GeoBAT: Crowdsourcing Dynamic Perception of Safety Data Through the Integration of Mobile GIS and Ecological Momentary Assessments

by

Brian Adam Jeantete

A Thesis Presented to the
Faculty of the USC Graduate School
University of Southern California
In Partial Fulfillment of the
Requirements for the Degree
Master of Science
(Geographic Information Science and Technology)

May 2019

To my wife, kids and parents.

Table of Contents

List of Figures

List of Tables

Acknowledgments

I would like to thank all staff of the Spatial Sciences Institute who have guided me through this program over the last two years. This includes Dr. Jennifer Bernstein, Dr. Robert Vos, and Dr. Vanessa Osbourne who provided guidance as this study progressed. Without them I would still be trying to develop a hypothesis. I would also like to thank Dr. Yao-Yi Chiang and Dr. Laura Loyola for their invaluable advice throughout this process, and finally Robyn MacNab, who always worked tirelessly to get me into the program, even though I was convinced I did not have the time to pursue my master's degree. Most importantly, I would like to acknowledge my family, who have stood by me throughout this process; my employer, who provided this opportunity, and to all the mentors and advocates in my life who believed in my potential to succeed.

List of Abbreviations

ACS	American Community Survey
APD	Albuquerque Police Department
API	Application Programming Interface
APK	Android Package File
ART	Android Runtime
AVD	Android Virtual Device
AWS	Amazon Web Services
BaaS	Backend-as-a-Service
CABQ	City of Albuquerque
EMA	Ecological Momentary Assessment
ESM	Experience Sampling Method
FCM	Firebase Cloud Messaging
FOCA	Fear of Crime Application
GEMA	Geographic Ecological Momentary Assessment
GeoBAT	Geographic Behavioral Assessment Tool
GISc	Geographic Information Science
GPS	Global Positioning System
HAL	Hardware Abstraction Layer
IDE	Integrated Development Environment
iOS	iPhone OS
IRB	Institutional Review Board
JDK	Java Development Kit
JSON	JavaScript Object Notation
KML	Keyhole Markup Language
OS	Operating System
POS	Perception of Safety
SDK	Software Development Kit
UI	User Interface

UX	User Experience
VGI	Volunteered Geographic Information
XML	Extensible Markup Language

Abstract

Perception of the surrounding environment influences personal behaviors and the way humans interact with each other over time. The fear of crime and perceptions of safety are a major contributor to these behaviors, and these perceptions influence the decisions made by law enforcement and city planners. Over time, a wide range of studies have been performed to understand the triggers that accentuate the fear of crime and the possible solutions to alleviate these fears. Most of these studies have been static in nature and rarely included a dynamic geospatial component. This thesis details the process used to integrate a dynamic geospatial component by developing a mobile Ecological Momentary Assessment (EMA) mobile GIS application prototype that: (1) enables users to collect spatiotemporal and fear of crime perception data in real time; (2) pushes notifications to users at specific times as a reminder to collect this data; (3) distributes this information to a Realtime database; and (4) provides values for integration into multiple GIS platforms for subsequent GIS analysis. Once the mobile application was ready for release, testers were distributed throughout the city of Albuquerque where they collected data and provided feedback on application functionality. At the conclusion of testing, all requirements to develop a functional EMA mobile application were achieved. Future work includes adding additional application features, external data sets for further analysis, and iPhone OS development for wider distribution.

Chapter 1 Introduction

The accelerated distribution of web and mobile devices over the last decade has led to a revolution in information sharing that provides users with an unlimited flow of news and social media. This information revolution provides individuals with the resources to share information and expand their social reach on a local and global scale; however, greater access to information also introduces an increased vulnerability to overexposure and disinformation. Attempting to manage and process this information is becoming a full-time job for a human brain that still uses an ancient operating system to generate information (McNaughton-Cassill 2017).

One of the significant side effects of this information overload is the increased exposure to crime events and the tendency of individuals to engage in protection and avoidance behaviors, even though the event does not directly impact them. Information over load, reduced social interactions, and prolonged exposure to criminal activity through mass and social media increases the fear of victimization amongst individuals whose perceptions of crime are shaped by media consumption (Intravia et. al. 2017). Even though the threat is far removed and out of an individual's control, an individual may experience indirect victimization when they hear of other individuals encounters with crime (Clark 2003). This feeling of helplessness and the fear of imminent victimization by an individual and a group of individuals can be more dangerous than the perceived crime itself. Overexposure to indirect victimization has been linked to increased levels of depression, anxiety, impaired health and well-being, and most importantly, it brings a heightened sense of mistrust with unfamiliar locations and people (Scarborough, 2010). The fear of crime, and perceptions of a higher risk of victimization, motivates individuals to advocate for tougher laws that punish those that pose a threat to society, even though most individuals are ill-informed about crime and the criminal justice process (O'Connell 1999; Callanan 2005).

As mentioned by Callanan (2005), crime captures the attention of the public because the sensationalism presented by the media perpetuates the fear of victimization. However, there are many local variables that also play a role in amplifying fear of crime behavior. First, an individual's perceptions can be influenced by physical variables such as graffiti and dilapidated environment, and social variables like public drunkenness and lack of order. These signs of disorder or visible cues can trigger a fear of crime, even though there may be no criminal activity in the area (Ross and Mirowsky 1999). Second, changes in the environment, such as changing demographics, can cause anxiety amongst individuals due to a lack of social control. Differing demographic backgrounds cause individuals to become uncertain about whom they can trust, which leads to a lack of community cohesion and increased anxiety (Lane and Meeker 2003). On the other hand, interpersonal communication within a community can increase an individual's fear of crime because these communications tend to be more localized and the stories hit closer to home. Hearing about the crime increases an individual's sense of urgency and results in their motivation to find coping strategies to deal with this fear (Hale 1996).

Since the fear of crime impacts society at large, even more than actual crimes, it is no surprise that it is one of the most researched topics in multiple disciplines (Callanan 2005). However, the integration of spatial perspective has been lacking in this research, and according to Goodchild (2004), the potential for growth in the application of crime perception research to GIS is promising.

Therefore, the objective of this thesis is to develop and test a mobile GIS application prototype that utilizes crowdsourced data to measure the fear of crime, and perceptions of safety, from the perspective of a diverse set of participants on a spatiotemporal level. The development of a mobile GIS application, and the collection of spatially explicit crowdsourced data for this project, will allow researchers in the future to track individual behaviors and emotions throughout a typical day. Tracking from a mobile device allows participants to report the status of their fears immediately without the impediments of traditional studies that are based on recollections or ‘what ifs’.

1.1 Statement of Need

The fear of crime and understanding individual actions to enhance safety within their personal space is important to this study. In addition, the side effect of bias, which could lead to wrongful accusations of individuals who enhance this fear needs to be researched. These perceptions are detrimental to the individual who is in fear and the individual who is the target of this fear. The sensationalism of crime on the news for ratings, and the spread of fear by politicians who guarantee safety for votes, helps to perpetuate the fear of crime in the U.S., while the actual number of crimes decrease (McCall 2007). The ability to track this fear and address the underlying issue would be a valuable resource for community cohesiveness and rebuilding.

Perceptions of crime and the subsequent consequences have inspired multiple research studies. Sakip (2013) researched the perception of safety (POS) and how it influences neighborhood design (gated vs. non-gated). A study by Rees-Punia (2018) links the fear of crime to the expanding obesity epidemic, due to the threats that individuals perceive in their environment. Most of these studies focus on criminology and psychology, with minimal focus on geospatial location and scale. The integration of Geographic Information Science (GISc) into the study of crime perception and its repercussions provides a spatial perspective that allows individuals to visualize locations perceived to exhibit higher crimes rates, and it supports analysis for comparison studies between fear of crime locations and actual crimes. As mentioned by Doran and Burgess (2012) the introduction of fear-based parameters into mapping platforms provides an additional layer of understanding, as well as information that is locally and geographically relevant when compared to traditional statistical approaches.

The development of a mobile application to measure the fear of crime and bias on an individual scale will add value to the research on the impact of individual perceptions of crime on societal behaviors. This approach will follow the studies by Chataway (2016) and Solymosi (2015), who developed a mobile application to measure the fear of crime using crowdsourced data. Their research focused on the behaviors of users based on dynamic locations to ensure the everyday experiences of individuals were recorded. Prior to the development of mobile applications that track and measure individual emotions, surveys were completed by paper when most respondents were in the safety of their homes. Measuring the fear of crime with paper surveys does not consider changes in individual experiences and environmental context in real time. With the ability to leverage mobile technology to locate areas where people feel safe or unsafe, it will help identify areas that need intervention to improve perceptions of safety (Solymosi 2015).

Realizing the significance of this research is important, because the intention is to accentuate this study through mobile technology, the methods used to measure an individual's fear of crime, and the subsequent bias that usually accompanies it. According to Chataway (2016), prior to Solymosi (2015), there was no known research on the use of mobile technology to generate micro-scale survey data on the fear of crime and risk perception formation. These studies opened the door to this research and left room for future study. The development of a mobile application for this project leverages the concept introduced by Solymosi and Chataway; however, this application provides extra parameters that may help to track individual perceptions in space and time. Specifically, by integrating both Google and Firebase API's, the mobile application provides opportunities for wider distribution for use by multiple agencies, real time data collection, notifications, and the ability to authenticate users. Most importantly, the goal of this application is to provide GIS data to subsequently analyze individual reactions to perceptions of crime triggers at their location and at the time of the report.

1.2 Motivation

This study is motivated by the need to develop a proof of concept for a functional mobile geospatial application that tracks an individual's fear of crime on a dynamic scale and provides a mechanism for future research to compare the results of participant input to actual crime data. Developing a mobile platform to measure the fear of crime provides a framework that could be used for subsequent studies to analyze crowdsourced data, determine if perceptions of victimization contrast by socio-demographic groups, and output situational variables from a diverse set of users to measure the extent of possible bias in respondent data. This could be accomplished by providing a mobile platform that measures an individual's perceptions of crime at specific places and times and defines areas of concern to provide solutions by locating priority areas for fear reduction. Focusing on disorders that may trigger fear, such as apparent social and physical disorders, or indicators that may cause individuals to avoid certain areas, can provide policy makers with the tools to implement innovative solutions to mitigate these fears and possibly decrease potential crime events at the same time.

1.3 General Overview

The mobile technology developed for this study leverages Geographic Information Systems to track participants on a spatiotemporal scale and the intention is to compare locational perception data to actual crime rates. Notably, this is an Ecological Momentary Assessment (EMA) tool that is event and signal contingent. Event contingent is a location and time determined by the user, and signal contingent is a notification sent at a specified time by the developer, reminding the mobile user to enter perception data. This application enables participants to enter information about specific disorders that may cause these perceptions of safety and the reasoning behind these perceptions. Location based reporting and the data that is generated from participants will provide information on what actions can be taken to alleviate negative perceptions while maintaining individual diligence to maintain safety.

1.3.1 Study participants

This application will be designed for a diverse set of users who will be provided with the ability to log in to the application, verify their location on a map, and respond to notifications to complete their surveys when directed. A diverse group of users will be important to this study because their responses will provide data that can be leveraged to measure perspectives that resonate from different economic, ethnicity, age, and gender backgrounds. Applying these measurements to actual crime data in the future will help GIS analysts compare diverse perception data results with actual data sets to determine how behaviors may vary between individuals with different backgrounds.

1.3.2 Intended customers

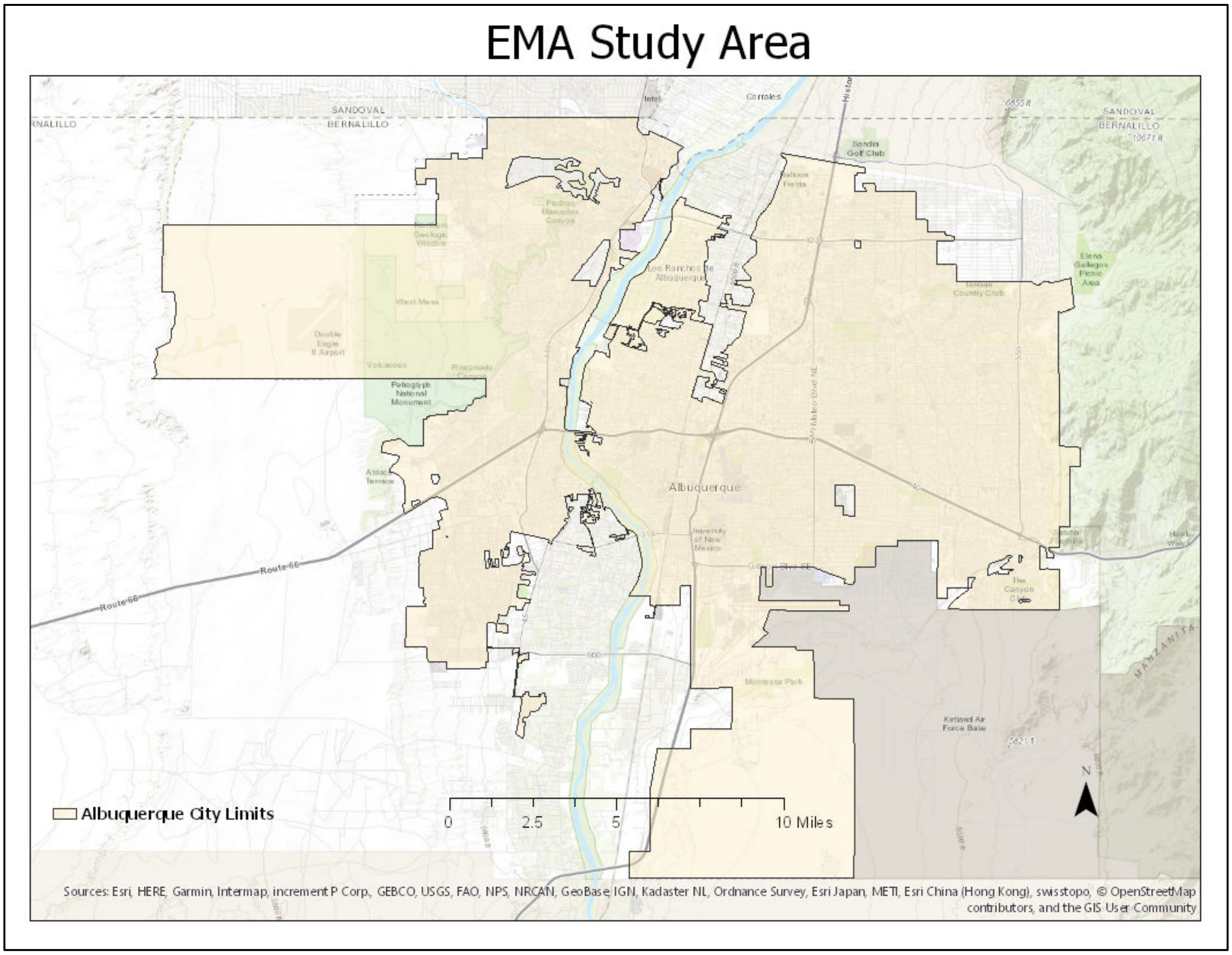
This study aims to provide a tool that can be used by local policy makers and law enforcement to analyze the extent of the fear of crime within their communities. Specifically, the intention is to generate interest within the City of Albuquerque (CABQ) and the Albuquerque Police Department (APD) to use this application as a tool to measure individual perceptions of crime throughout the city. Understanding which areas trigger fear-based responses most frequently will aid in determining if these areas need attention for future restoration or crime mitigation.

1.3.3 Study Area

The city of Albuquerque provides a case study due to perceptions of increasing crime rates, the constant coverage of the “crime crisis” on local news media, a high volume of individual posts relating to crime on social media, increasing cases of police bias (U.S. Department of Justice 2014), and the daily reminders from the media regarding high law enforcement attrition rates and the feeling that the community is not being protected.

The side effects of the increased fear of crime in Albuquerque can be observed through the population shift in this city and the rise in personal protective measures. Between 2000 and 2010 Albuquerque experienced a population increase of 21.90%; however, between 2010 and 2018, the growth rate struggled to rise above 2% (Albuquerque Population 2018). The struggling economy and the lack of opportunities in the city are both major factors in the slow growth, but the negative reports on crime, and the fear of being victimized also influence an individual’s decision to move in or out of the city (U.S. News 2017).

Due to the increased media focus on crime, the city of Albuquerque (Figure 1.1) is a prime candidate for the assessment of citizen perceptions of crime and its possible impact throughout the city. The choice of Albuquerque as a pilot city provides the ability to assess participant perceptions of disorder at specific locations and time of day. In addition, the subsequent comparison of perception of disorder data to actual crime rates through a GIS could help the City of Albuquerque and APD assess the extents of these perceptions in proportion to reality within the study boundary.



Sources: Esri, HERE, Garmin, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, IGN, Kadaster NL, Ordnance Survey, Esri Japan, METI, Esri China (Hong Kong), swisstopo, © OpenStreetMap contributors, and the GIS User Community

Figure 1.1: Map of study area

1.3.3 What are user responsibilities?

This application will require a great deal of user interaction to ensure that a large percentage of perception-based data is collected from user preferred locations. In order to make this possible the user must have a tool that provides the following: (1) notifications to collect data; (2) a connection to a robust map service; (3) symbology that marks the device location of each participant; (4) the ability to mark new or updated locations; (5) directs participants to a survey platform to answer questions, and confirms the data is collected. Users must follow directions, read the documentation, and understand how to switch between Android Activities, which are essentially displays that the user interacts with.

Each user will be expected to provide authentication information, read and approve a safety disclaimer, verify their device location, take the provided survey that is linked to their verified location and time, and confirm the data has been entered.

1.3.4 Application Backend Functionality

This project uses the Android operating system and Android Studio to compile data for the GeoBAT mobile application. Reference maps will be generated from the Google Maps Application Program Interface (API) and Backend as a Service (BaaS) functions will be provided from the Google Firebase API. The primary BaaS functions for this mobile application will include authentication, data storage, and notification services.

1.4 Document Structure

The objective of this thesis is to develop a mobile GIS application that tracks an individual's fears of crime on a spatiotemporal basis. Chapter 2 reviews the related work and mobile applications that have been developed by others to track individual behaviors related to the fear of crime. Chapter 3 covers the planning required to develop a functional EMA mobile application and the components required to ensure that this proof of concept can be distributed successfully to the intended users. Chapters 4 and 5 will cover workflow and results of the application development, and the results of the uses/surveys from the application. Finally, Chapter 6 will cover the conclusions and recommendations for further work.

