12-1-2020

EMERGENCY ALERTS AND THE APPLICATION OF LOCATION SHARING

Martha Taffa

EMERGENCY ALERTS AND THE APPLICATION
OF LOCATION SHARING

by

MARTHA TAFFA

Presented to the Faculty of the Honors College of

The University of Texas at Arlington in Partial Fulfillment

of the Requirements

for the Degree of

HONORS BACHELOR OF SCIENCE IN SOFTWARE ENGINEERING

THE UNIVERSITY OF TEXAS AT ARLINGTON

December 2020

ACKNOWLEDGMENTS

I would like to thank Dr. Christopher Conly for being my mentor and guiding me in this project. He was always available for answering my questions related to the project. I also want to express my gratitude to Bobbie Brown for assisting in documentation through different workshops. Lastly, I would like to thank my senior design group members Arpan Patel, Allea Campell and Alvaz Khan for collaborating on this Android™ application.

December 07, 2020

ABSTRACT

EMERGENCY ALERTS AND THE APPLICATION

OF LOCATION SHARING

Martha Taffa, B.S. Software Engineering

The University of Texas at Arlington, 2020

Faculty Mentor: Christopher Conly

With the advancement in technology, a lot of work has been done for the welfare and betterment of people globally. Nevertheless, there are not as many systems implemented to inform deaf or hard of hearing (DHH) people when an emergency alert is going off, or when a similar situation is taking place. This project is about the implementation of a mobile app that gives different pop-up alerts when detecting emergency sounds. The paper mainly focuses on the implementation of the audio classifier and the feature that lets a user share their location with family and friends. The Google Maps application programming interface (API) is used to implement the location sharing feature of the application. It communicates with the Android layer and audio classifier layer of the application. Location sharing is significant as it will be helpful for future feature implementations, such as distance approximation and localization of emergency signals approaching the user from a distance.

TABLE OF CONTENTS

LIST OF ILLUSTRATIONS

CHAPTER 1

INTRODUCTION

1.1 Background

Technology advancement is making a real difference in solving important problems. The focus has especially been given to minorities and underrepresented people, as well as people with disabilities. Features such as translators, subtitles and captioning for audio, smart home devices, text-to-speech, virtual assistants, and video services have been a game-changer in that respect. Nevertheless, one of the areas where there is not a lot of work is in emergency alerting systems. There is an emergency alert system in the United States that requires wireless cable systems, satellite, radio, and terrestrial TV broadcasters to address people within a few minutes when there is a national emergency. Additionally, there are some sophisticated systems, like Amber Alerts, that inform people when an abduction occurs, or severe weather alerts systems for hurricanes or tornadoes. These warning systems are critical for state and local officials to deliver significant emergency information to the impacted parts of the community.

1.2 Related Works

There are some works that are closely related to this project. Research shows that there is a neural network architecture that includes the application of deep learning techniques to detect different sounds, such as breaking glass and other sounds (Heynderickx, 2019). Environmental sounds are composed of mixtures of different sounds and despite this challenge, neural networks have been created for such classification (Medhat et al., 2020).

Hearing aids, visual alert systems, cochlear implants, accessible video phones, and loop systems are some of the technologies that are used by DHH people. Nevertheless, there are times where these technologies are inconvenient to wear or use depending on time and place. Some of these devices notify DHH people by increasing audible alerts and by flashing a special high-intensity strobe light that can wake a sleeping person. This might not be effective when people are not in their house, or when it is not suitable to carry those devices with you.

The Frankfort Fire & EMS Department has worked on hearing-impaired smoke alarms and a bed shaker. They came up with this idea for supporting those who wear hearing aids or remove cochlear implants when sleeping ("Hearing Impaired Smoke Alarms & Bed Shaker", 2020). The smoke alarm created by the Frankfort Fire & EMS Department uses strobe lights and bed shakers to alert DHH people when there is an audible fire alarm. These devices use approaches that include visuals and sound-induced vibrations to create notifications (Ketabdar & Polzehl, 2009). Although this sound awareness device can be helpful, it is restricted to just fire alarms. It might also be uncomfortable or sometimes result in side effects (McCormack & Fortnum, 2013). Moreover, it is ineffective when used outside compared to its indoor usage. The Silent Siren Android application serves as a good solution when hearing aids are inconvenient for DHH people.

1.3 System Overview

The Silent Siren is an application that notifies a user of an emergency based on the signals it identifies in the user’s surrounding. It uses an audio classifier that cross-checks the audio input from the user's device against an audio dataset of emergency signals. As depicted in Figure 1.1, the type of the notifications to be received can be customized by the user. Specifically, the notification method and alarm sound can be set based on the user's preference.

