Urban Security: Final Report

Andrew Amore, Dane Miller, Joe Simons, Maxwell Powell, Nick Skiljan, Sunny Patel

The origins of the 911 system dates back to the late 60s and was designed for landlines. Being permanently located it was easy for emergency personnel to find the source of a caller. However, with mobile cell phones, this information can be slightly skewed or not sent at all. This is important information when dealing with someone in distress and the group was determined to possibly figure out a way to improve this system.

The group originally started this project by completing a brainstorming session of other relevant ideas that included a home network MAC address monitoring program, application that improves user data security, and an anti cyber bullying program. The group ultimately walked away from this first gathering with a weekly meeting time (Wednesday nights), a sprint focused methodology, and a ranked list of ideas. After the group was granted its first selection, a mobile application that improves user safety by sharing their location with first responders the team elected to call the application UrbanSecurity and wanted to establish goals.

Goals and potential project components for UrbanSecurity are shown below in Table A. The team developed each goal by looking at the what, not necessarily the how. An example of this type of thinking approach can be demonstrated to the first goal: transmission of user data. Rather than focus in on specific how the team could accomplish this goal, the team instead generated a list of potential components first. Next, a hand vote was applied and the most compelling option was selected to move forward with development.

Table A: Goal & Component Overview

Goal/Objective	Component Options	Number of Supporters
Gather & Transmit Relevant User Information	Web Application	2
	Mobile Application	3
	Radio Network	1
Store Information	Real Time Database	6
	Relational Database	0
User Assurance & Confidentiality	Encryption	6
	Permissions	6
	User Agreement	6

With effective components selected the team decided to break these down further into the following six modules: call log monitoring, information encryption, firebase realtime database, first responder side mobile application, user side mobile application, and a map to generate directions to the location. Results of each of the individual modules can be viewed below Figure 1, which shows the ideal 9-1-1 process flow when the user places an emergency call using UrbanSecurity.

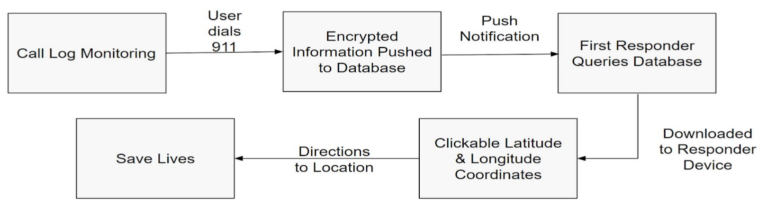

Figure 1: 9-1-1 UrbanSecurity Process Flow

A brief description of each module follows:

Call Log Monitoring:

Call log monitoring was determined to be an essential aspect to UrbanSecurity's success. This module run's in the background of an Android's processes and screens every phone call for the correct sequence of digits. When the correct sequence of digits is identified, the module, embedded in the user application, flags the relevant information for transmission.

Information Encryption:

Figure 2 shows the information flow through the UrbanSecurity encryption scheme. Java classes were created on both apps for AES encryption and decryption and in each, a predefined, 128 bit AES key was locally generated. Both apps took advantage of the java.security API to implement securely then further unit testing was added. Initially, the team had planned on using an AES key with RSA encryption to provide two layers of security, but ran into database storage type errors with the encrypted key.

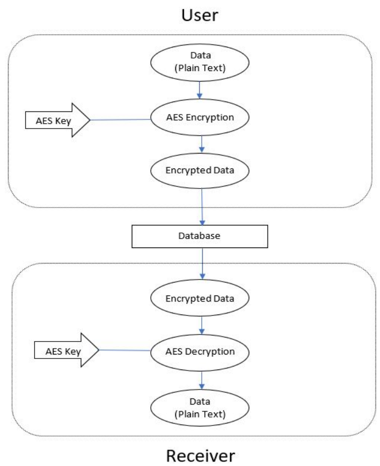

Figure 2: Encryption Schematic

Firebase Realtime Database:

When deciding how to store the data, a realtime database was strongly preferred over a relational database due the speed and ease of action the app requires, thus firebase realtime database was chosen.

Figure 3: Firebase Realtime Database

Figure 3 shows two data sets: the users data which shows each users information and the sos table showing a list of all users in an emergency. Data stored in this table gets encrypted beforehand then gets decrypted after being sent to the receiver app.