Chapter 2 Related Work

This chapter provides a review of mobile application technology and reviews the application of this technology to use crowdsourced data for dynamic geospatial data sets. Specifically, this chapter covers the development of mobile technology, its expanding influence on society, the integration of this technology into tracking individuals on a spatiotemporal basis, and how this has morphed into a sample of EMA tools that can dynamically track a user's perception of safety in a study area. Finally, the intricacies of analyzing bias values derived from crowdsourced data is briefly discussed, and we conclude with case studies that can use GIS to map the fear of crime.

2.1 Expansion of mobile technology

The advanced development of mobile technologies over the last decade, and their use in geospatial data collection, has resulted in the exponential growth of an industry that was not as influential a decade ago. It is expected that the number of mobile phone users worldwide will exceed the five billion mark by 2019 (Statista 2017). With the increase in mobile users, the opportunity to acquire data from these users opens the door to an unlimited amount of research regarding human behaviors and interactions. The geospatial community is one of the main beneficiaries of this technology, due to the distribution of five billion potential data collectors worldwide. The use of mobile phones as a source of information has morphed into a source of production, as consumers take on an increased role of data distributors. According to Sui (2013), the advances in technology have changed the landscape in how geographic data is collected and used. This expansion in technology has led to a flood of data that is pushing demand for larger IT-infrastructures, also known as Big Data.

2.1.1 Mobile Application Development

As the number of mobile phone users expand, and the technology to build platforms for these user's advances, the number of developed mobile applications is increasing at a significant pace. Mobile users have endless choices when looking for applications that are either native to their OS or hybrid applications that cross all platforms. In fact, according to projections, in the first quarter of 2018, Android users will be able to choose between 3.8 million applications, and Apple users will be able to choose between 2 million applications (Statista, 2017). This is a total of approximately six to seven million applications, if we add the alternatives provided to Android and Apple. With the integration of location and mapping services into a large number of these applications, consumers are slowly integrated into the geo-spatial world. It is now becoming a part of daily life for mobile users to look at a map to reference the best place to eat, the quickest path to their destination, the best/worst schools, etc. The list of location-based applications is extensive, and the potential to mine users for geo-spatial data acquisition adds value to GISc by increasing the availability of data that is valuable to geospatial analysis, and in the case of this study, real time crime perception research.

2.1.2 Mobile Applications and the advent of virtual crime related tracking

The increasing number of mobile applications, and the advancement of cellular network technology, has expanded opportunities for innovative research in assessing crowdsourced perceptions and bias in fear of crime related reporting analysis. The ability to analyze information produced from mobile users, and direct it towards a specific purpose, is becoming an invaluable asset. One popular topic pursued by developers and law enforcement organizations is the development of mobile applications that use existing and crowdsourced data to analyze dynamic spatial-temporal crime and behavioral trends. With the advancement in the ability to upload photos, videos, and share text from an accurate GPS location, the distribution of real-time crime and perception data can be distributed quickly to other mobile users, government, and law enforcement. Prior to the development of web and mobile applications, the collection and distribution of crime and perception data would take days or months after the initial report from the participant, resulting in outdated datasets that lost value as time progressed.

Today, crowdsourcing is used to provide crime and perception data for applications such as Nextdoor, Naber, and Neighbors by Ring. This data is used with increasing frequency to track suspicious and criminal activity, and social media platforms such as Twitter and Facebook are used for crime predictability analysis and tracking. As mentioned by Bendler (2014), social media networks provide meaningful user information because individuals tend to share more information about themselves on social media platforms. Social media is a prime investigation tool used by law enforcement, and with a growing number of users, the interaction leaves behind a long informative trail that can be used by others (Golbeck 2015).

2.2 Existing Mobile Crime Tracking and Safety Perception Applications

This project is influenced by multiple studies that use existing mobile applications to track crime and perceptions of safety through citizen participation and law enforcement input. Understanding research applications that have been developed to track crowdsourced input, and their underlying designs and distribution systems, provides a benchmark to develop newer systems that will continue to improve the analysis of individual perceptions of crime on a geospatial scale.

Since 2010 there have been multiple studies on the most efficient application design to track crime and perceptions of safety. This includes Transafe, a conceptual mobile application developed by Hamilton, Cheng, and Choy (2011) that attempts to measure public perceptions of safety within the public transportation system in Melbourne, Australia. iSAFE, a mobile application developed by Ballesteros, Carbunar, Rahman, Rishe, and Iyengar (2014) that combines indexed crime datasets with social networks to provide safety recommendations based on a user's device location. Finally, iValet, an application developed by (Razip et. al. 2014) that equips law enforcement with effective situation awareness and risk assessment tools to perform analysis and detect trends. These projects make up a small sample of the research that is available on this topic; however, they stand out due to their emphasis on crowdsourced behaviors, safety perceptions, and crime trend analysis capabilities. In addition, there is an endless stream of mobile applications that specifically track and report crime, however each one differs in their configuration and purpose.

2.3 Ecological Momentary Assessment

Ecological Momentary Assessments (EMA) are conducted to assess individual behaviors on a spatial-temporal basis in real-time, and within an individual's natural environment (Burke et. al. 2018). With embedded Global Positioning System (GPS) technology, temporal sensors, and the ability to store informational data to a database, mobile applications can be designed to implement Geographic Ecological Momentary Assessment's (GEMA's) for perception studies and provide opportunities to involve large groups of participants. The use of EMA's are predominately applied in the health and psychological fields, but the addition of spatial and temporal characteristics makes these assessments beneficial in the geospatial realm, notably by enabling comparisons of individual behaviors to the physical environment.

2.3.1 Assessing the accuracy and precision of EMA's

Dependence of embedded GPS on mobile devices for successful tracking of respondents is crucial to most EMA studies; however, interference with GPS reception and receiver error needs to be accounted for. The spatial accuracy of an individual's input data depends on a robust GPS receiver that can communicate with available orbiting satellites (Mennis et. al. 2017). Location accuracy can be compromised if an individual is in an environment that blocks the receiver, or if the receiver's functionality is diminishing. This could be concerning because the possible location errors would be difficult to verify once delivered to the end user. Unless parameters are placed in the EMA to specify an individual's physical location, such as indoor/outdoor, the exact location of the user may be impossible to verify. Initializing an EMA study would require awareness of these issues and the implementation of this awareness into the interpretations of the study is essential (Mennis et. al. 2017).

2.3.2 Applying EMA in mobile technology

As of 2017, two studies have applied mobile technology to analyze individual perceptions of crime through EMA's. As mentioned in the previous chapter Solymosi, Bowers, and Fujiyama (2015) introduced a mobile platform named the Fear of Crime Application (FOCA) to analyze individual perceptions of fear in real time within the city of London. This application applied the Experience Sampling Method (ESM), another form of EMA, to assess dynamic input from an individual at their reported location and at a specified time frame. The application of the ESM to mobile application platforms deviated from past studies that gathered static data through mailed surveys, took months to complete, and lacked the valid information to assess the fear of crime within the respondent's natural environment. Testing the FOCA application involved six university students who were pinged at different times during the day over a period of one month. Each ping was a notification for each participant to submit data on the state of their perceptions of safety at that specific time. The data collected from this study revealed that the fear of crime is a dynamic variable that varies between individuals and can change within a person over space and time. At the end of this study Solymosi determined that the sample group had to be larger to produce reliable data; the lack of participants left room for further development of the FOCA prototype.

The study by Chataway, Coomber, and Bond (2017) paralleled Solymosi's, however this study triggered participants to respond to survey questions when they entered specified boundaries set by the developers, which limited the study to locational triggers and negated temporal triggers. This study was limited to a small sample group (N = 20) of students within a study area in Australia; and, like the study by Solymosi, it was concluded that larger sample sizes were required to obtain significant results. The goal of both studies was to alleviate traditional static fear of crime measures and apply dynamic spatial and temporal factors to measure individual perceptions within their proximate environment using mobile technology. Initiating EMA's using mobile devices opened the door to real time data collection on individual perceptions of crimes at localized areas and left room for future development to address this phenomenon on a microscale.

2.4 Applying the Bias Factor in Crime Perception

The development of mobile applications to track crime use crowdsourced data with the intention of building a safer place in which to live. With millions of cameras focused on so many areas, and with added camera functionality to report suspicious occurrences, the consensus would be that crowdsourcing to track individual perceptions of crime is guaranteed to be effective. However, this argument assumes all members of the crowd are objective, but as we know, most participants are human beings and tend to think subjectively.

2.4.1 Perceptions of crime and crime reporting

The term "Wisdom of the Crowd" is the common perception among those who believe in the effectiveness of crowdsourcing in analyzing crime trends. The ability for citizens to report crimes in real-time and send this data to law enforcement for tracking seems like a promising premise. The problem starts when we analyze each person's perception of the crime and the person that is being reported. Hipp (2010) analyzed these perceptions by factoring in a person's social environment and demographics by studying homogenous perceptions of crime in individual neighborhoods, and he concluded that perceptions of crimes among minorities significantly differed than perceptions of crimes among Caucasians. Perceptions of crime and safety, whether they are positive or negative, also fluctuate between gender, age, and economic background.

2.4.2 Race as a factor in big data analysis

The advent of big data and the integration of GIS into crime analysis has provided additional capability to law enforcement, including the ability to dynamically track and predict potential crimes. Many municipalities embrace this functionality due to the perceived notion of objective outputs from algorithms developed to perform crime analysis. This leads to a false legitimization of the data in most criminal cases due to the belief that it is scientific (Jefferson 2017). Realistically, the evidence of objectivity stops there. Objective crime tracking models become subjective quickly when the data input and the algorithm development are derived from human beings. Studies have demonstrated that implicit bias is inherent in all human beings and it shapes the input data that enters these systems, in other words implicit bias determines who gets targeted (Barocas and Selbst 2016).

To mitigate or be aware of this bias, new technological capabilities need to acknowledge race in data input and output and consider how these capabilities may affect marginalized communities (Ferguson 2017). At this time the field of crime analysis for law enforcement has not been fully subjected to critical geographic scrutiny (Jefferson 2017).

2.4.3 The influence of media on perpetuating fears

For most Americans the time spent on TV, movies, and videogames far surpasses the time spent at work, or with family and friends. The influence of media continues to grow as consumers become less aware of how the media operates, how news decisions are made, and of the role media plays in influencing an individual's perceptions of the world (McCall 2007). As the media leans more towards sensationalism and profit, consumers are provided with content that is subjective and leads them to believe that existential threats are close by. This leads individuals to feel like they must protect themselves, even though the threat is far removed (McNaughton-Cassill 2017). The result is an increase in suspicion amongst individuals and perceptions of bias towards those portrayed negatively through the media, which are typically marginalized communities.

Another form of media, social media, is a platform created to bring people together to engage in meaningful conversations and moments of sharing. This is the case for popular application such as Facebook, Twitter, and Snapchat. Additional applications are also available that promote neighborhood cohesion such as the aforementioned Nextdoor and Naber. The integration of these applications into crime detection and reporting introduces varying amounts of bias due to the lack of validation of each crime report. In addition, the availability of these platforms can skew the number of reports due to the disproportionate usage across demographic groups. As an example, Masden and colleagues (2014) explain that most users of the Nextdoor app are usually active in their communities, and they have easy access to these types of platforms. In these cases, it proves that those with the most resources are more likely to report crimes, while marginalized communities with little access and social cohesion are more than likely to report fewer crimes and be ones reported on.

2.5 Analysis of individual perceptions of crime and GIS

As mentioned by Kwan (2000), Geographic Information Systems, and mapping through various platforms is useful when studying large and complex datasets with multiple attributes, where inferential statistics and pattern recognition algorithms may fail. Past studies, including those mentioned in the prior subsections, analyze individual fear of crime statistics without an inference to time and space, or research spatiotemporal responses on a limited scale. Mapping the fear of crime delivers the baseline data and a localized means to analyze individual perceptions; however, with the addition of an EMA real time component, and connections to active mapping services, the integration of GIS data into an analytical framework can dynamically reveal the differentiation of individual perceptions to actual crime rates. The integration of demographic data and spatially joining GIS layers opens the door to understanding multiple perspectives from respondents who come from different demographic backgrounds and may help to understand the extent of individual bias in fear of crime perception data.

This study leverages methods used in past studies to develop a mobile application that generates geospatial data to measure fear of crime perceptions in real time. Generating real time data that can be used for subsequent GIS analysis, integrating Google maps and Firebase backend services, issuing dynamic surveys, and distributing this application through Google Play services opens the door for future analytical research by a larger pool of users, which was the main drawback the Solymosi and Chataway studies.

Chapter 3 Planning and Methodology

Shifting the focus from a static to a situational approach when analyzing human perception of safety required the development of tools to measure these perceptions. This chapter summarizes the process of development for an Ecological Momentary Assessment (EMA) tool that integrates Google Maps with the Android Mobile platform to track individual perceptions of safety on a dynamic scale through space and time. Specifically, this chapter covers the goals of the application, user requirements, application functionality, UX design, software/hardware used, and the data used to perform a cohesive analysis.

3.1 Application Requirements

The successful deployment and functionality of this application is contingent on choosing the platforms that would be most effective for building the application, including the operating system, mapping Application Programming Interface (API), Software Development Kit (SDK), and Backend Services. For the purposes of this study each platform was required to be widely distributable, scalable, and low cost. In addition, since this application required that a diverse set of users were able to operate it, user friendly interfaces and operability were are also prioritized.

3.1.1 Application goals

The primary driver behind the development of this EMA mobile GIS application was to create a tool that provided the ability to crowdsource data from a diverse set of participants and integrate the resulting data into a real time database. This application was developed as a widely distributed spatiotemporal data collector that provided the ability to track participant perceptions dynamically. The goal of the application was for a user to submit spatiotemporal and perception information by choosing a location from the map and answering four perception-based survey questions. Once the survey was complete, the next step was to synchronize data to a real time database and export to a JavaScript Object Notation (JSON) data interchange format. The location, temporal, and perception data within the JSON file was then parsed and integrated into multiple GIS software platforms for analysis, specifically to enable future comparisons of perception data with universal crime data using such tools as the Getis-Ord Gi statistic to define areas of intensity (high and low values) on a micro scale.

3.1.2 User sample

The effectiveness of this mobile application and its implications for further research was based on input from a diverse set of human subjects who were instructed to input data using location-based map views and subsequent survey activities on an Android platform. The studies by Solymosi and Chataway focused on specific groups, such as college students, but the intent of this application was to expand the user base to obtain perception of safety information from a broader population sample, including participants from different neighborhoods, economic backgrounds, race, gender, and age range. By including a diverse set of users, this mobile application expands future research opportunities for analysts to compare user demographic data with existing crime data and American Community Survey (ACS) demographic data to determine how demographically diverse environments affect participation and participant perceptions.

In order to run the GeoBAT application users were required to provide a valid e-mail account to install the application from the Google Play Store and to log into the GeoBAT application. Once a user name was established, the user was prompted to provide a name associated with the account and instructed to provide a password. User authentication was necessary to ensure user data was private and write access was limited only to users who possessed authentication credentials. Authentication was handled through Google Firebase, where its implementation into this EMA application will be covered in later sections.

Due to requirements set forth by the Institutional Review Board (IRB) and for this study, each user must have been over 18 years of age, resided within the city limits of Albuquerque, and had a general understanding of how to operate an Android mobile application. This includes a general understanding of Google Maps, and the ability to follow directions and read prompts generated from the application. Therefore, visualizing small font and the comprehension of the English language was required for this study.

3.1.3 User Requirements

The core functionality of this application was to provide users with a tool to generate spatiotemporal and perception of safety data by: (1) providing a map template and location button that could be referenced by users to find device locations or locations of interest; (2) initiating the Map Activity at the participants device location with a marker and information window that displayed location, time, and a prompt to take survey; (3) providing the user with functionality to add or drag a marker to an alternate location as the information window updated values; (4) creating an event from the information window to drive to the Survey Activity, and providing a template to answer four perception based questions; and (5) accessing a backend service to authenticate users, generate notifications, and store latitude, longitude, time, and survey values for subsequent geospatial analysis.

The convergence of a mobile application platform with the requirements for a comprehensive Ecological Momentary Assessment provided the tools to assess a participant’s perceptions of safety in real time and in a real-world setting. The development of a map interface, interactive symbols, transitional mechanisms, and responsive backend services were essential components to ensure a full assessment of participant moods could be recorded and analyzed.

3.1.4 Platform requirements

Users of this application had to meet specific technical/platform requirements to qualify for testing. First, since this application was built on an Android platform, participants were required to possess Android devices with a minimum version of API 23: Android 6.0 Marshmallow. This minimum API version was chosen because it streamlined the process of handling location permissions at runtime instead of when users installed and upgraded the application. Even though this version covers only 71% of devices on the market (Android Developers 2018), it ensures users have flexibility when managing permissions and access to location data.

3.1.5 Data collection protocols

Participant data collection followed two types of EMA data collection protocols: event contingent and signal contingent (Burke 2018). Event contingent was based on a predefined event that causes the participant to add data and includes events where participants volunteer to input data outside of their notified time frame. Signal contingent was based on predefined notifications that are part of the applications functionality. The notification signals were temporal reminders for participants to add their locations and add data to the survey.

For this study signal contingent notifications were sent through the Firebase Cloud Messaging platform. Participants received three to five notifications daily within respectable timeframes, approximately one notification late morning, three during the afternoon, and two in the early evening. Event contingent protocols were followed when participants were active during “off” hours and volunteered their perceptions out of the normal notification time frame.

3.1.6 Application Functionality

The mobile GIS application developed for this study goes by the title of GeoBAT (Geographic Behavioral Analysis Tool) and was designed to enable input from users to report on physical or social disorders based from their perceptions of safety. Physical disorders are those that are aesthetic; an example would be graffiti and rundown environments, and social disorders are observed human behaviors; an example would be public drunkenness or suspicious persons. This application measured a participant’s perceptions when reporting and uses emotion-based measures to determine if the responses exhibited bias as a factor in their reporting.

In order to provide functionality for users to collect data, and for the data to be saved, the application provided dependencies to support Google map display and Google Firebase implementations. In addition, permissions needed to be specified in the Android Manifest allowing for Internet and location access. Integrating these implementations, dependencies, and permissions on an Android platform allowed the user to authenticate, specify their location on a map, and submit survey data to a real time database.

