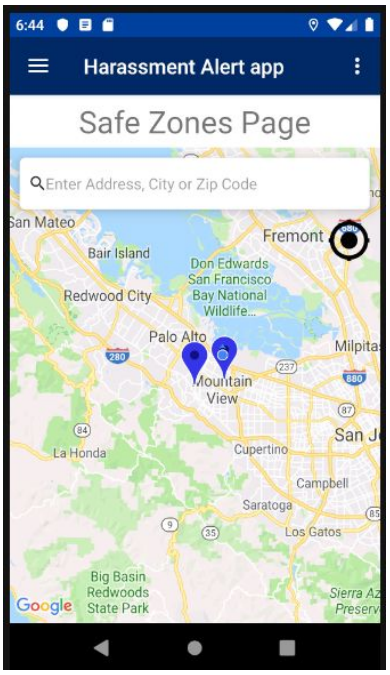

Figure 8: Safe Zones in Google Maps

7. Location History

The user's location, when they alert the police will be saved along with a timestamp in order to alert those who wander in those locations at the same time. This is an important information to avoid danger. The database has the user id, timestamp and the location.

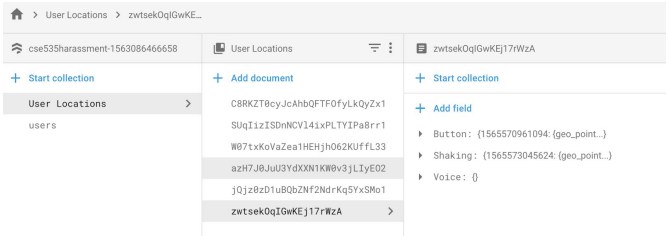

Figure 9: Location history database view

When alert is done through shaking:

Button: {1565570961094: {geo_point...}

Shaking

1565573045624

geo_point: "GeoPoint { latitude=33.4189983, longitude=-111.9399983 }"

timestamp: "2019-08-11 18:24:05.624"

user_id: "SUqIizISDnNCVl4ixPLTYIPa8rr1"

1565586643343

geo_point: "GeoPoint { latitude=33.4189983, longitude=-111.9399983 }"

timestamp: "2019-08-11 22:10:43.343"

user_id: "SUqIizISDnNCVl4ixPLTYIPa8rr1"

Figure 10: Shake gesture

8. Contact Details

For each contact we can add the required details as shown below. This is again stored in the database for the users to retrieve it whenever they click existing contacts.

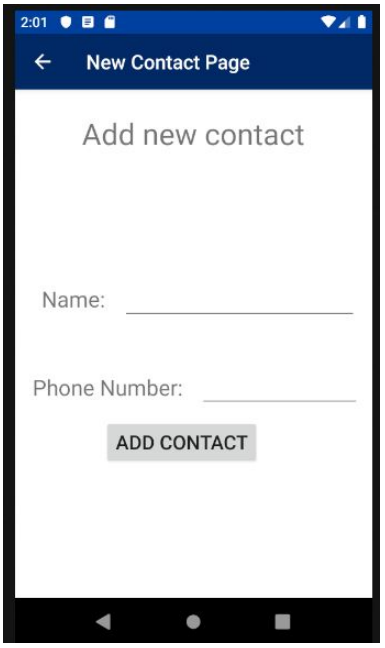

Figure 11: Adding a new contact

9. Shaking gesture recognition

We implemented a shaking gesture recognition feature in order for the user to be able to activate the alarm when they are not in any situation to open the app and press Alert button.

10. Alert button pressing

When the alert button is pressed we send a text message immediately to the emergency contacts to alert them of the danger the user might be in.

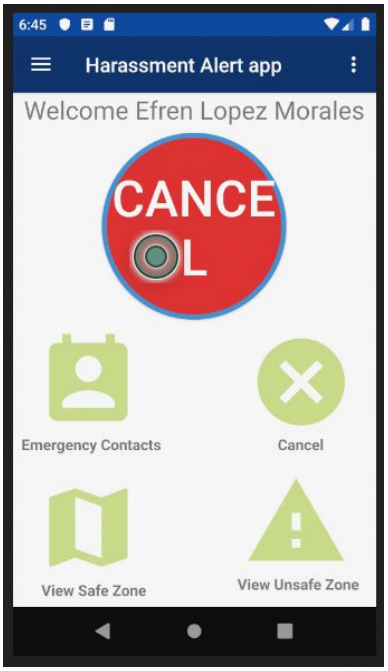

Figure 12: Alert button activated

11. Emergency contacts

In the emergency contacts tab, the user can add an existing contact or can add a new contact. These contacts will be the ones used to which the alert text messages will be sent.