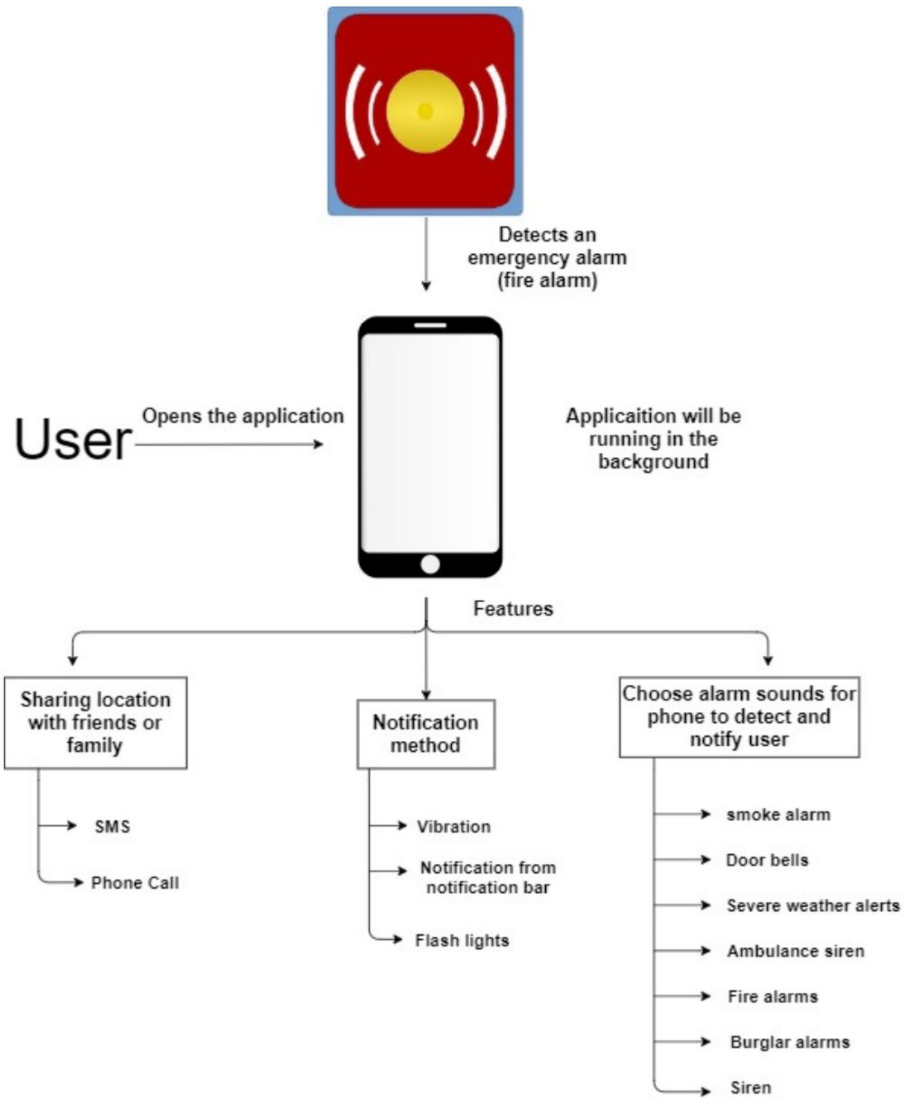

Figure 1.1: Emergency Alarm Detector System Overview Diagram

The emergency sounds the application detects includes smoke alarms, fire alarms, burglar alarms, doorbells, severe weather alerts, sirens, ambulance sirens, and police sirens.

As the application listens to an emergency signal in the background, a notification will appear at the notification bar of the device, or a vibration will be sensed depending on the user's customization. Additionally, there is a feature that enables users to share their current location to family or friends via SMS or direct phone call as illustrated in Figure 1.1.

CHAPTER 2

METHODOLOGY

2.1 System Architecture

The Silent Siren mobile app will consist of two major layers as shown in Figure 2.1. The Android layer is where most of the functionalities of the application exist. It is where the application interacts with the Android operating system to get the data needed by Silent Siren. The other layer is the Audio Classifier API, which is the backend of the system that interacts with the Android layer using the web server.