3.2 Application Platform and Design

Developing an EMA mobile GIS tool required the integration of multiple software tools, platforms, and an easy to interpret design layout. There were many choices when determining which operating system (OS) to use, the optimal Integrated Development Environment (IDE), mobile backend as a service (BaaS) provider, mapping API’s, and testing devices. For this study the choice was determined by the ratio of performance to cost. The following subsections provide a description of the tools that were used to develop this application and why they were chosen.

3.2.1 Integrated Development Environment

Android Studio is the official IDE for Android application development and is built upon IntelliJ IDEA's code editing and developer tools (Android Developers 2018). The versions of Windows, Android Studio and Java used to develop the GeoBAT application respectively were Windows 10 64-bit, Android Studio v.3.1.4 and Java SE 8. Android Studio is open-source, and it was a primary choice, due to its robust set of tools that could be used to quickly and cheaply deploy an Android application for testing and eventual production release.

3.2.2 Operating System

This application was developed on the Android OS, which is a mobile operating system developed by Google. The Android OS can be distributed on multiple devices, and development supports the Java programming language, which is widely used and documented, as opposed to Swift, which is specific to one platform, and is the primary programming language for iPhone OS (iOS). The focus is not on which OS is superior to the other, but on which OS works best for the components of this study. The ability to develop on a multiplatform environment is one of the reasons the Android OS was chosen.

The Android OS was also chosen due to the vast amounts of documentation, both in Android development and the Java programming language. This capability was enhanced by a robust user community, specifically through Android Developers and Stack Overflow, where a wide range of issues have been encountered by developers and addressed by the community. Most importantly, mobile applications developed on the Android OS are easily distributable through the Google Play store. Android apps are published in hours, unlike the Apple store, which takes weeks to review an application, with the possibility of rejection after the waiting period. For the purposes of this study it was important to keep costs low and distribute testing versions to the public as quickly as possible. Using the Android SDK and Android Studio IDE enables the developer to process a signed Android Package (APK) file, pay a one-time $25 fee, and publish the application to the Google Play store in minutes. Publishing to the iOS store is a little more expensive, however Apple is more stringent on approving applications (Viswanathan 2018), which made the decision easier to publish from an Android environment.

3.2.3 Firebase

Google Firebase is a mobile and web application platform that provides developers with tools and services to provide fully functional applications to their user base (Hackernoon 2017). Firebase is a mobile backend as a service provider that optimizes a developer's ability to add authentication, real time database, storage, and cloud messaging functions to the application through connection to the Firebase platform. The GeoBAT application leveraged Firebase's authentication, real time database, and cloud messaging services to authenticate users upon sign in, store and sync user information in real time, and send notifications to complete the provided survey. Chapter Four provides additional detail on the process used to integrate the Firebase platform into the application and how it is applied to the GeoBAT application.

3.3 Data Description

The main functionality of this application provides researchers with an EMA tool to analyze perceptions of victimization across varying times and places. Data generated from this mobile GIS tool included the location and time of the assessment, and tabular data that analyzed the participants feelings at the time of reporting. Additional data layers acquired for the development of this application included maps provided from Google Maps SDK for Android and the World Street Map.

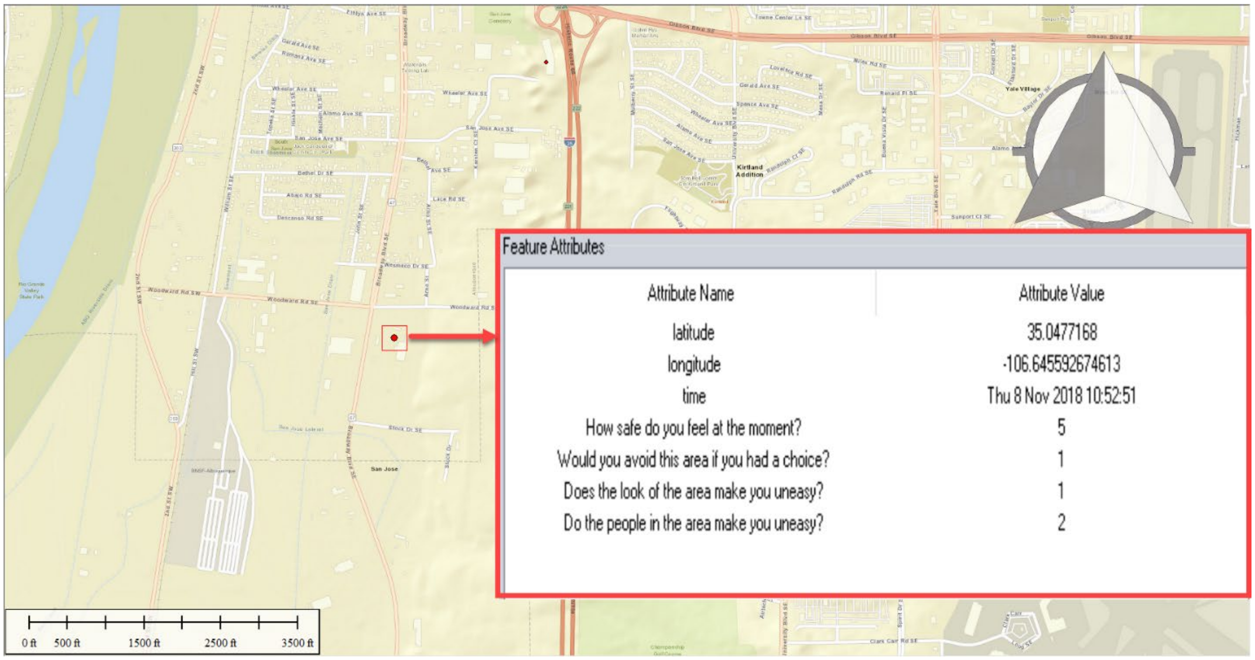
Data collected from the application was synced to a real time database, and the data was parsed from a JSON file for import into multiple GIS platforms as point data. Each point contained an attribute table with location, time, and participant input from the survey. With this information the data could be imported into multiple map platforms for subsequent analysis. The goal is to be able to use this crowdsourced GIS data in the future to provide benchmarks for city leaders and planners to reference when attempting to understand the perception of crime trends and comprehend how the design of an urban area can influence human perceptions. Table 1 lists the required data for this study, and the following subsections briefly cover how each dataset was acquired and used.

Table 1. Data requirements to provide analytical data

Layer	Source	Availability
VGI data	Human subjects (minimum of 5 test subjects to ensure the application is functional.)	Based on availability of participants
Normal, Satellite, Terrain, and Hybrid map displays	Google SDK	Free up to 25,000 map loads per day/$0.50 USD 1,000 additional map loads
World Street Data	Esri Map Service	Open source

3.3.1 VGI data

This mobile GIS application was developed as a platform to allow users to enter EMA data and assess their perceptions of crime on a spatial and temporal scale. Each entry that was committed by the user was a data point that contained spatiotemporal and perception information which could then be imported into a GIS platform. Information from each user was parsed from a JSON file that had been exported from the Firebase real time database and imported as a point feature into Blue Marble Global Mapper v.19 and ArcGIS Pro (Figure 3.1).



Attribute Name	Attribute Value
latitude	35.0477168
longitude	-106.645592674613
time	Thu 8 Nov 2018 10:52:51
How safe do you feel at the moment?	5
Would you avoid this area if you had a choice?	1
Does the look of the area make you uneasy?	1
Do the people in the area make you uneasy?	2

Figure 3.1. User collected data point (Global Mapper)

3.3.2 Google Maps

The Google Maps SDK (Software Development Kit) offers five map types to display. This application allowed the user to choose these map types from a pull-down menu provided in the map activity: (1) Normal display portrayed physical features, as well as natural features with labels on a transparent backdrop; (2) Hybrid display combined satellite data with the features portrayed in the normal display; (3) Satellite display was stand-alone imagery with no underlying features; (4) Terrain display included topographic data, which included colors, contour lines and labels, and perspective shading; and (5) None rendered an empty space. (Google Maps Platform 2018).

3.3.3 World Street Map

World Street Map is an open source base map provided by Esri. This base map was used in Global Mapper and ArcGIS Pro to assess if point data was positioned correctly, and as a general map reference for this study. World Street map includes most physical and natural features including transportation, structures, water features, and boundaries to name a few. In addition, the base map is compiled from a variety of sources including multiple government agencies and the GIS community (ArcGIS 2018).

3.4 UX Wireframe Design

Prior to the development of the mobile application a user experience (UX) design for application functionality was implemented using the Adobe XD Creative Cloud software platform. To ensure the mobile application met the standards for this proof of concept a wireframe design and interactive testing process was developed to visually plan the steps required to ensure the user experience was relevant to the goals of this study (Figure 3.2). The wireframe design provided a prototype of the application, while the interactive testing process provided a preview of component functionality, and Android Activity interactions, on a physical device.

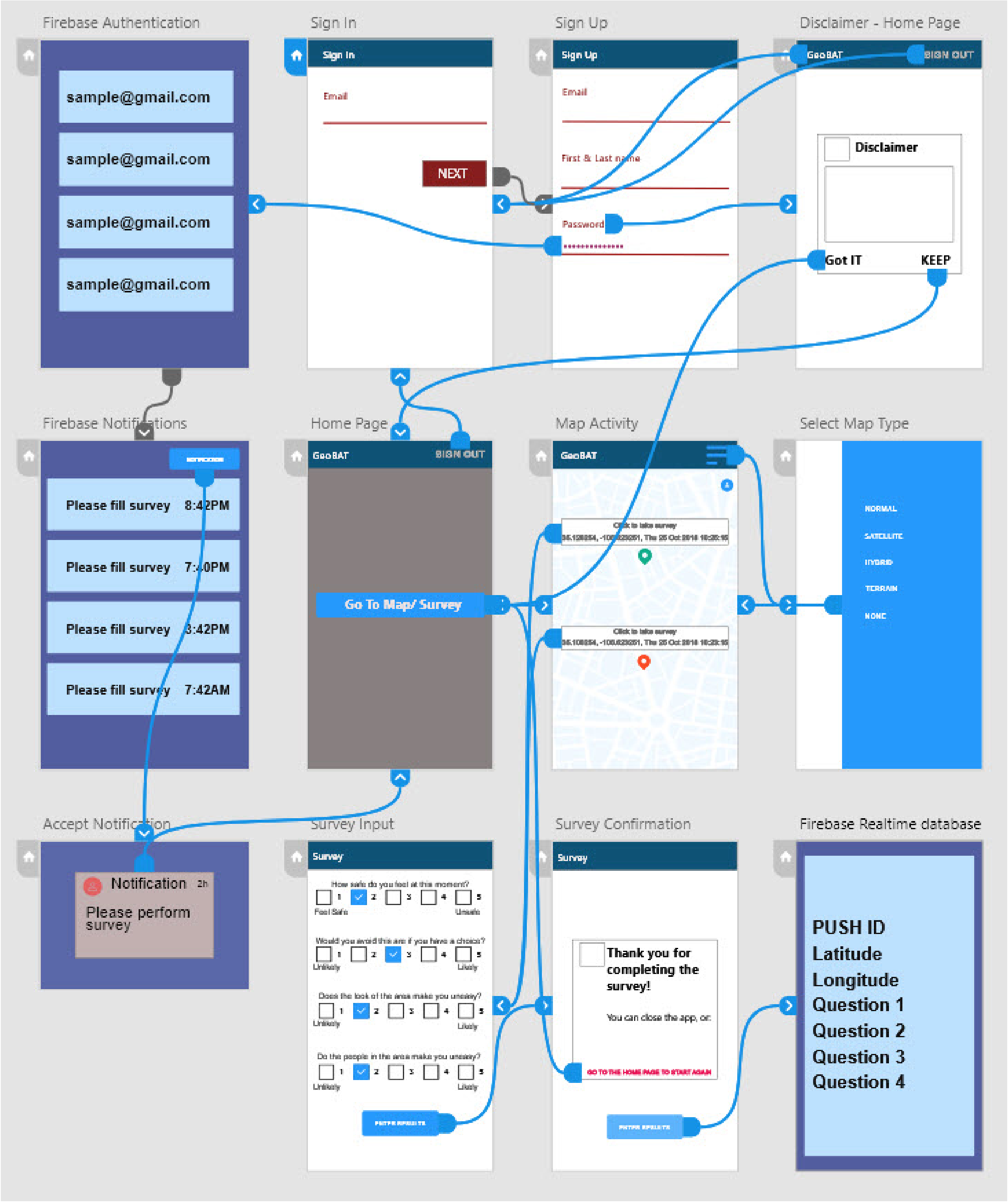


Figure 3.2 Application wireframe concept

Chapter 4 Application Development

The development of a geospatial EMA tool on an Android mobile platform required connections to multiple dependencies and services that provided the functionality needed to execute this project. Most important was the integration of requirements to implement connections to the Android, Google Maps and Google Firebase API's to provide access to location, map, authentication, and real time database platforms. A general overview of setting up the application environment was covered in the Planning and Methodology chapter, including the types of platforms used to develop the application, the UX functional design, and a brief description of BaaS services used to transfer data. This chapter covers the technological process in the development of the GeoBAT application and defines the processes used to connect the application to services required to make it completely functional.

4.1 Android Architecture

Within the Android operating system is a built-in architecture that is tightly integrated and carefully tuned to provide the optimal application development and execution environment for mobile devices (Smyth 2017). The Android architecture is open source and structured in the form of a Linux-based software stack that contains, from the bottom, the Linux Kernel, Hardware Abstraction Layer (HAL), Android Runtime (ART), Android Libraries, the Application Framework, and Applications. The Linux Kernel provides underlying functionality and is the foundation of the Android platform; the HAL contains multiple library modules that implement an interface for a specific hardware component; ART translates compiled data to native instructions required by the device processor for faster performance; Android Libraries are Java-based libraries specific to the Android environment; the Application Framework is a set of services that form the environment in which Android applications are managed; and Applications are comprised of native and third-party applications (Figure 4.1).

Understanding how the Android development structure functions is a basic prerequisite to mobile application development. In addition, understanding how each element of the stack relates to the requirements of the mobile application ensures that functionality meets the standards set forth by the developer and meets the demands of the user.

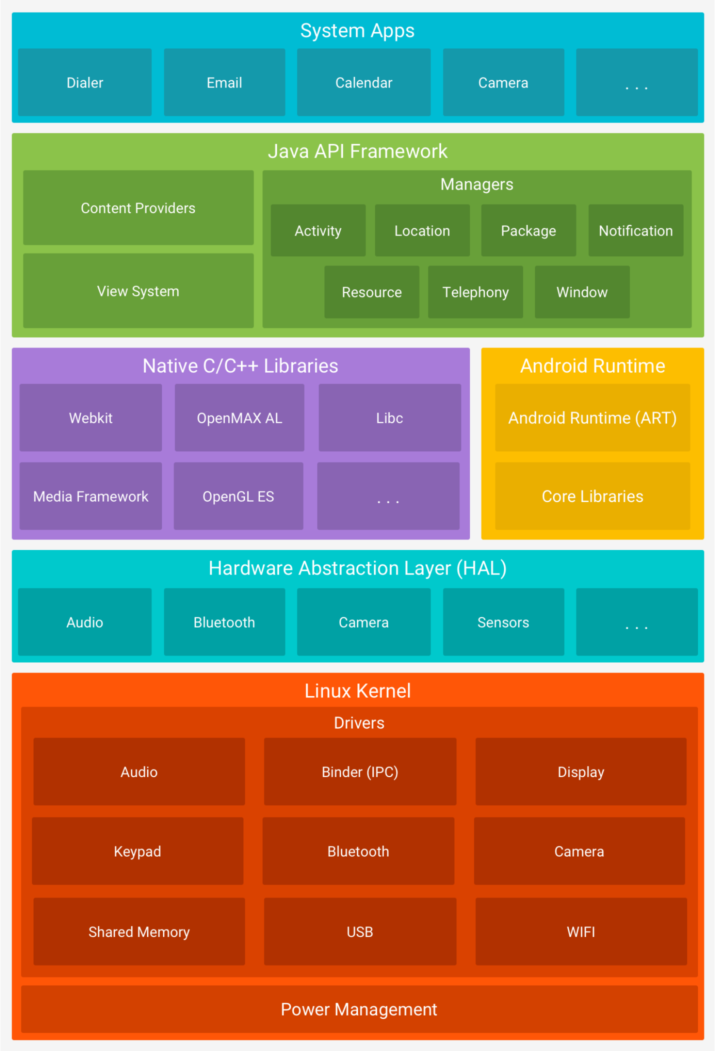


Figure 4.1. The Android Software Stack (Android Developers, “*Platform Architecture”* 2018)

4.2 Android Target Versions

Choosing a target SDK that fits the latest version is recommended when developing a new application, however if the application has alternate requirements, the option to target earlier versions is allowed. The goal of application development may be to target a large percentage of users, or to target versions that optimize the functionality of the application.

This application was built and compiled on Android 8.1 (Oreo) and the target SDK version was API level 27. Prior to and during the development of this mobile application Android 8.1 (Oreo) with a target SDK of level 27 was the latest version. The minimum SDK version used for this application was Android 6.0 (Marshmallow) API Level 23.

At the time of this study, as was mentioned in Chapter 3, the application developed for this study was distributable to 71% of Android users (Android Developers, *"Distribution Dashboard"* 2018). Reducing the minimum SDK would add to the user base, however choosing this option would impede the process for users to grant permissions, as each user would have to grant permissions upon install, instead of when the application is running. Also, as of August 1, 2018, the Google Play Store stopped accepting new applications below a target SDK of level 26, and these factors helped in the decision to develop an application that targeted three quarters of Android users and provided the functionality required to execute this study.

4.3 Application Development Environment

Android Studio supports Android application development and provides the tools necessary to create and deploy applications for all Android devices and a wide array of users. These Android Studio tools include a gradle-based build system, an emulator that can emulate multiple screen configurations, a unified environment to develop for all Android devices, instant run to push changes in real time, and GitHub integration to name a few (Android Developers, *“Meet Android Studio”* 2018).

The implementation and use Android Studio to develop an EMA mobile application for this study required the installation of the Java Development Kit (JDK) prior to Android Studio installation. The version of Java used for this study was 1.8.0_171. At the time of development this was the latest version, however subsequent versions have been released as the project progressed. Once the JDK was installed an environment variable was set up to point to the JDK installation and clear the way for the installation of Android Studio version 3.1.4.