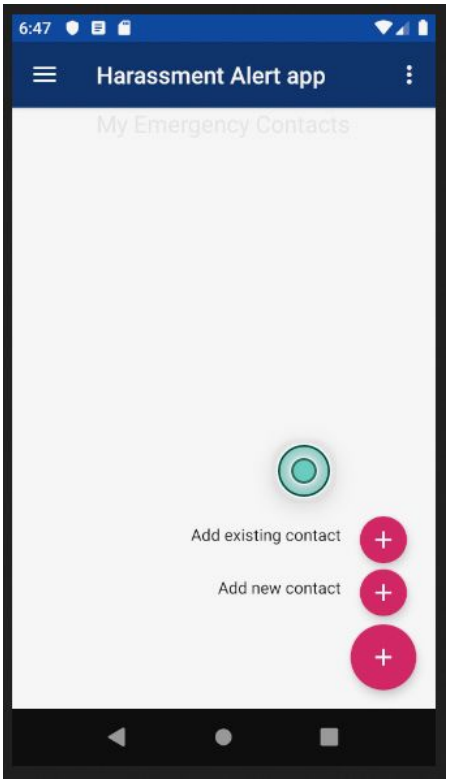

Figure 13: Emergency contacts view

12. Call the police

There is also a Call the police feature. User can call the police by pressing the Call button in the Emergency contacts tab.

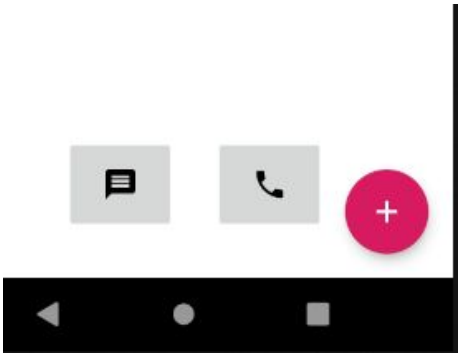

Figure 14: Call the police button

13. Send Text messages

User can send a message to the emergency contacts from the inbuilt text message option for quick access as shown in Fig 14.

14. Integration Testing

We performed integration testing after finishing all the features that every member of the team implemented in order to make sure everything worked as expected. We found a few glitches which we fixed like the back button which lead to the home screen instead of the previous screen. So we fixed a few errors like that.

15. Unsafe Zones

Unsafe zones are locations on the map where users have reported harassment incidents. This is retrieved from the location history whenever the person alerts about harassment.

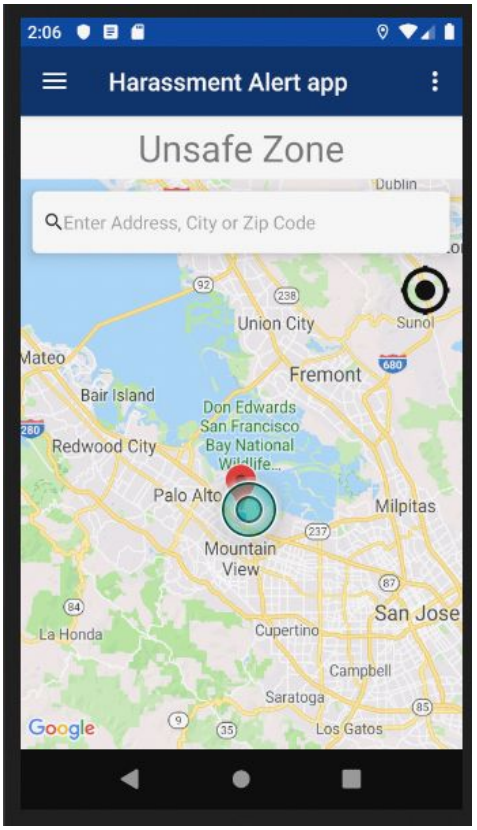

Figure 15: Unsafe zones in Google Maps

16. User Feedback

We implemented a user feedback feature to learn what users think about our app or what could be made better. This is again saved in the firebase for our own improvement purpose.

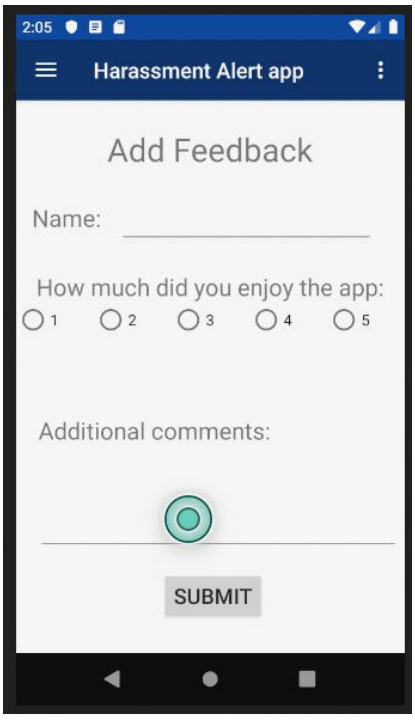

Figure 16: User Feedback form

17. Voice recognition

Since sometimes the shaking of phone fails, we have installed a voice recognition feature. Through this feature, the user can click on the voice button and say "Help". As long as the sentence includes the word "Help", the alert will be sent.