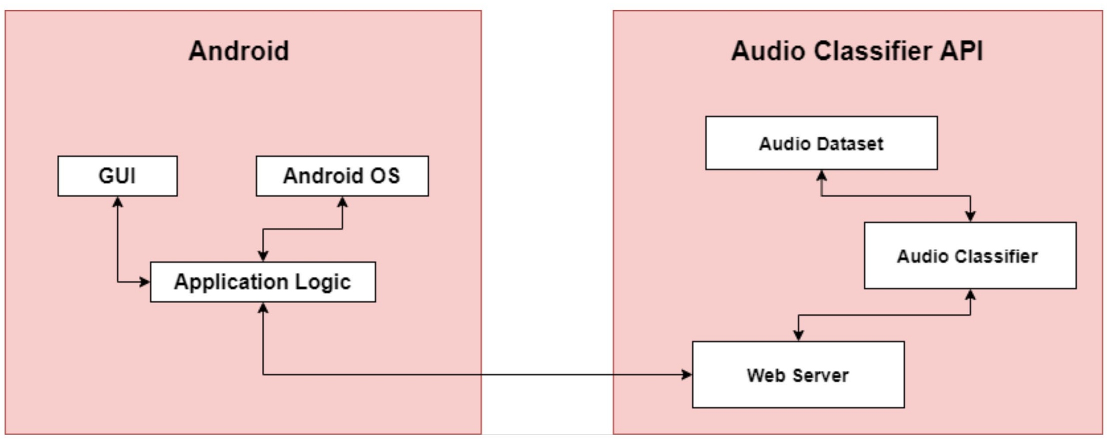

Figure 2.1: A Simple System Architecture Diagram

2.2 System Layers

2.2.1 Android Layer

The three subsystems in the Android layer are the graphical user interface (GUI), Android OS, and application logic. The Android OS provides access to the device's location, microphone, internet and other information needed for the application to function properly. The application logic processes the device's information retrieved from the Android OS. It also interacts with audio classifier API through the web server to get the classification of the audio fetched from the user's microphone. Depending on the user's preference, a notification will be displayed using the Android OS to alert the user if there is an emergency sound in their surroundings. The application asks for permission to access the device's audio input by including a permission tag in the application's manifest file. The Android multimedia framework in Android Studio includes support for capturing and encoding a variety of common audio and video formats. Additionally, the Android layer will use the operating system's GPS if the user wants to share their location in an emergency. The location APIs provided by the Android OS facilitate adding location awareness to applications with automated location tracking, geofencing, and activity recognition. Thus, the location API is used to get location permission from the user and share it with friends or family. The GUI wraps around the application logic to enable users to interact with the application.

2.2.2 Audio Classifier API Layer

The audio classifier API layer has an audio dataset sourced from a website called FindSounds, the audio classifier subsystem and the web server. The system uses the audio classifier to identify the input it gets from the application logic of the Android layer. The input from the microphone can be an ambulance siren, a doorbell, a police car siren, a fire truck siren, or a fire alarm. The audio dataset mainly consists of various audio files from FindSounds. The dataset is used to increase the classifier's accuracy in order to make the notification accurate.

The web server is used to connect the audio classifier subsystem to the application logic of the Android layer. The audio classifier is responsible for identifying the emergency alarm sounds that the Android layer receives from the microphone of the device. It returns the correct alarm type to the web server. The application logic in the Android layer uses the MediaRecorder class to capture the audio from the mobile device, save the audio and send it to the web server that has the audio classifier. The audio classifier contains a function to convert mp3 to a waveform audio file. This is done to make the mp3 audio sent from Android layer compatible with the type of audio that the classifier recognizes.

Python libraries like NumPy, soundfile, scipy.signal, glob, pyaudio, wave, pydub, and time are used for the audio classifier subsystem. After converting the source audio to a waveform audio file, it compares the spectrogram graph of the waveform audio with the spectrogram of the emergency signal audio set from the FindSounds website.

Although Java is used for application logic of the Android layer, the audio classifier is written in Python. A Python web application framework called Flask is used to create the web server subsystem for the audio classifier API layer. As the application only needs the accurate audio category type, the audio classifier API only provides a GET hypertext transfer protocol (http) request. As indicated in Figure 2.2, the web server returns a JSON (JavaScript Object Notation) object which has audio type as a property.

```
root@goorm:/workspace/FlaskAPI# python3 /workspace/
FlaskAPI/application.py 80
 * Serving Flask app "application" (lazy loading)
 * Environment: production
   WARNING: This is a development server. Do not us
e it in a production deployment.
   Use a production WSGI server instead.
 * Debug mode: off
 * Running on http://0.0.0.0:80/ (Press CTRL+C to q
uit)

172.17.0.1 - - [01/Dec/2020 17:42:48] "GET / HTTP/1
.1" 200 -
```

{"Audio Type":"Alarm"}

Figure 2.2: A Running Server

CHAPTER 3

RESULTS

The Silent Siren app is convenient and effective because of the microphone usage of devices to identify the different emergency alerts. It successfully classifies alarms and sirens from other external sounds. The audio classifier requests updates about a device's location from the Android OS using the location APIs and depends on the WiFi connection of the Android layer.