With the installation of Android Studio, integration of the Android SDK, and the installation of the appropriate support repositories, the process for mobile application development was initiated. The following subsections cover set up and initial development of the GeoBAT application.

4.3.1 Set up virtual and physical device for testing

Testing Android applications was made possible through the Android emulator, which emulated virtual devices in Android Studio. Multiple virtual device configurations could be tested by choosing the target device or devices through the Android Virtual Device (AVD) manager. Additionally, testing could be performed on single or multiple physical devices when connected to the Android Studio platform.

For this study virtual device set up required the installation of the Intel Emulator Accelerator (HAXM installer) version 6.2.1. The installation of this software was performed by selecting the Intel Emulator Accelerator in the SDK manager and running the haxm_check executable in the sdk folder. Installing this software ensured that the applied virtual devices were operational, and the device performance was optimal for testing. The virtual device used to test the functionality of the GeoBAT application was the Nexus 5X, due to its compatibility with the Google Play platform and access to Google Maps.

Adding a physical device for testing required the activation of developer options on the device, enabling the USB debugging feature, and granting permissions to connect the device to Android Studio. A Samsung Galaxy s3 tablet was used as the primary physical device to test this application, while the virtual device was used to test normal device configurations.

4.3.2 Gradle build system

Android Studio uses an advanced build toolkit called Gradle to compile app resources and source code, and packages them into APK's that can be deployed, signed, and distributed (Android Studio, *"Configure your build"*, 2018). Within the development environment, a newly created project is populated with two gradle scripts and associated configuration files. The scripts share the same name and appear in the Android scope as project and module. The top-level project build file specifies paths to specific plugins and linked repositories, while the app-level module build file specifies declarations to application ID's, target Android SDK's, versioning, and dependencies; which are a listing of different components the application depends on.

When this project was created two class paths were added to the top-level build file, including the gradle build version and Google services version, which were v.3.2.0 and v. 4.0.1 respectively. The Google services gradle plugin was required to enable Google API's and Firebase services. A link to Google maven repository was added to integrate Google's support libraries into the project.

The app-level build file specified the target SDK version as well as the minimum SDK version, which were v27 and v23 respectively. In addition, the package name was specified, and the version code was initialized to version one. This module implemented over one dozen services including different components of Google Play services, Google Maps, and Google Firebase. Implementing these components provided the functionality required to successfully build and run the application developed for this project.

4.3.3 Android Manifest File

At the center of Android mobile application development is the Android Manifest file. The Android Manifest file is an Extensible Markup Language (XML) file that provides information about the application to the application framework. This includes declaring the application package name, the components of the application, permissions the application needs to access system resources, and hardware/software features that the application requires to function (Android Developers, *"App Manifest Overview"*, 2018).

The components of the Android Manifest file generated for this EMA application included the package name, permissions to access the internet and device location, variables to control the behavior and appearance of the application to the user, registration of three activities that refer to each java class, and meta-data tags to embed Google Play services versions and add the Google Maps API key to the application.

4.3.4 Layout files

Each activity defined in the Android manifest refers to an XML layout file that defines the appearance of each activity. As mentioned in the prior chapter, three activities were integrated in the mobile application design. This included the Main Activity, Map Activity, and Survey Activity.

First, the Main Activity was defined as the launcher activity and was set up to be launched by the user without receiving any data. The corresponding layout file assigned to this activity was the activity_main.xml file. Within this xml file, the parent view, Constraint Layout was selected to ensure all views conformed to constraint settings, therefore improving layout performance. An image view was added to display a background image, and a Go to Map button was added to direct the user to the next activity. From the Main Activity an intent object from the Intent class was called to switch to the Map Activity; this action could be completed by pressing the Go to Map button. The corresponding layout file assigned to the Map Activity was activity_map.xml, and within this xml file the Relative Layout was assigned for Google Maps display. Inside the Maps Activity an intent object from the Intent class was set up to be called when the title snippet was selected from the location marker to transition to the Survey Activity. Finally, the Survey Activity was assigned to the activity_survey.xml, and Constraint Layout was selected as the parent view. Multiple text views and check boxes were added on top of the Constraint Layout; in addition, a Scroll View was added to adapt to multiple screen configurations, and an Enter Results button was added to confirm survey completion.

4.3.5 Alert Dialogs

Dialogs are small windows designed for users to interact, or in the case of this study, dialogs were designed to present a disclaimer upon sign in, and to confirm the survey data entered by the user was successfully collected.

AlertDialog is a class that provides API's to create AlertDialog windows, and this class was used to build dialog designs in the Main Activity and Survey Activity. A dialogShow method was created in the Main Activity to develop a disclaimer dialog and provide the user with a message, the option to dismiss the dialog permanently, or keep the dialog as reference. An instance of the SharedPreferences class was called within the getSharedPreferences method to provide the users with options to never display again or store the dialog upon session start up.

The dialogShow method was called from the onCreate method to display upon the launch of the Main Activity.

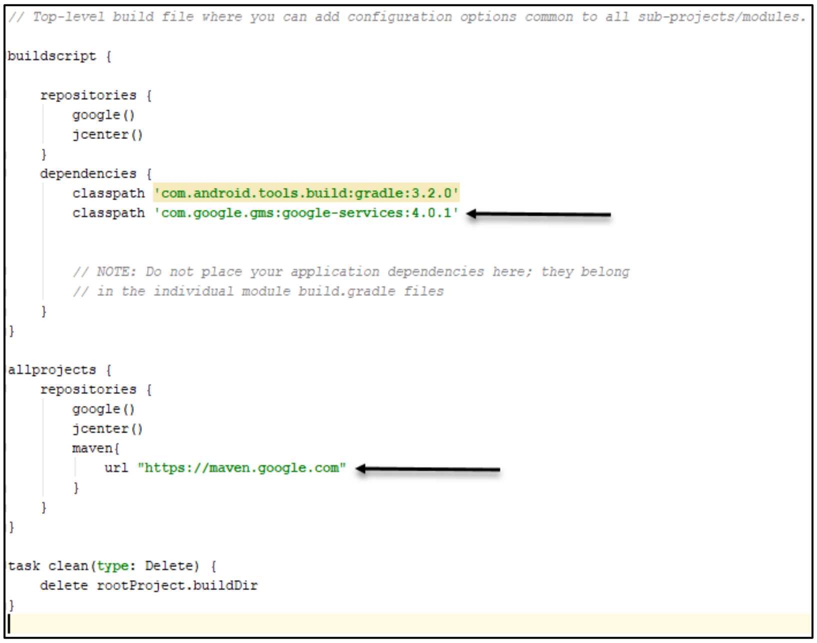
In the Survey Activity a simple dialog confirming the survey was completed successfully was embedded in the onResultPressed method. This method was developed to send information entered by the user to the Firebase real time database and to display the dialog. An intent within the setPositiveButton method provided the user with a choice to return to the home page, or Main Activity.

4.4 Google Maps API

Google Maps integration provided users of this application with the resources to reference their location using various map layouts to specify their location when reporting their perceptions of safety. Connecting this API into an Android development environment required the installation and implementation of multiple services that were used to provide added mapping functionality to this application. Once services were installed, and the components were implemented in the gradle build system, the components of the map and underlying functionality could be developed.

4.4.1 Connecting to Google Maps API

Adding the map functionality into this Android project required the installation of the Google Play services SDK in Android Studio. These installs and updates were performed through the Android Studio SDK Manager and gradle build files. First, within the Android Studio SDK manager, the installation of Google Play Services and the Google Repository was required in order to provide a connection to the Google Maps API. Second, a url and plugin reference was implemented in the top-level build.gradle file to the maven repository, and map and location dependencies were implemented in the app-level build.gradle file (Figure 4.2 and 4.3).



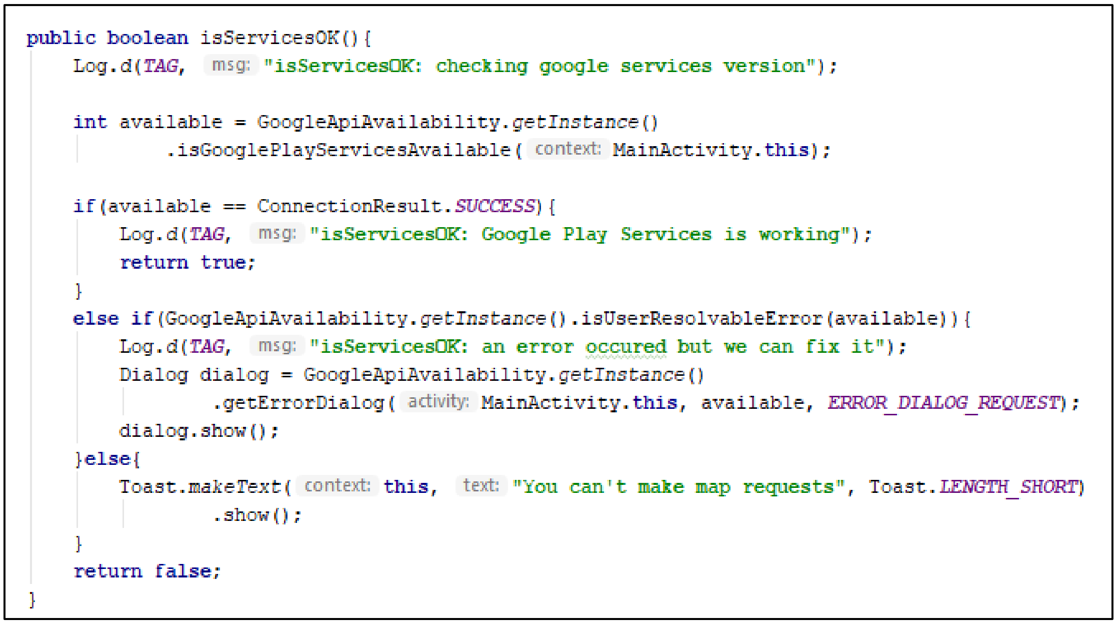
```
// Top-level build file where you can add configuration options common to all sub-projects/modules.

buildscript {

    repositories {
        google()
        jcenter()
    }
    dependencies {
        classpath 'com.android.tools.build:gradle:3.2.0'
        classpath 'com.google.gms:google-services:4.0.1'


        // NOTE: Do not place your application dependencies here; they belong
        // in the individual module build.gradle files
    }
}

allprojects {
    repositories {
        google()
        jcenter()
        maven{
            url "https://maven.google.com"
        }
    }
}

task clean(type: Delete) {
    delete rootProject.buildDir
}
```

Figure 4.2 Top-level build.gradle file; Google API connections

```
apply plugin: 'com.android.application'

android {
    compileSdkVersion 27
    defaultConfig {
        applicationId "com.thegeobat.geobat"
        minSdkVersion 23
        targetSdkVersion 27
        versionCode 3
        versionName "1.0.2"
        testInstrumentationRunner "android.support.test.runner.AndroidJUnitRunner"
    }
    buildTypes {
        release {
            minifyEnabled false
            proguardFiles getDefaultProguardFile('proguard-android.txt'), 'proguard-rules.pro'
        }
    }
}

dependencies {
    implementation fileTree(dir: 'libs', include: ['*.jar'])
    implementation 'com.android.support:appcompat-v7:27.1.1'
    implementation 'com.android.support.constraint:constraint-layout:1.1.3'
    implementation 'com.google.firebase:firebase-database:16.0.3'
    implementation 'com.google.firebase:firebase-core:16.0.4'
    implementation 'com.google.firebase:firebase-auth:16.0.4'
    implementation 'com.firebaseui:firebase-ui-auth:4.2.0'
    implementation 'com.google.firebase:firebase-messaging:17.3.3'
    implementation 'com.google.android.gms:play-services-maps:16.0.0'
    implementation 'com.google.android.gms:play-services-location:16.0.0'
    implementation 'com.google.maps.android:android-maps-utils:0.5'
    testImplementation 'junit:junit:4.12'
    androidTestImplementation 'com.android.support.test:runner:1.0.2'
    androidTestImplementation 'com.android.support.test.espresso:espresso-core:3.0.2'
}

apply plugin: 'com.google.gms.google-services'
```

Figure 4.3 App-level build.gradle file; Google API connections

Finally, an isServicesOK method was created in the Main Activity class to check if the user had the correct version of Google Play services. If a connection was granted a log was created to confirm the connection, and if the connection was unsuccessful, a log would also confirm this, along with a dialog that could be viewed in the application confirming the error. Figure 4.4 displays the logic behind the isServicesOk method, which was called from the onCreate method to ensure the user is warned about issues with connections to the API.



```
public boolean isServicesOK(){
    Log.d(TAG,  msg: "isServicesOK: checking google services version");

    int available = GoogleApiAvailability.getInstance()
            .isGooglePlayServicesAvailable( context: MainActivity.this);

    if(available == ConnectionResult.SUCCESS){
        Log.d(TAG,  msg: "isServicesOK: Google Play Services is working");
        return true;
    }
    else if(GoogleApiAvailability.getInstance().isUserResolvableError(available)){
        Log.d(TAG,  msg: "isServicesOK: an error occured but we can fix it");
        Dialog dialog = GoogleApiAvailability.getInstance()
                .getErrorDialog( activity: MainActivity.this, available, ERROR_DIALOG_REQUEST);
        dialog.show();
    }else{
        Toast.makeText( context: this,  text: "You can't make map requests", Toast.LENGTH_SHORT)
                .show();
    }
    return false;
}
```

Figure 4.4. Code to confirm connection to Google Maps API

After the implementation of components for the Google API were set, the retrieval of an API key was required to access the Google Map servers. The process to obtain the API involved the creation of a new project in the Google Cloud Platform console, developing credentials, enabling the Google Maps Android API, and copying the API key assigned from the console to the Android Manifest XML file. The Google Play version and API key were embedded in the Android Manifest file under the <meta-data> element.

Referencing and attaching the API key to this project enabled Google to track usage of their mapping services from this mobile application. Google Maps SDK for Android was the primary service linked to this application and provided unlimited free usage. However, additional services, such as the Places SDK, would incur a cost after 150,000 requests (Google Maps Platform, *"Pricing and Plans",* 2018). In the case of this application, and plans for further research and expansion, pricing details would be scrutinized to ensure Google does not alter their pricing structure to avoid exceeding possible data limits.

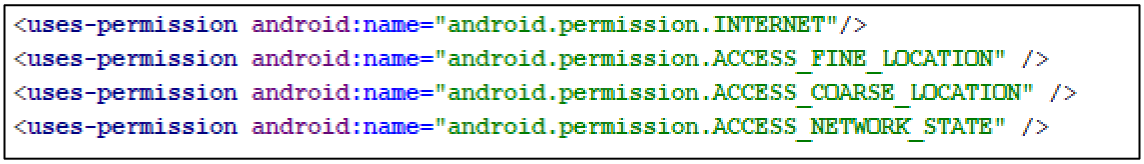
4.4.2 Implementing the map

The implementation of a map display from the Google Maps API required a new Java class named Map Activity, an activity_map.xml layout file, and a link to the <activity> element in the Android Manifest. Map Activity was developed as a class to control map display and modifications to the Google Map SDK, while the layout file was created to define the map view as a fragment. A fragment is a modular section of an activity that provides stand-alone behavior while the activity is running. In this case, the map had its own lifecycle, received its own input events, and could be added and removed while the activity ran (Android Developers, *"Fragments"*, 2018).

Within the Map Activity an instance of the OnMapReadyCallback interface was set on a MapFragment to trigger an instance of GoogleMap from the onMapReady method. A connection to the Google Map platform was dependent on the availability of the Google API platform and granted user permissions.

4.4.3 Request Permissions

Android applications must request permissions to access sensitive user data and certain system features (Android Developers, *"Permissions Overview"*, 2018). Initially, user permissions were set in the Android Manifest to allow the API to download map tiles from Google Map servers, check connection status, and access user location. This was specified with the <uses-permission> element (Figure 4.5).

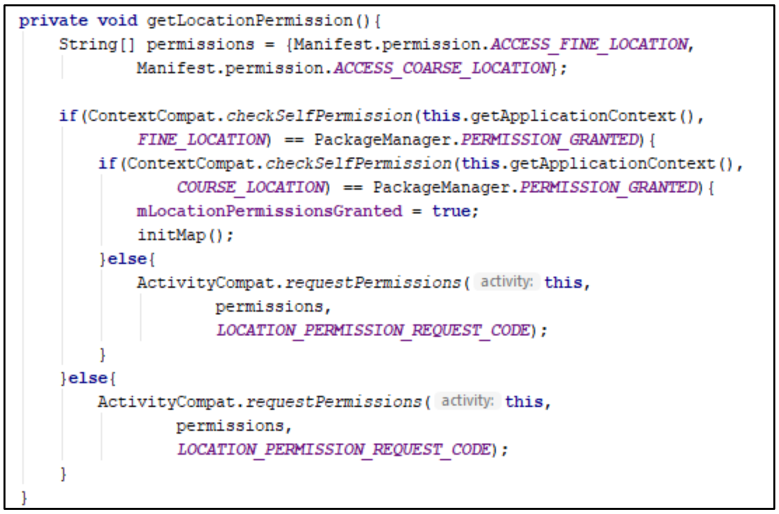


```
<uses-permission android:name="android.permission.INTERNET"/>
<uses-permission android:name="android.permission.ACCESS_FINE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_COARSE_LOCATION" />
<uses-permission android:name="android.permission.ACCESS_NETWORK_STATE" />
```

Figure 4.5 Permissions required to access Google Map and device location

Permission approvals are listed as normal or dangerous. If the permissions do not pose much of a risk to user privacy, the approval will be normal, and the system automatically grants access. However, if the permissions affect user privacy, the protection level is dangerous, and the user needs to approve access at runtime.

This study relied heavily on device location to detect user perceptions of safety at a specified place and time. Due to this requirement, code was integrated into the Map Activity to provide the user with the option to grant permission to access the map and their location (Figure 4.6).



```
private void getLocationPermission(){
    String[] permissions = {Manifest.permission.ACCESS_FINE_LOCATION,
            Manifest.permission.ACCESS_COARSE_LOCATION};

    if(ContextCompat.checkSelfPermission(this.getApplicationContext(),
            FINE_LOCATION) == PackageManager.PERMISSION_GRANTED){
        if(ContextCompat.checkSelfPermission(this.getApplicationContext(),
                COURSE_LOCATION) == PackageManager.PERMISSION_GRANTED){
            mLocationPermissionsGranted = true;
            initMap();
        }else{
            ActivityCompat.requestPermissions( activity: this,
                    permissions,
                    LOCATION_PERMISSION_REQUEST_CODE);
        }
    }else{
        ActivityCompat.requestPermissions( activity: this,
                permissions,
                LOCATION_PERMISSION_REQUEST_CODE);
    }
}
```

Figure 4.6. Code to check permissions

The code above called the ContextCompat.checkSelfPermission method to check if permissions were activated. If confirmed, the initMap method would be initialized to display the map. The initMap method was created with a call to the SupportMapFragment class, which places the map in the application. If permissions were denied a request permissions method was created to prompt the user for access.