18. Share on social Media

Users can share their experience on Social Media or if they want to report any incident. We have implemented gmail, bluetooth , Google drive and message sharing methods.

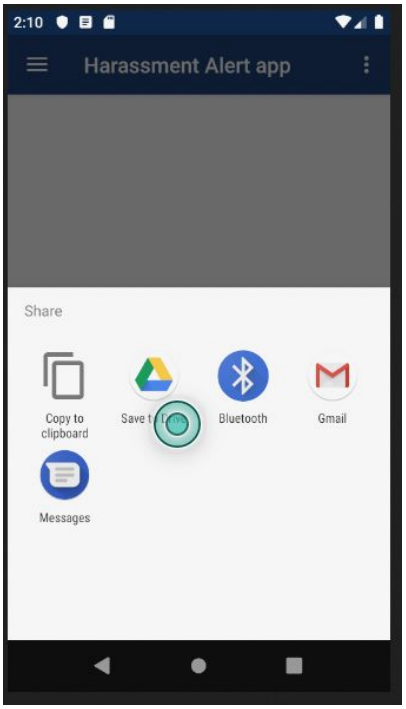

Figure 17: Share view

19. Send Text messages on alert

When the alert button is pressed or when the person shakes the phone, Police or the family members receive message as below. It'll show the location of the danger happening in order to alert the emergency contacts.

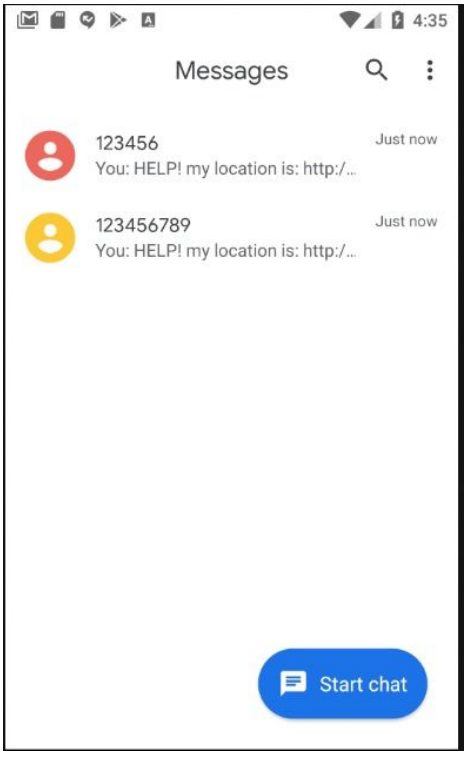

Figure 18: Text messages sent

20. Cancel in case of mistake

In case the user activates the alert by mistake, they can cancel it. The cancel button sends a text message informing the emergency contacts that it was a mistake so that they know of it immediately before they start worrying.

IV. Completed Tasks

No	Task	Assignee
1	Prototype of the app	Vipanchi
2	GUI Development	Efren
3	Setting up the database	Mario
4	Authentication	Mario
5	Live location of user	Anh
6	Location of "safe zones"	Anh

7	Location History	Efren
8	Manage contact details	Mario
9	Alert app after shaking of the phone	Anh
10	Pressing of the alert button	Efren
11	Emergency contact list	Mario
12	Call the Police	Vipanchi
13	Send Text messages in contacts- mario (include in shake part/ button)	Efren
14	Integration Testing	Vipanchi
15	Unsafe zones Warning	Anh
16	Get feedback from user	Vipanchi
17	Voice recognition	Anh
18	Share on Social media	Vipanchi
19	Send message alert button	Mario
20	Cancel in case of a mistake	Efren

V. Limitations

Our application is designed to be used within and nearby the ASU Tempe campus at the moment. However, in the future, it could be extended to include any other locations.

VI. Conclusion

In this paper we presented Anti Harassment Alert, an application which is an incredibly important app. We have implemented features like the Alert text messages, emergency calls and Safe Zone identification that help the user feel safer around ASU Tempe campus. We have made our application context-aware through the use of different sensors like accelerometer and GPS and we believe that our novel application has a lot of potential to impact the lives of students on campus.

VII. Acknowledgment

We would like to thank Dr. Ayan Banerjee for his constant guidance, support and mentoring during the writing of this academic paper, it definitely could not have been done without him.

VIII. References

[1] Statistics. (2014). Retrieved from http://www.stopstreetharassment.org/resources/statistics/

[2] Firebase (2019, August). Retrieved from https://en.wikipedia.org/wiki/Firebase