3.1 Screenshots

The screen shot of the three pages of the application is provided below.

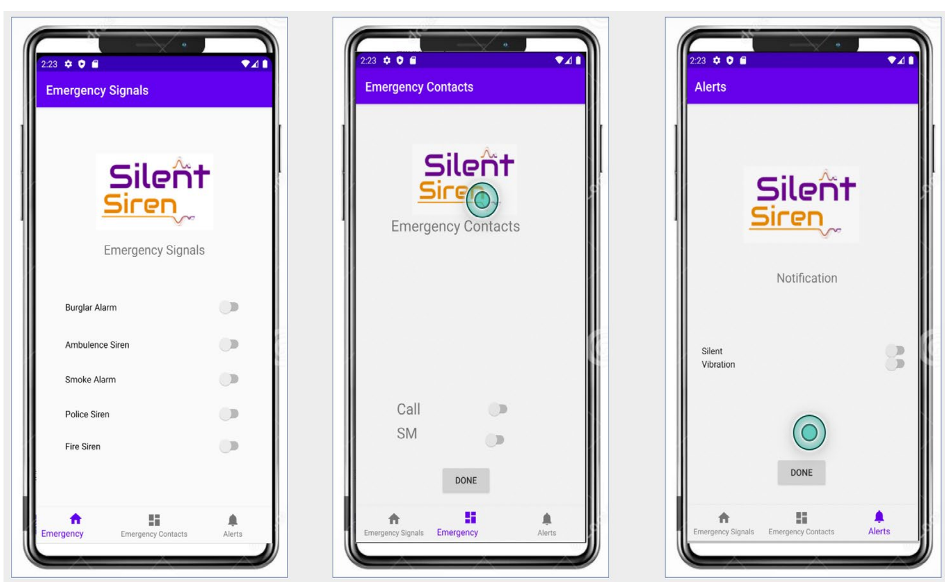

Figure 3.1: Screenshots of Silent Siren Application

CHAPTER 4

CONCLUSION

The Silent Siren application offers features relevant to different groups of people. It assists users in identifying emergencies that may occur in their environment. The sound identifier and audio classifier functionalities bundled with the user-friendly interface makes the application convenient to use. The location sharing and notification features implemented will be useful for users that may want assistance from friends or family in critical situations. Additionally, this application can be valuable for individuals who find themselves in a position where they are not able to listen to sounds in their surroundings, such as fire alarms or ambulance sirens. In future implementations, the location-sharing feature can be enhanced by including features like localization of emergency signals approaching the user from a distance and distance approximation. Moreover, the accuracy of the audio classifier can be improved by training a model using a more varied emergency signal dataset. The audio file that is sent to the audio classifier can also be preprocessed by eliminating distracting background noises such as music, rain, or wind.

REFERENCES

Hearing Impaired Smoke Alarms & Bed Shaker | Frankfort, KY. (2020). Retrieved 14 December 2020, from https://frankfort.ky.gov/571/Hearing-Impaired-Smoke-Alarms-Bed-Shaker

Heyndericks, P. (2019). Deep Learning for Security Applications: the Sound.

Ketabdar, H., & Polzehl, T. (2009, October). Tactile and visual alerts for deaf people by mobile phones. In Proceedings of the 11th international ACM SIGACCESS conference on Computers and accessibility (pp. 253-254).

McCormack, A., & Fortnum, H. (2013). Why do people fitted with hearing aids not wear them?. International journal of audiology, 52(5), 360–368. https://doi.org/10.3109/14992027.2013.769066

Medhat, F., Chesmore, D., & Robinson, J. (2020). Masked Conditional Neural Networks for sound classification. Applied Soft Computing, 90, 106073.

BIOGRAPHICAL INFORMATION

Martha Taffa is a senior software engineering student at the University of Texas at Arlington. She was born and raised in Addis Ababa, the capital city of Ethiopia. After she completed high school in Addis Ababa, she decided to study abroad. She came to the United States to pursue a degree in software engineering. She received her Associate of Science degree at Richland College, and then transferred to the University of Texas at Arlington. She worked as a peer educator on campus at the I.D.E.A.S. Center, one of the student support service programs. She enjoyed tutoring and mentoring students and hosting study sessions for computer science and math classes. After working for a year and a half at the I.D.E.A.S. Center, she received an internship at a company called Veryable. She worked there for the Fall semester as a web engineer intern. She will now earn her Bachelor of Science in software engineering with a minor in Mathematics. She will then be looking for a full-time software engineer position.