4.4.4 Accessing user location

Providing the ability to generate a user's perceptions at their last known location, and displaying their location information, was performed by accessing the fused location provider. The fused location provider managed the underlying location technology and part of the location API's in Google Play services (Android Developers, *"Get the last known location"*, 2018).

In the Map Activity class an instance of the Fused Location Provider Client was created within a getDeviceLocation method. Accessing the fused location provider provided the opportunity to declare a location object and call the getLastLocation method. The location object was used to provide location data with latitude and latitude coordinates, and a moveCamera method was called to drive the map display to the device location. Finally, a Boolean value was set to display the icon for the current device location by calling the setMyLocationEnabled method and setting it to true.

4.4.5 Marker display and functionality

Developing this EMA mobile application required a marker to display user locations on the map, and an information window, otherwise known as a snippet, to display a Click to Take Survey title, latitude, longitude, the time of day, and function as a button to transition to the Survey Activity. Marker functionality was developed through the Marker and MarkerOptions classes, whereby marker and options objects were declared from these classes respectively to build on this functionality. An instance of the Marker class was used to define marker placement, while an instance of the MarkerOptions class was called to define the underlying functionality of the marker. Defining and building marker functionality was performed in the moveCamera method, which took in three parameters, a latlng object declared from the LatLng class, zoom, and title.

Upon map initialization a customized marker and information window was set to display at the user location, however functionality was added to the marker to fit the requirements of this study. A customized marker was created by adding an icon method to the options object and creating a bitmap descriptor using the Bitmap Descriptor Factory. The marker’s initial display was set to red, but the color is modified when additional events, such as dragging to a new location, are implemented.

Developing a dynamic marker required code to listen for and handle a variety of events. This included implementing the setOnMapClickListener to display a marker with updated latitude/longitude display on the information window when the user chose a different location. The setOnMarkerDragListener to enable dragging the marker and updating displays in the information window. Finally, the setInfoWindowClickListener to call an instance of the intent class, update location data, and define an intent object to switch to the Survey Activity.

4.5 Google Firebase

Google Firebase provides backend functionality for mobile and web applications, including user authentication, real time database, cloud messaging, analytics, storage, and crash reporting to name a few. Firebase provides tools to interact with application users over the network and provides developers with tools to track these interactions. Firebase was the primary Backend as a Service (Baas) platform for this EMA application due to its compatibility with the

Android and Google Maps API. Integrating all services from similar providers in one platform was ideal for this application, and fulfilled the requirements needed to successfully deploy for this study. Three main features of Firebase were used extensively in this study including, authentication, the real time database, and cloud messaging. In addition, Firebase analytics was used to track user engagement and retainment.

At the time of this study Firebase offered three pricing plans. The Spark Plan was free for up to 100 simultaneous connections and 10GB/month in data. Beyond the Spark Plan was the Flame Plan, at $25/month for up to 100K connections and 20 GB data usage/month. The highest plan was the Blaze Plan which was ‘pay as you go’ (Firebase, *“Pricing Plans”*, 2018).

4.5.1 Connecting to Firebase

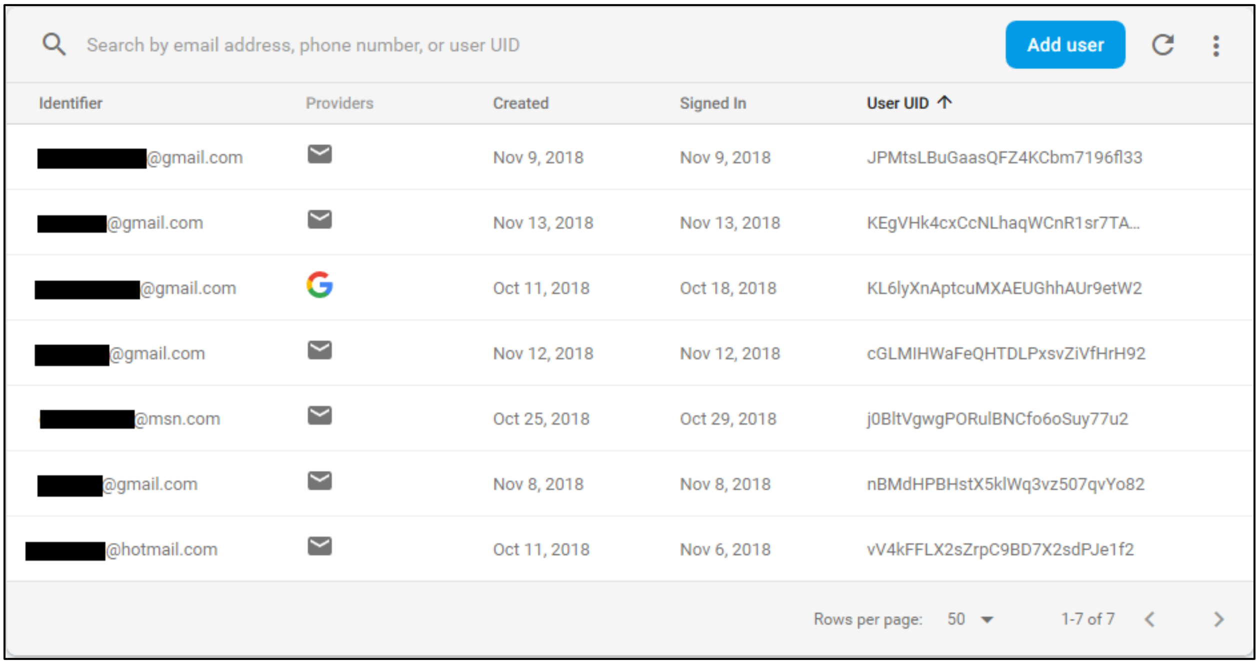
Implementing backend services into the EMA mobile application required a connection to a Firebase project. The first step was the creation of a project in the Firebase console that matched the name of the application and specified the package name assigned to this project. Second, a debug signing certificate was added to support Google sign-in authentication. Third, a google-services.json file was generated and copied to the project app directory. This file connected the client-side application with the Firebase project that handled the server-side components of the application (Udacity, 2018). Fourth, a google-services plugin was added to the top level build.gradle file, and dependencies along with the apply plug-in were added to the app-level build.gradle file (refer to Figure 4.2 for the google-services version next to classpath, and Figure 4.3 for dependencies and plug-in details). Finally, an option was provided to connect the application to Firebase. Successfully integrating Firebase with the application provided access to the backend features required for authentication, data storage, and notifications.

4.5.2 Authentication

The Firebase User Interface (UI) Authentication library was implemented is this EMA mobile application to provide an Email/Password sign up authentication scheme to users. This UI handled the UI flow for authentication with Firebase and implemented the best practices for sign in. Authentication was important to limit database write access to authenticated users only and identified who the users were. In addition, Firebase UI provided a platform to keep user credentials safe and secure.

The first step in implementing Firebase authentication required set up from the authentication tab in the Firebase console. A list of sign-in providers was provided with the option to select which providers to use for sign in. The Email/Password provider was the one chosen for this application.

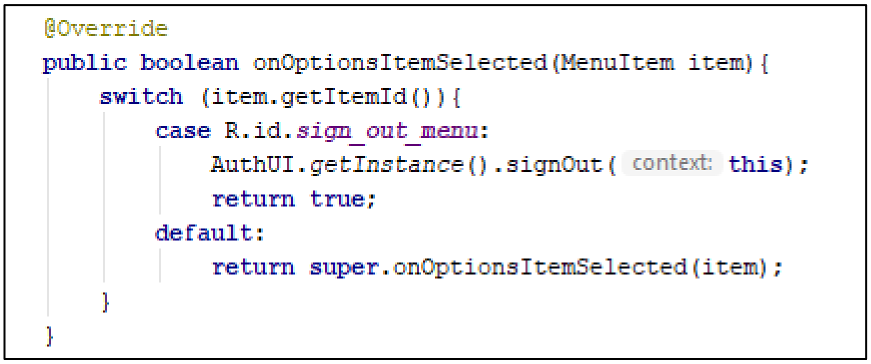
Second, an AuthStateListener was set up in the Main Activity class to attach to the Firebase auth object and execute whenever the authentication state changes. These state changes involved setting up the application upon sign in and tearing down the application once the user signs out (Udacity, *"Firebase in a Weekend"*, 2018). An instance of FirebaseAuth and FirebaseAuth.AuthStateListener classes were created, along with the associated instance variables. Within the onCreate method the Firebase auth object was initialized by calling the static method getInstance on FirebaseAuth. Additionally, the authentication state listener was initialized within the onCreate method and attached to the onResume method to add, and onPause method to remove. Finally, at the point where the authentication state listener was initialized, conditional statements were specified to check whether a user was authenticated or not. This provided the user with the option to log in and sign up and detected the user's authentication status when using the application. It also provided authentication information in the Firebase console to keep track of users by e-mail address and user ID (Figure 4.7).



Identifier	Providers	Created	Signed In	User UID ↑
[redacted]@gmail.com	✉	Nov 9, 2018	Nov 9, 2018	JPMtsLBuGaasQFZ4KCbm7196fl33
[redacted]@gmail.com	✉	Nov 13, 2018	Nov 13, 2018	KEgVHk4cxCcNLhaqWCnR1sr7TA...
[redacted]@gmail.com	G	Oct 11, 2018	Oct 18, 2018	KL6lyXnAptcuMXAEUGhhAUr9etW2
[redacted]@gmail.com	✉	Nov 12, 2018	Nov 12, 2018	cGLMIHWaFeQHTDLPxsvZiVfHrH92
[redacted]@msn.com	✉	Oct 25, 2018	Oct 29, 2018	j0BltVgwgPORulBNCfo6oSuy77u2
[redacted]@gmail.com	✉	Nov 8, 2018	Nov 8, 2018	nBMdHPBHstX5klWq3vz507qvYo82
[redacted]@hotmail.com	✉	Oct 11, 2018	Nov 6, 2018	vV4kFFLX2sZrpC9BD7X2sdPJe1f2

Figure 4.7 Firebase Console: Authentication List

Finally, a sign out menu was created from the onOptionsItemSelected callback by calling AuthUI, getting an instance, and calling the sign out method (Figure 4.8).



```
@Override
public boolean onOptionsItemSelected(MenuItem item){
    switch (item.getItemId()){
        case R.id.sign_out_menu:
            AuthUI.getInstance().signOut( context: this);
            return true;
        default:
            return super.onOptionsItemSelected(item);
    }
}
```

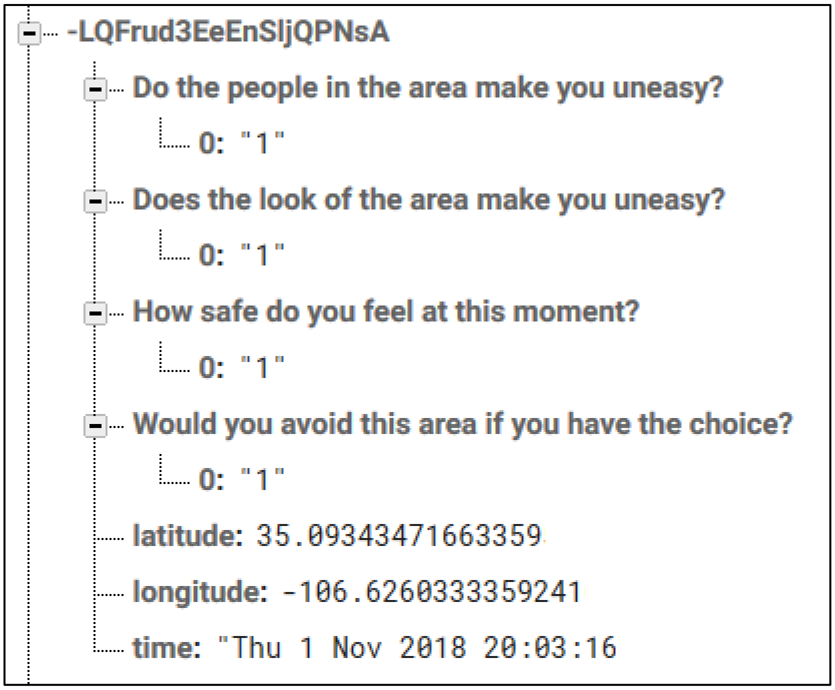
Figure 4.8 Code to display sign out menu and provide functionality to sign out of application

4.5.3 Firebase Realtime database

The Firebase Realtime Database API is a NoSQL cloud database that provides application developers with the tools to create applications that synchronize data in real time. Whenever data pushes from the application to the server, the data can be updated in milliseconds. Additionally, the Realtime database remains responsive when the user is offline by caching data in the application; when the user re-connects to the network the application automatically synchronizes with the Firebase servers.

All data synchronized to the Realtime database is stored as JSON objects. Unlike a SQL database, there are no tables or records (Google Firebase 2018); the data is structured in a JSON tree that contains nodes, user ID's, and associated key-value pairs. Keys are always Strings, while node values can have different types, such as integer, Boolean, String, or float. Unique keys, or push ID's, provide unique identifiers for each user to avoid conflicts.

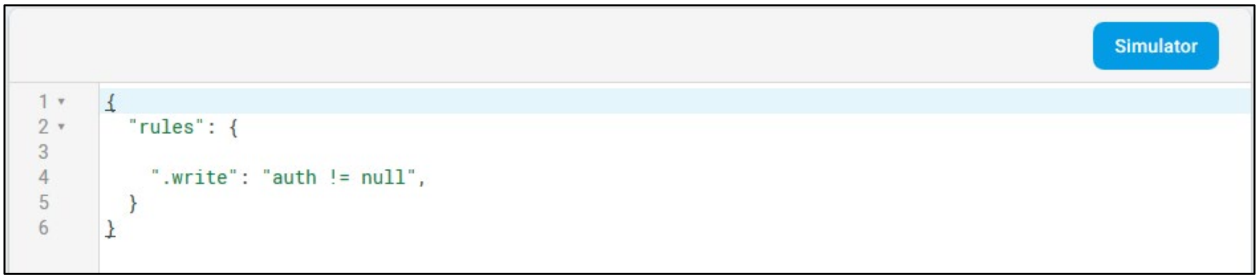
The integration of the Realtime database into this EMA application required the implementation of the Firebase Realtime Database dependency to the app-level build.gradle file. Next, two classes were added to the Survey Activity from the Firebase Database API. These two classes were the FirebaseDatabase class and the DatabaseReference class. An object was instanced from the FirebaseDatabase class to create an entry point to access the database, and an object was instanced from the DatabaseReference class to reference a specific part of the database. Within the Survey Activity class an instance of the DatabaseReference was created, and a myRootRef object declared, to push a unique identifier and all values required for this study to the server when the Enter Results button was pressed (Figure 4.9).



```
-LQFrud3EeEnSljQPNsA
    Do the people in the area make you uneasy?
        0: "1"
    Does the look of the area make you uneasy?
        0: "1"
    How safe do you feel at this moment?
        0: "1"
    Would you avoid this area if you have the choice?
        0: "1"
    latitude: 35.09343471663359
    longitude: -106.6260333359241
    time: "Thu 1 Nov 2018 20:03:16
```

Figure 4.9 Data synched to Firebase Realtime database

Finally, database rules were set to ensure authenticated users had write access only (Figure 4.10). These rules ensured that authentication was required to write data from the EMA application and deny access to users who were not authenticated. The application for this study had additional security features due to the implementation of a controlled tester list that determined who could install and authenticate, thereby providing additional limitations to the number of users with access to the database.



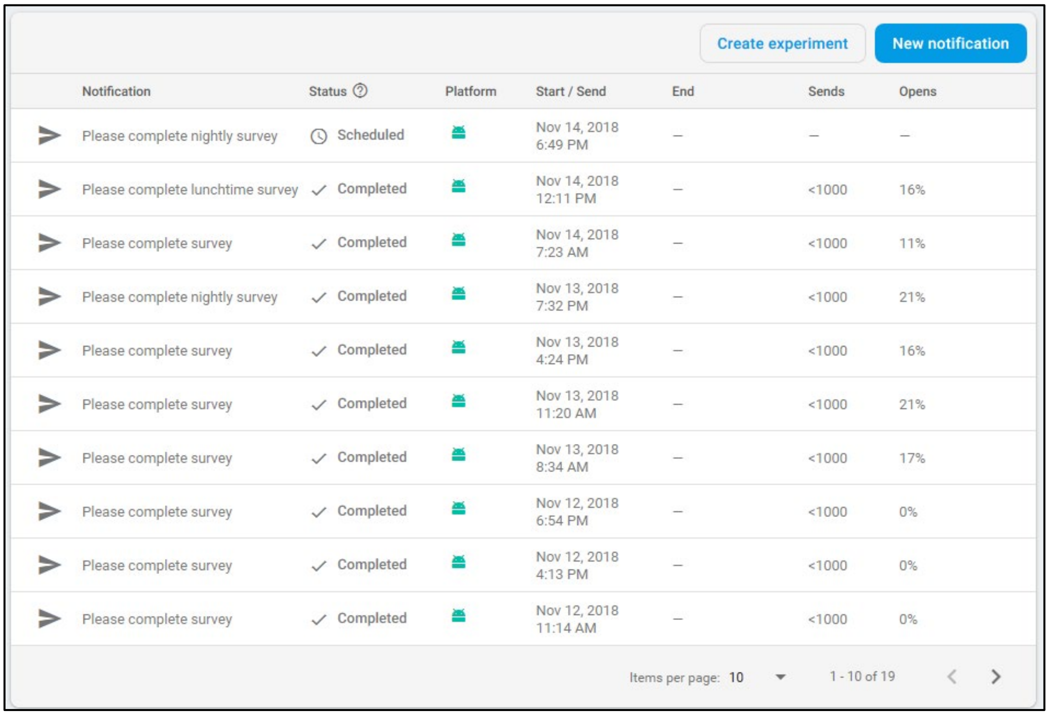
```
{
  "rules": {

    ".write": "auth != null",
  }
}
```

Figure 4.10 Realtime Database rules for authenticated user write access

4.5.4 Firebase Cloud Messaging

Firebase Cloud Messaging (FCM) allows developers to send notifications to a target application through the Firebase console. These notifications can be sent to all users, groups of users, or targeted to a single user of the mobile application.