User Mobile Application:

Figure 4 represents a high level user interaction chart showing how a typical user will interact with the application after download. When you open the user app you are prompted to sign in using your name and phone number.

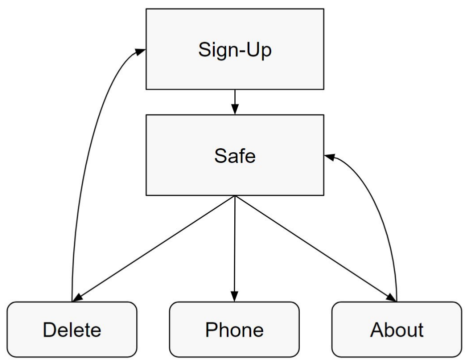

Figure 4: User Application Schematic

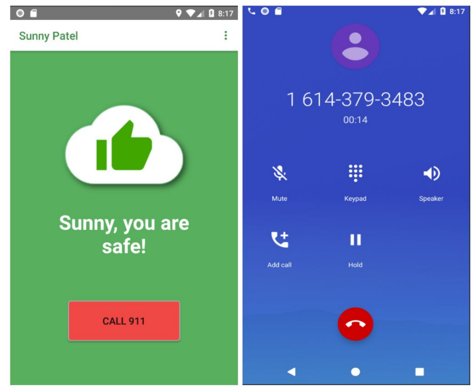

Figure 5: User Application

After signing up, the green box seen in Figure appears. Key design elements on this display include personalized text boxes that show the user's name and a "Call 911" button, shown in red, that directly dials 911. The blue box in Figure 5 shows the screen the "Call 911" button directs the user to. Please note that for testing purposes 614-379-3483 was used instead of 911 which would be displayed in the release version

When the user calls 9-1-1 or pushes the button the user's latitude, longitude, and altitude is retrieved from memory and passed to the encryption module for transmission. With an emulator the data retrieval step always pulls the same information so further validation of this feature was completed using a real life Android mobile device

First Responder Mobile Application:

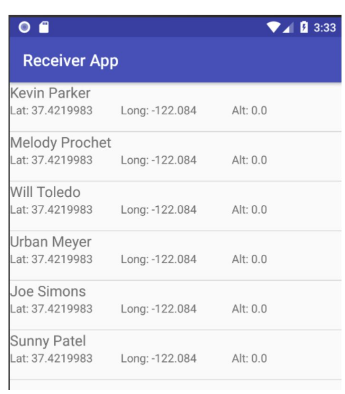

Figure 6: First Responder Application

Figure 6 shows the receiver application interface, which is a time ordered list of distressed users. All of the users displayed in this figure were generated using the Android Studio emulator, which has default latitude and longitude values of the Google Headquarters and an altitude of 0. Testing of this application was also performed on a real life Android mobile device for verification purposes.

Map Directions:

Knowing the location of someone in distress is only half the battle. The group also decided there needs to be a way for emergency personnel to have directions to the user without having to waste valuable time looking it up. Thus a maps view was implemented into the app. Figure 6 shows how each user in an emergency is presented in the receiver app so emergency responders can click on a user and directions to the their location are presented

Results and Conclusion:

The group was successfully able to complete all modules and functionality. Although there were some unforeseen and unavoidable issues, such as restrictions in the android API and data-type constraints in the database, the group was able to work together to find solutions to these issues. Through weekly meetings, close communication in Slack, task documentation in Trello, and clear responsibilities, the group was able to keep momentum through the entire project and completed not only the group's original goals but also several stretch goals such as Google Maps integration.

The application is a successful proof of concept, but there is lots of room for potential future work. A full solution would be integrated into the phone's operating system, collect more useful data (with the user's permission), and have a professionally developed user interface. A full solution for first responders would include more than just a mobile application. For example, full web interface with the ability to visualize more than one SOS would be useful to dispatcher.

In a time when Google maps and Uber can access a users location with pinpoint accuracy, it seems only logical that the 911 emergency system should also be able to do so. Due to the inefficiency of many 911 calls today this app represents a good starting point for possible improvements to the process. The original plan of sending the user's location was expanded to include other information, possibly saving valuable time during the actual phone call.