Ecological Momentary Assessments on a mobile platform rely on timed notifications to remind users to enter perceptions of safety at their device or chosen location. In order to successfully deploy notifications to the users of this platform a connection was set to the FCM by implementing a messaging dependency in the project app-level build.gradle file. Enabling this dependency provided the resources to send notifications from the Cloud Messaging platform to mobile applications with a GeoBAT installation. Notifications were written manually, a target application was selected, and each notification was set to deploy immediately (Figure 4.11).



Notification	Status	Platform	Start / Send	End	Sends	Opens
Please complete nightly survey	Scheduled		Nov 14, 2018 6:49 PM	–	–	–
Please complete lunchtime survey	✓ Completed		Nov 14, 2018 12:11 PM	–	<1000	16%
Please complete survey	✓ Completed		Nov 14, 2018 7:23 AM	–	<1000	11%
Please complete nightly survey	✓ Completed		Nov 13, 2018 7:32 PM	–	<1000	21%
Please complete survey	✓ Completed		Nov 13, 2018 4:24 PM	–	<1000	16%
Please complete survey	✓ Completed		Nov 13, 2018 11:20 AM	–	<1000	21%
Please complete survey	✓ Completed		Nov 13, 2018 8:34 AM	–	<1000	17%
Please complete survey	✓ Completed		Nov 12, 2018 6:54 PM	–	<1000	0%
Please complete survey	✓ Completed		Nov 12, 2018 4:13 PM	–	<1000	0%
Please complete survey	✓ Completed		Nov 12, 2018 11:14 AM	–	<1000	0%

Figure 4.11 Cloud Messaging panel in Firebase Console

4.6 Survey Activity and Intents

The implementation of dependencies for Google Maps and Firebase API's provided the functionality to access services that were fundamental to operate an EMA application, however one activity brought these services together, and that was the Survey Activity. This activity was developed to connect user input with the database and deliver the input by clicking the 'Enter Results" button.

In the Map Activity the collection of participant data started through the declaration of an intent object in the setInfoWindowClickListener. An information window positioned above a location marker took location and temporal information from an instance of the LatLng and SimpleDateFormat classes, and once this window was clicked, passed the location data to the Survey Activity. Location data was passed to the Survey Activity by using the putExtra method defined in the Intent class. Within the Survey Activity a position instance variable was declared to reference the putExtra and getParceable methods, and the position value was called from the Database Reference object named myRootRef to record latitude and longitude values in the database.

Developing an interface to enable the user to input data consisted of the implementation of three widgets, namely twelve text views, twenty checkboxes, and one button. Text views were displayed as questions and indicators of safety, checkboxes were displayed on a level of one to five for each question, and the 'Enter Results' button was placed at the bottom to ensure users read all questions prior to committing their answers.

This EMA application required users to choose a value from each checkbox in relation to each of the four questions asked. For each checkbox to function a Boolean variable was called within the onCheckboxClicked method. At this location a switch statement was developed to confirm if a checkbox was checked, and an instance variable from an Array List was created to reference which checkbox was chosen. Once all values were entered, a reference to the database was declared through the myRootRef object to link the questions to the checkbox selections within the onResultPressed method. This method contained all the functionality behind the 'Enter Results' button (Figure 4.12).

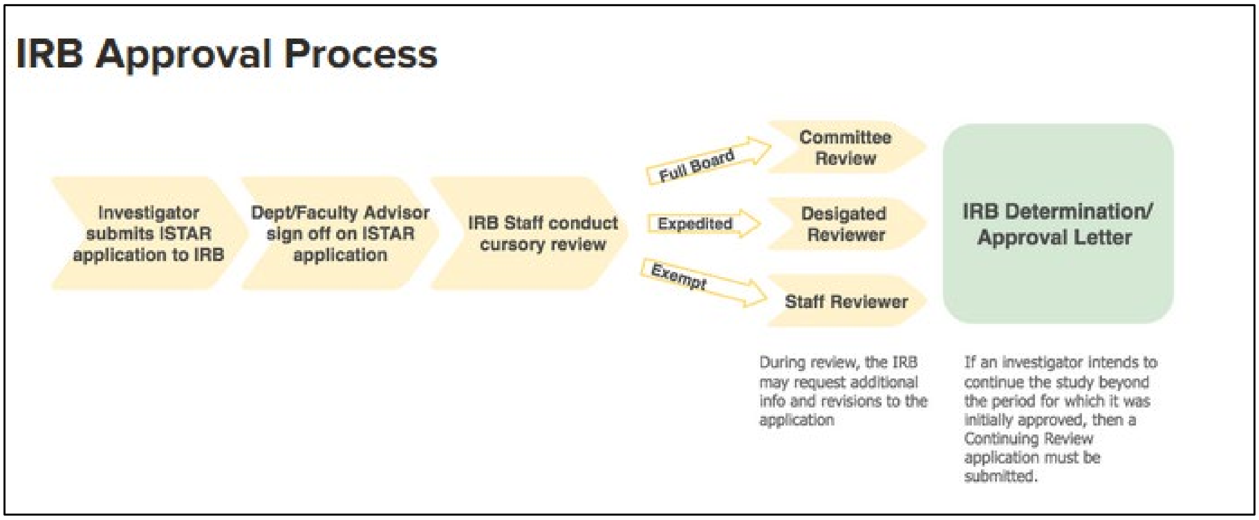
```
public void onResultPressed(View view){

    Intent intent = this.getIntent();

    DatabaseReference myRootRef = mLatLongDatabaseReference.push();
    LatLng position = intent.getExtras().getParcelable( key: "position");

    myRootRef.setValue(position);
    myRootRef.child("time").setValue(time);
    myRootRef.child(question1).setValue(selection1);
    myRootRef.child(question2).setValue(selection2);
    myRootRef.child(question3).setValue(selection3);
    myRootRef.child(question4).setValue(selection4);

    new AlertDialog.Builder( context: this).setIcon(R.drawable.notify_512x512)
            .setTitle("Thank you for completing the survey!")
            .setMessage("You can close the app, or:")
            .setPositiveButton( text: "Go To Home Page to Start Again", (dialog, which) → {
                    Intent intent = new Intent( packageContext: SurveyActivity.this, MainActivity.class);
                    startActivity(intent);

            })
            .show();

}
```

Figure 4.12 Intent, Realtime Database, and Alert Dialog configuration in Survey Activity

4.7 Institutional Review Board (IRB) Review

Prior to the release of this EMA application an IRB review through the University of Southern California (USC) was required to ensure human subjects received the proper protections while collecting and disseminating data. The IRB review was coordinated to ensure this project complied with institutional policies and state, local, and federal laws. An IRB application and Human Subjects Protection Training (CITI) were required of all study personnel.

The application provided an extensive description of the study, and the submission was used as a benchmark to assist reviewers in their determination of whether the study qualified as human subjects research. One month from initial submission, and subsequent requests from the staff reviewer, it was determined that this study qualified for the USC Human Research Protection Program Flexibility Policy and was found to have no more than minimal risk. Therefore, the IRB application was approved, a waiver of signed informed consent was granted, and the recruitment of testers for this EMA application could commence.



During review, the IRB may request additional info and revisions to the application

If an investigator intends to continue the study beyond the period for which it was initially approved, then a Continuing Review application must be submitted.

Figure 4.13 IRB Approval Process; Data from Office for Protection of Research Subjects, accessed November 20, 2018, https://oprs.usc.edu/irb-review/istar/

Chapter 5 Results

The objective of this project was to develop a mobile application as a proof of concept that provided users with a platform to generate perception of safety data at specific locations and times of the day. The main requirement was to develop a platform that enabled users to navigate a map, enter a data point from their device or alternate location, and switch to a survey platform to answer questions based on their perceptions (Figure 5.1). This chapter presents the final components of the mobile application developed for this study and the evaluation process performed to verify the application functioned as intended.

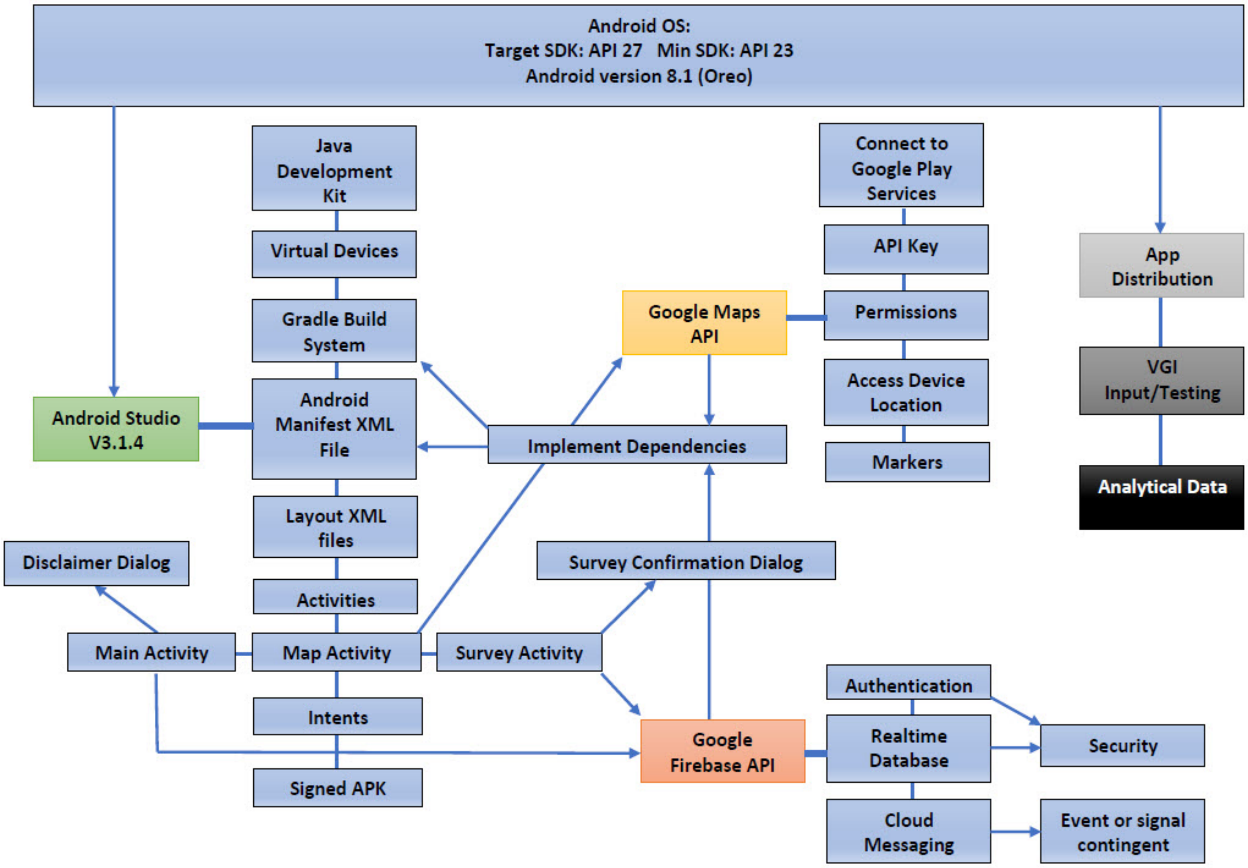


Figure 5.1 High level application functionality

5.1 Application Layout

The design of this application ensured that a diverse subset of users could understand the functionality of the application and be able to participate. Verifying that the application could be operated by novice and professional users ensured that perception of safety data would be sufficiently collected and disseminated to the Firebase Realtime database, parsed from the exiting JSON file, and integrated into a GIS for added analytical capability.

5.1.1 Main Activity

The application had three distinct activities that were presented to the user (Figure 5.2-5.5). The first activity was the authentication page which prompted the user to sign in with a valid e-mail address and sign up with a user name and password. Once the user was signed in, an alert dialog displayed a disclaimer that advised the users to use caution when collecting data. Users had the option to dismiss the dialog permanently by selecting ‘GOT IT’, or if they preferred to be reminded each time the application was opened, the user could select ‘KEEP DISCLAIMER’. Finally, the user had the option to select the ‘GO TO MAP / SURVEY’ button. This button led the user to the map activity. An option was also available for the user to sign out anytime from the main activity page.

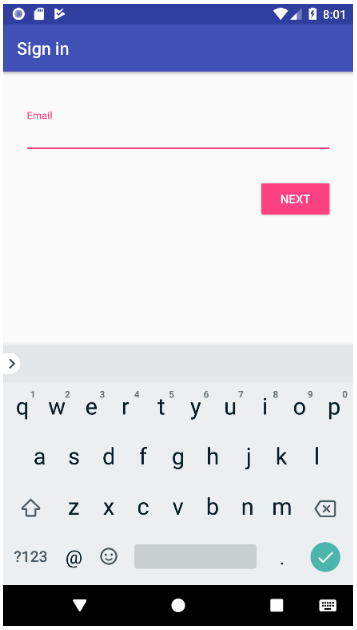
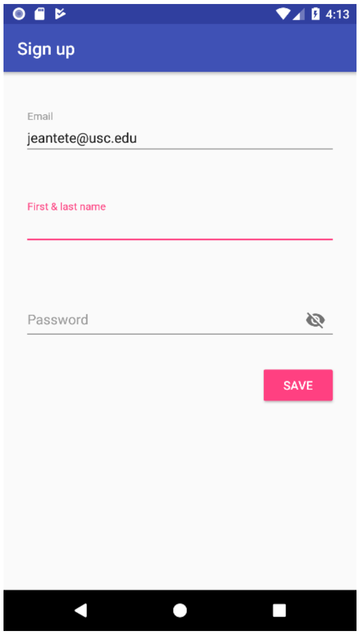
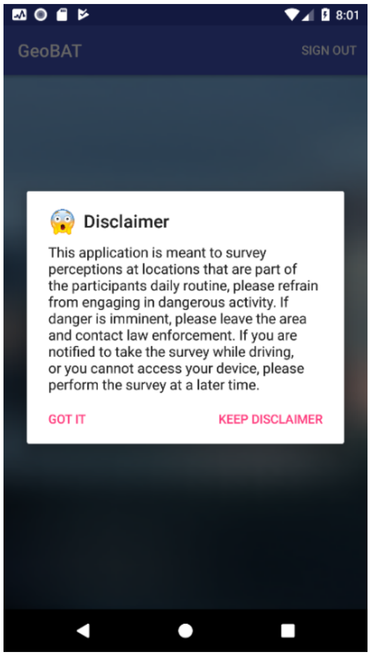


Figure 5.2. Sign in



Email: jeantete@usc.edu

Figure 5.3. Sign up



Disclaimer

This application is meant to survey perceptions at locations that are part of the participants daily routine, please refrain from engaging in dangerous activity. If danger is imminent, please leave the area and contact law enforcement. If you are notified to take the survey while driving, or you cannot access your device, please perform the survey at a later time.

Figure 5.4. Disclaimer

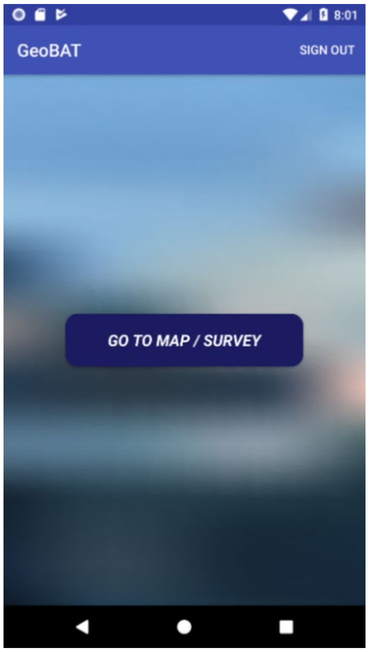
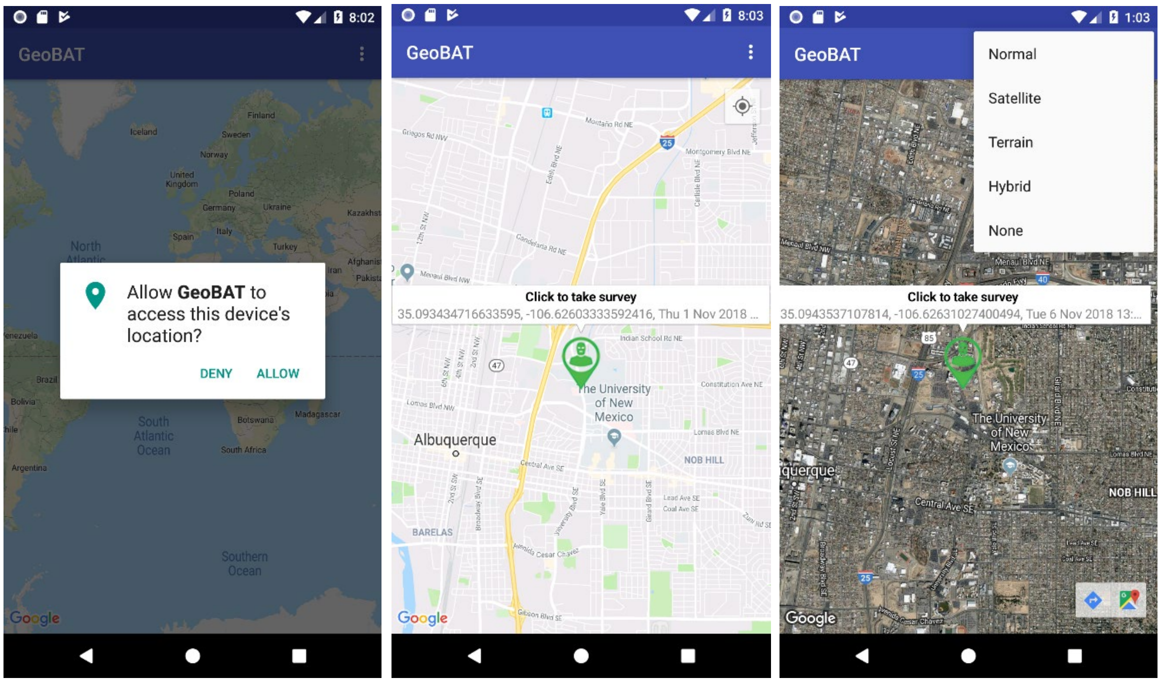


Figure 5.5. Button to the next activity

5.1.2 Map Activity

The second activity was the map activity, which integrated the Google Maps API to display multiple map configurations. Once the activity was engaged, the camera panned to the user's location, and a custom marker was displayed, along with a title bar that displayed the user location, time of day, and the option to click on the information window to transfer to the survey activity. During a session a user could 'long press' to create a new marker at a new location or press on the marker to drag to a preferred location. Once a location was determined, the user could click on the information window above the marker and go to the survey (Figure 5.6).



Allow **GeoBAT** to access this device's location?

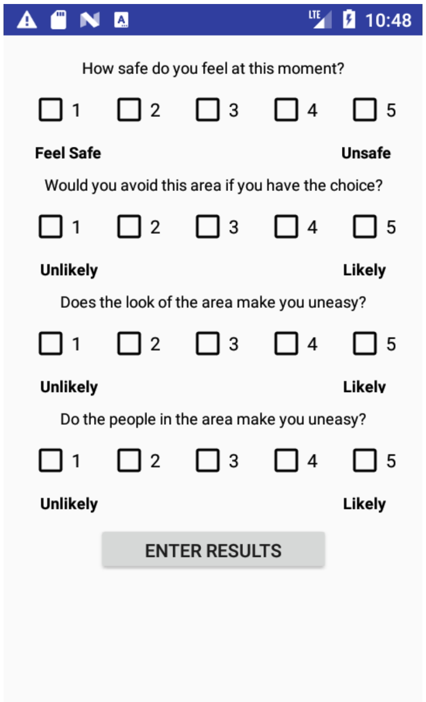
Click to take survey
35.093434716633595, -106.62603333592416, Thu 1 Nov 2018 ...

Click to take survey
35.0943537107814, -106.62631027400494, Tue 6 Nov 2018 13:...

Figure 5.6. Map Activity. *Left*, request permissions; *Middle,* marker on map; *Right,* option to change map views.

5.1.3 Survey Activity

The third activity was the survey activity, where the user was prompted to answer four questions on a one to five scale. The lower end of the spectrum represented positive perceptions of the user location, and the higher end of the spectrum represented negative perceptions of the user location. This activity provided the user with an opportunity to answer all four questions by checking the appropriate value and selecting the Enter Results button on the touch screen when finished. On the backend, when the user selected the Enter Results button, the latitude/longitude values, time stamp, and values chosen for each question were sent to the Firebase Realtime Database (refer to Figure 4.9). On the front end, at the conclusion of the survey, a dialog appeared confirming the user's selections were valid. Figure 5.7 and 5.8 demonstrate the configuration for the survey activity.



How safe do you feel at this moment?

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Feel Safe Unsafe

Would you avoid this area if you have the choice?

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Unlikely Likely

Does the look of the area make you uneasy?

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

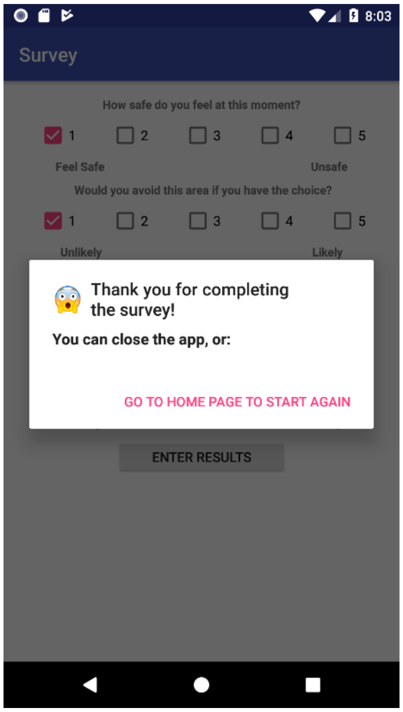
Unlikely Likely

Do the people in the area make you uneasy?

☐ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Unlikely Likely

Figure 5.7. Survey Activity



How safe do you feel at this moment?

☑ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Feel Safe Unsafe

Would you avoid this area if you have the choice?

☑ 1 ☐ 2 ☐ 3 ☐ 4 ☐ 5

Unlikely Likely

Thank you for completing the survey!

You can close the app, or:

Figure 5.8. Confirmation dialog

Once the survey was completed the user had the option to close the application or return to the home page to enter another location and take the survey again.

5.2 Application Release

After exhaustive testing of the application on multiple Android emulator configurations and a personal physical device the next step was to prepare the application for submission to the Google Play App Store for internal testing (Figure 5.9). Internal testing was ideal for testing this application due to the limited number of testers and the ability to control which users had access to the application.

The first step to releasing the application for testing was to specify a new project in the Google Play Console. Second, application restrictions were set to Android apps in the Google Cloud Platform, and usage of the Google Maps SDK for this application was restricted by adding the package name and SHA-1 certificate fingerprint to the platform. Third, the Build Variant in Android Studio was changed from debug to release mode, a password protected Keystore file was created, a signed APK file was generated, and the APK file was uploaded to the Google Play Console. Fourth, a store listing was developed in the Google Play Console to summarize application functionality and display graphic assets to the user. Finally, an internal rollout of this application was initiated, and the list of internal testers was added.

Throughout the testing process minor bugs were detected by the initial testers and subsequent versions were released before the final testing pool was deployed. Changes were made to the version code and version name in the app-level build.gradle file, and a new signed APK file was generated and uploaded to the Google Play Console. Figure 4.3 in Chapter Four displays the app-level build.gradle file with the final version code and name for this application.

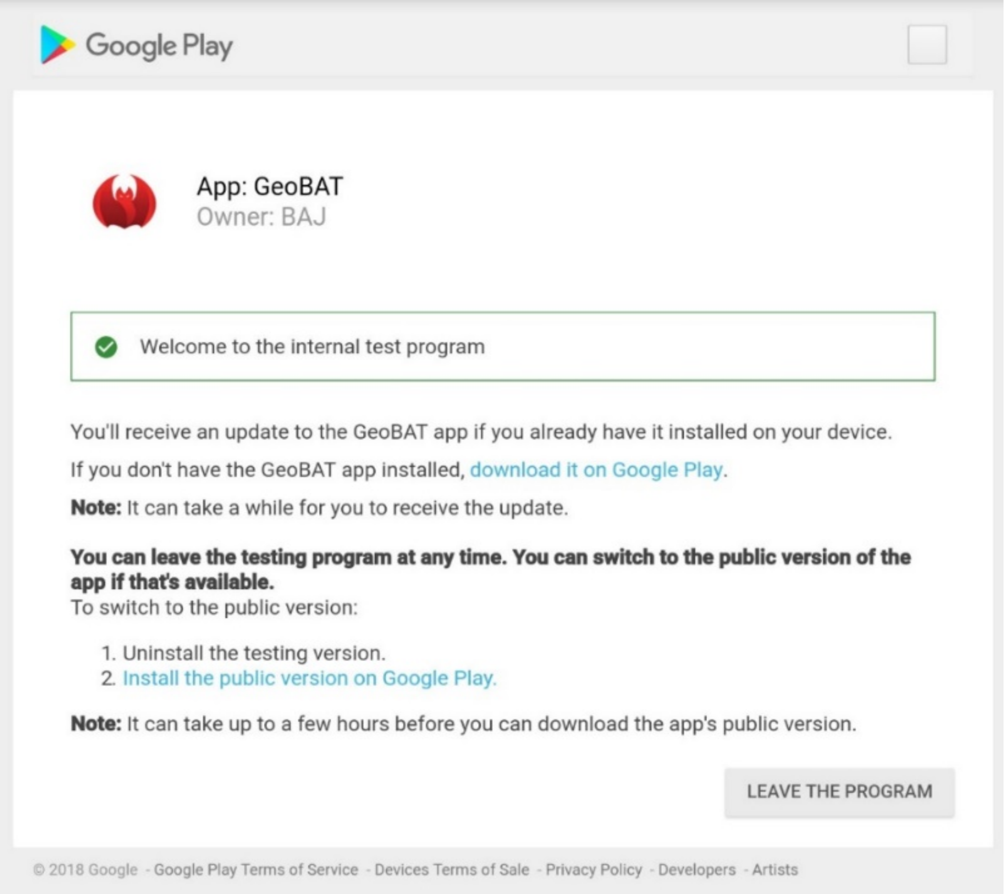
Figure 5.9 Process to deploy application for internal testing

5.3 Application Evaluation and Testing

Upon IRB approval, documents were generated to recruit users and to provide participants with an informed consent template for their review. These documents were distributed to faculty at the University of New Mexico and personal acquaintances to generate interest in testing the application for this study. The goal was to provide enough participants (N = 6) to test the functionality of the application and the viability of this EMA study. Feedback was encouraged at the completion of the EMA study and participants were directed to a primary e-mail address to provide this feedback.

5.3.1 Human Subjects

Recruitment to participate in application testing and the EMA study generated a total of six participants. This included students from the University of New Mexico and professionals from various fields. Once a tester consented to participate in this study they were added to the list of testers and provided with a link through their g-mail provider to install the application from the Google Play Store (Figure 5.10).



App: GeoBAT
Owner: BAJ

Welcome to the internal test program

You'll receive an update to the GeoBAT app if you already have it installed on your device.

If you don't have the GeoBAT app installed, download it on Google Play.

Note: It can take a while for you to receive the update.

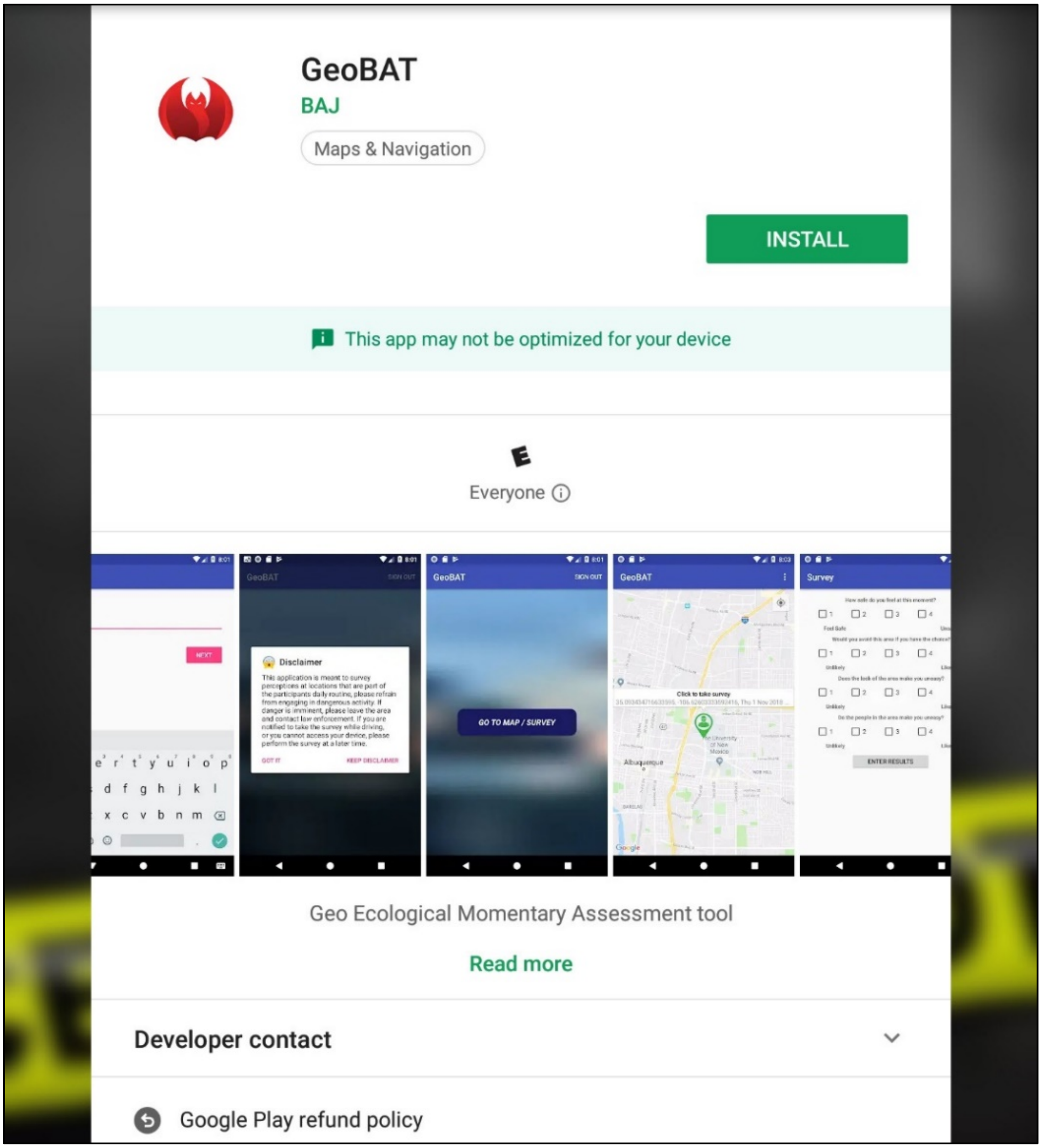
You can leave the testing program at any time. You can switch to the public version of the app if that's available.
To switch to the public version:

1. Uninstall the testing version.
2. Install the public version on Google Play.

Note: It can take up to a few hours before you can download the app's public version.

Figure 5.10 Invite to install GeoBAT mobile EMA application

The invite prompted the user to download the application from Google Play, and if the user consented, the store listing was made available to install the application. The store listing provided graphics of each activity, the purpose of the EMA study, and instructions guiding the user on how to properly operate the mobile application (Figure 5.11 and 5.12).



This app may not be optimized for your device

Disclaimer

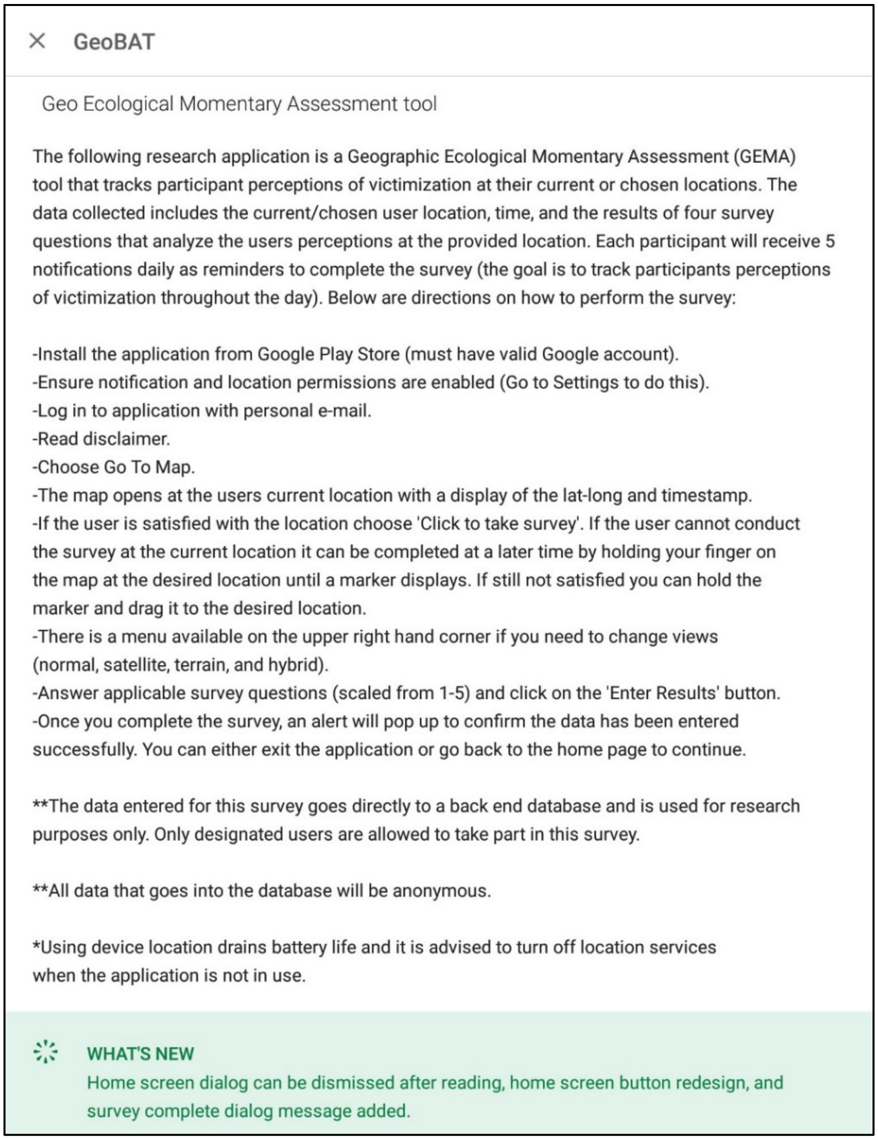
This application is meant to survey perceptions at locations that are part of the participants daily routine, please refrain from engaging in dangerous activity. If danger is imminent, please leave the area and contact low enforcement. If you are notified to take the survey while driving, or you cannot access your device, please perform the survey at a later time.

GOT IT   KEEP DISCLAIMER

Geo Ecological Momentary Assessment tool

Google Play refund policy

Figure 5.11 Google Play Store install page



Geo Ecological Momentary Assessment tool

The following research application is a Geographic Ecological Momentary Assessment (GEMA) tool that tracks participant perceptions of victimization at their current or chosen locations. The data collected includes the current/chosen user location, time, and the results of four survey questions that analyze the users perceptions at the provided location. Each participant will receive 5 notifications daily as reminders to complete the survey (the goal is to track participants perceptions of victimization throughout the day). Below are directions on how to perform the survey:

-Install the application from Google Play Store (must have valid Google account).
-Ensure notification and location permissions are enabled (Go to Settings to do this).
-Log in to application with personal e-mail.
-Read disclaimer.
-Choose Go To Map.
-The map opens at the users current location with a display of the lat-long and timestamp.
-If the user is satisfied with the location choose 'Click to take survey'. If the user cannot conduct the survey at the current location it can be completed at a later time by holding your finger on the map at the desired location until a marker displays. If still not satisfied you can hold the marker and drag it to the desired location.
-There is a menu available on the upper right hand corner if you need to change views (normal, satellite, terrain, and hybrid).
-Answer applicable survey questions (scaled from 1-5) and click on the 'Enter Results' button.
-Once you complete the survey, an alert will pop up to confirm the data has been entered successfully. You can either exit the application or go back to the home page to continue.

**The data entered for this survey goes directly to a back end database and is used for research purposes only. Only designated users are allowed to take part in this survey.

**All data that goes into the database will be anonymous.

*Using device location drains battery life and it is advised to turn off location services when the application is not in use.

WHAT'S NEW
Home screen dialog can be dismissed after reading, home screen button redesign, and survey complete dialog message added.

Figure 5.12 Google Play description of study and instructions on mobile application operation

5.3.2 Testing time frame

Application testing was performed in three phases between October 11, 2018 and November 16, 2018. The initial phase involved two users who tested the basic functionality of the application on multiple Android devices. Starting on November 5, 2018 additional testers were added to the roster. The median phase consisted of data collection via the application and the implementation of data extraction processes to import data into a GIS. The final testing phase was implemented between November 12 and November 16, 2018. This phase included a total of six testers who were sent three to five notifications at random times, reminded to select their device or random location at a specific time, and to complete the EMA survey that tied into this chosen location.

5.3.3 User Metrics

User metrics were analyzed through Firebase Analytics to determine the extent of user interaction and responsiveness to the notifications sent. Utilizing these analytics helped to determine the extent of user interest throughout the study and determine if the amount of notifications pushed to the users was enough. In order to scientifically investigate participant perceptions of safety all variables must be accurately measured including the amount of responses to the sent notifications.

First, each notification received was tagged as an event in Firebase Analytics. In this case a total of 92 events were received by all participants, which was an average of 15 per user throughout the final phase of testing (Figure 5.13).

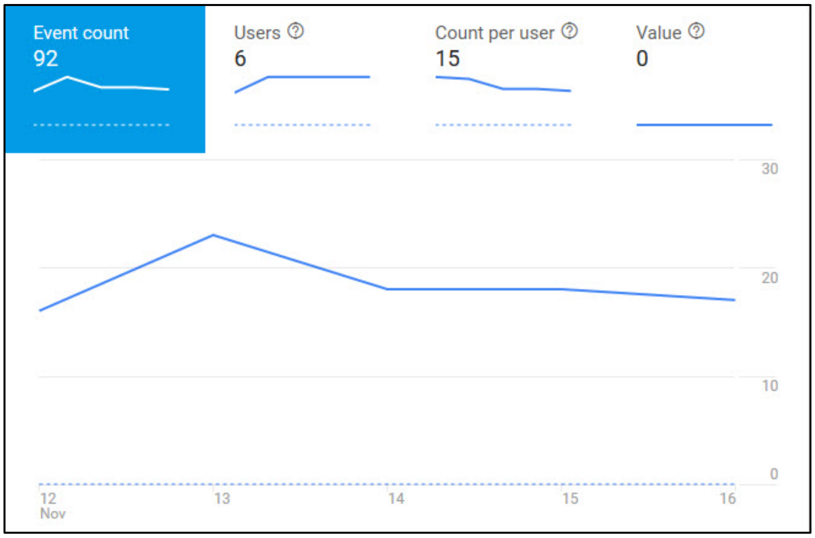


Figure 5.13 Notifications received by users throughout final phase of testing

Second, and most important, were the number of notifications opened (Figure 5.14). The number of opened notifications was an opportunistic metric to gauge if the user was responsive to performing the survey at the provided time.

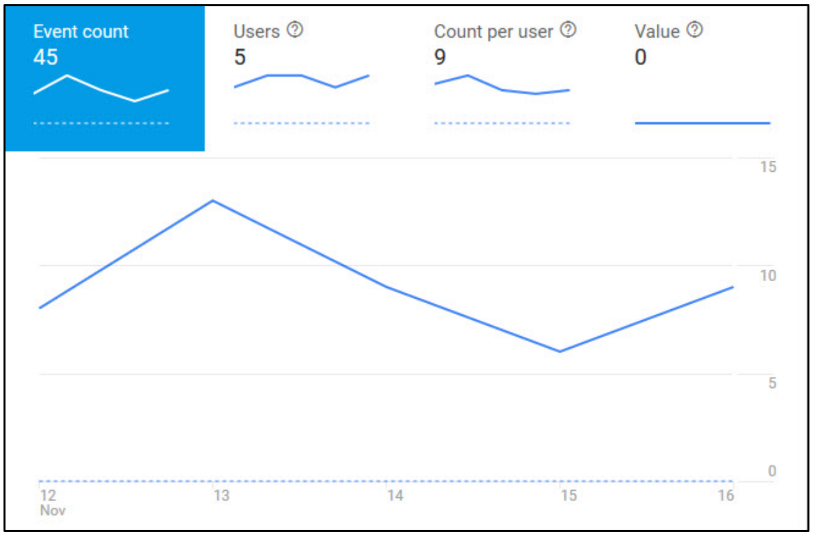


Figure 5.14 Notifications opened by users throughout final phase of testing

A total of 45 notification events were opened by all users and averaged 9 opens per user. The total number of notifications opened was 48.91% when compared to the number of notifications received. Since one user was unresponsive to notifications this skewed the number per user to 60% response rate when compared to the number of notifications received. Ideally, the goal was a response rate > 90% for a small group of testers, however the data was still enough to provide proof of mobile application functionality and the ability to collect data for future GIS analysis.

Event contingent notifications were not recorded during the testing process. All participants used the signal contingent option to generate data; however, the process to measure event contingent data can be applied by assessing the number of recorded replies in the FCM and

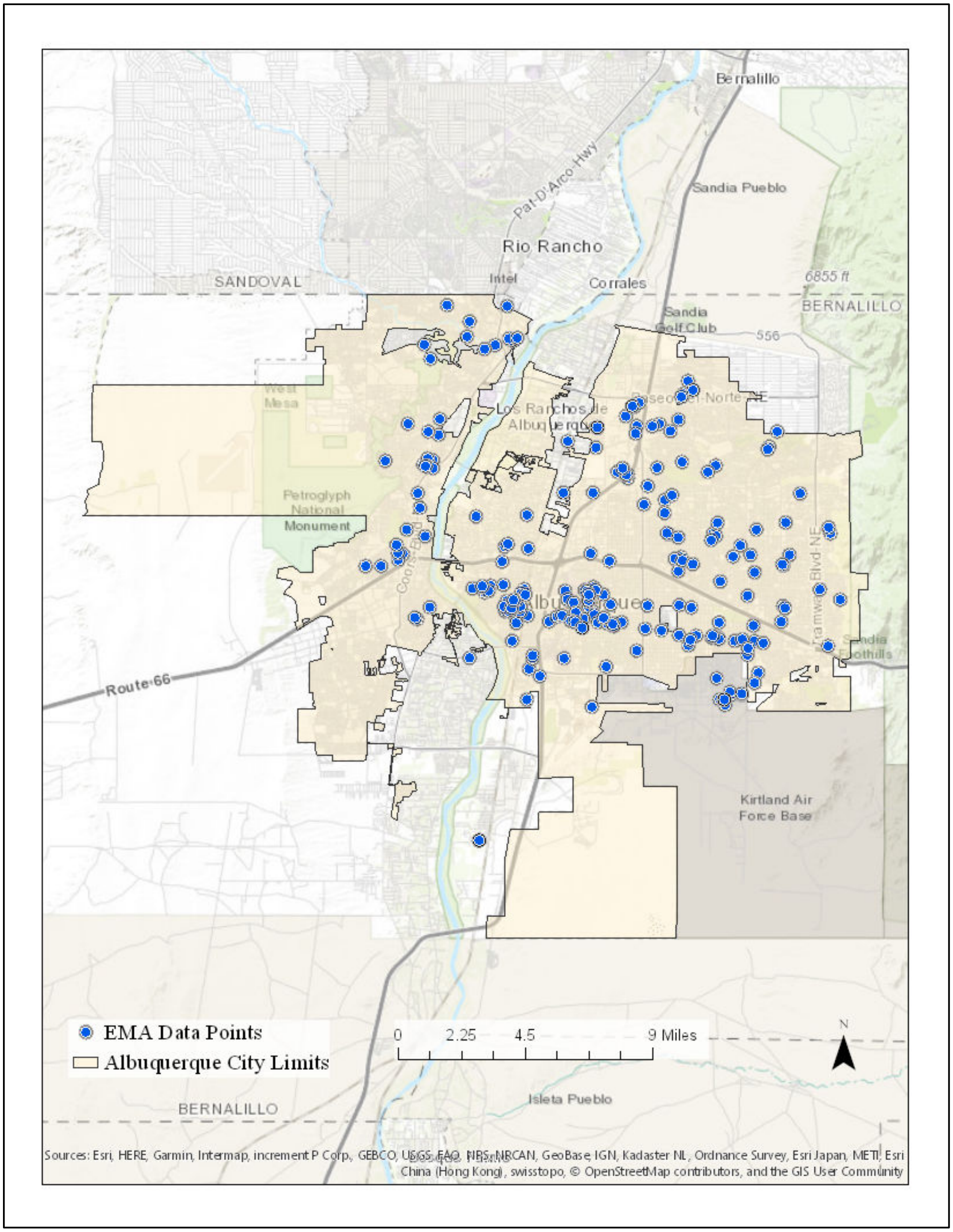
Realtime database and using Firebase to query those time stamped values that fall out of signal contingent time frame.

5.3.4 Evaluation of the application

Feedback of application functionality was an option provided to users during recruitment and throughout the testing phase. During the initial testing phase comments about the application functionality were positive, however users provided input on possible modifications to the UI, specifically the adaptability of the Survey Activity to different screen configurations. One feature of importance was the lack of a scroll feature, as one user could not scroll through the survey questions on a device with a smaller screen size. In addition, the disclaimer on the home screen could not be dismissed during the initial phase, and there was no conformation screen in the Survey Activity to alert the user that they successfully completed the survey. Subsequent updates remedied these issues and were applied to existing and future installs. During the final testing phase users found a small deficiency in the Survey Activity that allowed them to check more than one check box per question and pass these values to the database. It was suggested by users to use radio buttons to limit the choice to one value per question. This was something that could be changed in the Survey Activity class and redistributed for later release. After the final testing phase all participants did not experience additional issues. Most of the feedback focused on the concept of this study and on the viability of performing EMA's on a mobile application to collect real time data, which was a new concept to the testers.

5.4 Data Collection and Evaluation

EMA data collected from participant mobile devices distributed throughout the city of Albuquerque netted a total of 253 data points (Figure 5.15). These data points were collected over three testing phases to ensure the functionality of the application met the requirements of this EMA study.

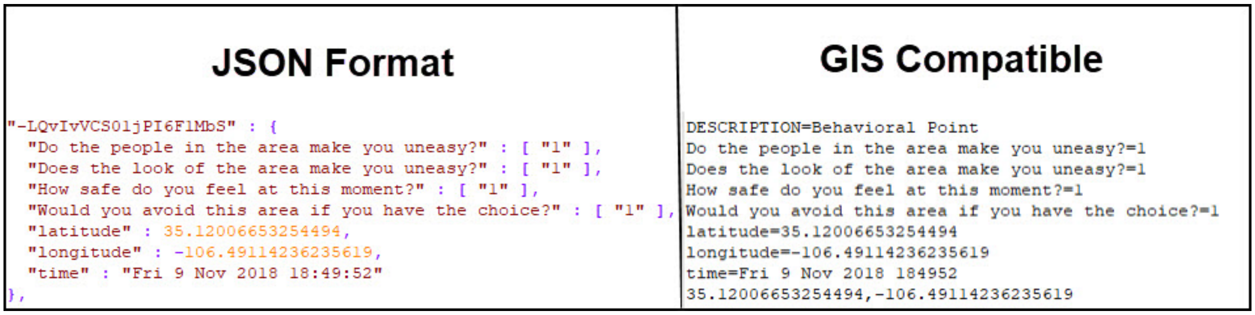


Sources: Esri, HERE, Garmin, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, IGN, Kadaster NL, Ordnance Survey, Esri Japan, METI, Esri China (Hong Kong), swisstopo, © OpenStreetMap contributors, and the GIS User Community

Figure 5.15 EMA data points collected from study participants

5.4.1 Parsing the data

At the conclusion of all testing phases participant data was exported from the Firebase Realtime database in the form of a JSON file. As mentioned in prior sections, data was stored in the database as JSON objects and added to the JSON tree as a value once the Enter Results button was pressed from the Survey Activity. All data from the JSON tree was exported from the database to a JSON file and parsed in Notepad++ to a format that was compatible for fast import into Global Mapper (Figure 5.16).

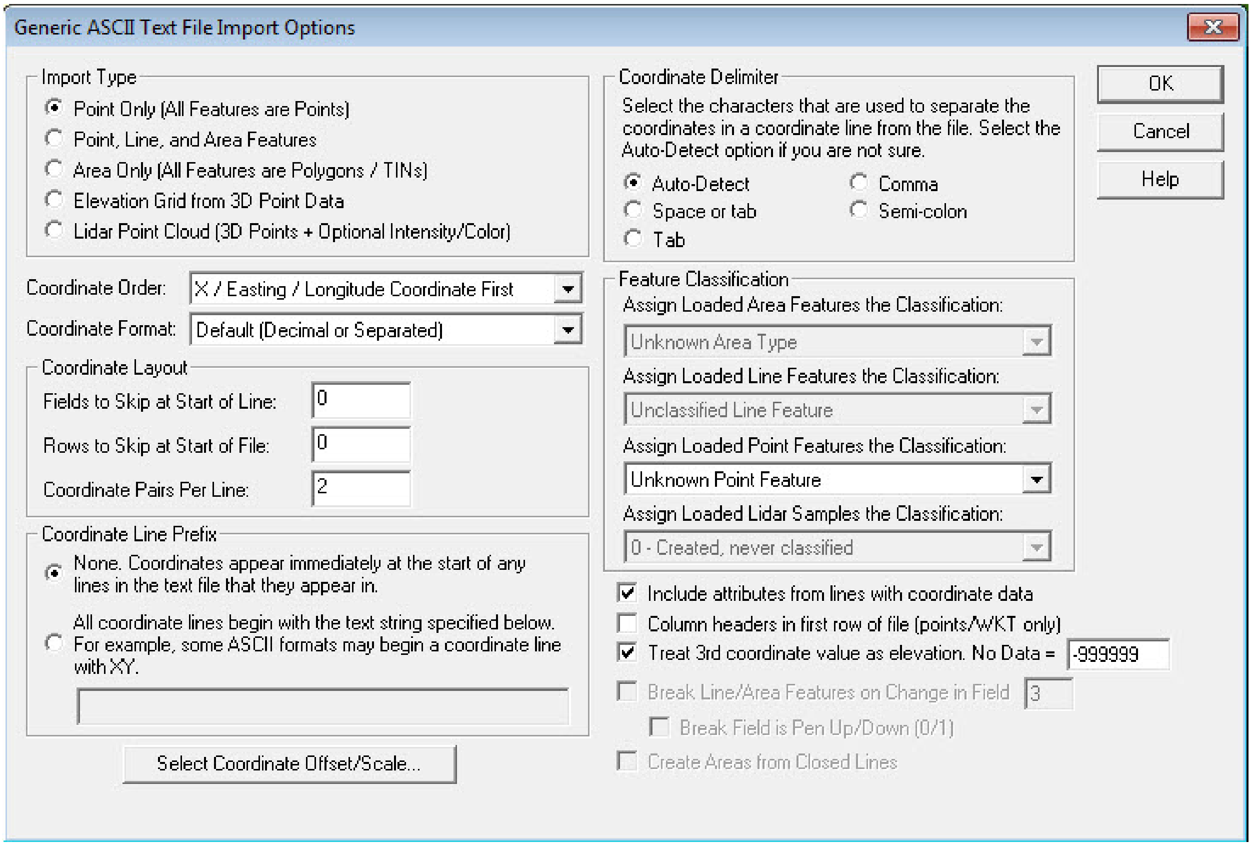


```
"-LQvIvVCS01jPI6F1MbS" : {
  "Do the people in the area make you uneasy?" : [ "1" ],
  "Does the look of the area make you uneasy?" : [ "1" ],
  "How safe do you feel at this moment?" : [ "1" ],
  "Would you avoid this area if you have the choice?" : [ "1" ],
  "latitude" : 35.12006653254494,
  "longitude" : -106.49114236235619,
  "time" : "Fri 9 Nov 2018 18:49:52"
},
```

```
DESCRIPTION=Behavioral Point
Do the people in the area make you uneasy?=1
Does the look of the area make you uneasy?=1
How safe do you feel at this moment?=1
Would you avoid this area if you have the choice?=1
latitude=35.12006653254494
longitude=-106.49114236235619
time=Fri 9 Nov 2018 184952
35.12006653254494,-106.49114236235619
```

Figure 5.16 Conversion from JSON to GIS formatted text file

Data was parsed through the following methods: (1) replaced unnecessary punctuation with blank space; (2) bookmarked the user id and removed bookmarks by line; (3) added the description by marking a blank line with a regular expression in the Find what parameter and replaced with the desired value; (4) replaced spaces between equal signs with no spaces; and (5) applied line spaces between all values. The process was applied to all 253 values at once and took approximately ten minutes to complete. The final text file was imported into Global Mapper for review. Global Mapper contains options to import generic text files once they are formatted correctly (Figure 5.17).



Generic ASCII Text File Import Options

Import Type
● Point Only (All Features are Points)
○ Point, Line, and Area Features
○ Area Only (All Features are Polygons / TINs)
○ Elevation Grid from 3D Point Data
○ Lidar Point Cloud (3D Points + Optional Intensity/Color)

Coordinate Order: X / Easting / Longitude Coordinate First
Coordinate Format: Default (Decimal or Separated)

Coordinate Layout
Fields to Skip at Start of Line: 0
Rows to Skip at Start of File: 0
Coordinate Pairs Per Line: 2

Coordinate Line Prefix
● None. Coordinates appear immediately at the start of any lines in the text file that they appear in.
○ All coordinate lines begin with the text string specified below. For example, some ASCII formats may begin a coordinate line with XY.

Select Coordinate Offset/Scale...

Coordinate Delimiter
Select the characters that are used to separate the coordinates in a coordinate line from the file. Select the Auto-Detect option if you are not sure.
● Auto-Detect ○ Comma
○ Space or tab ○ Semi-colon
○ Tab

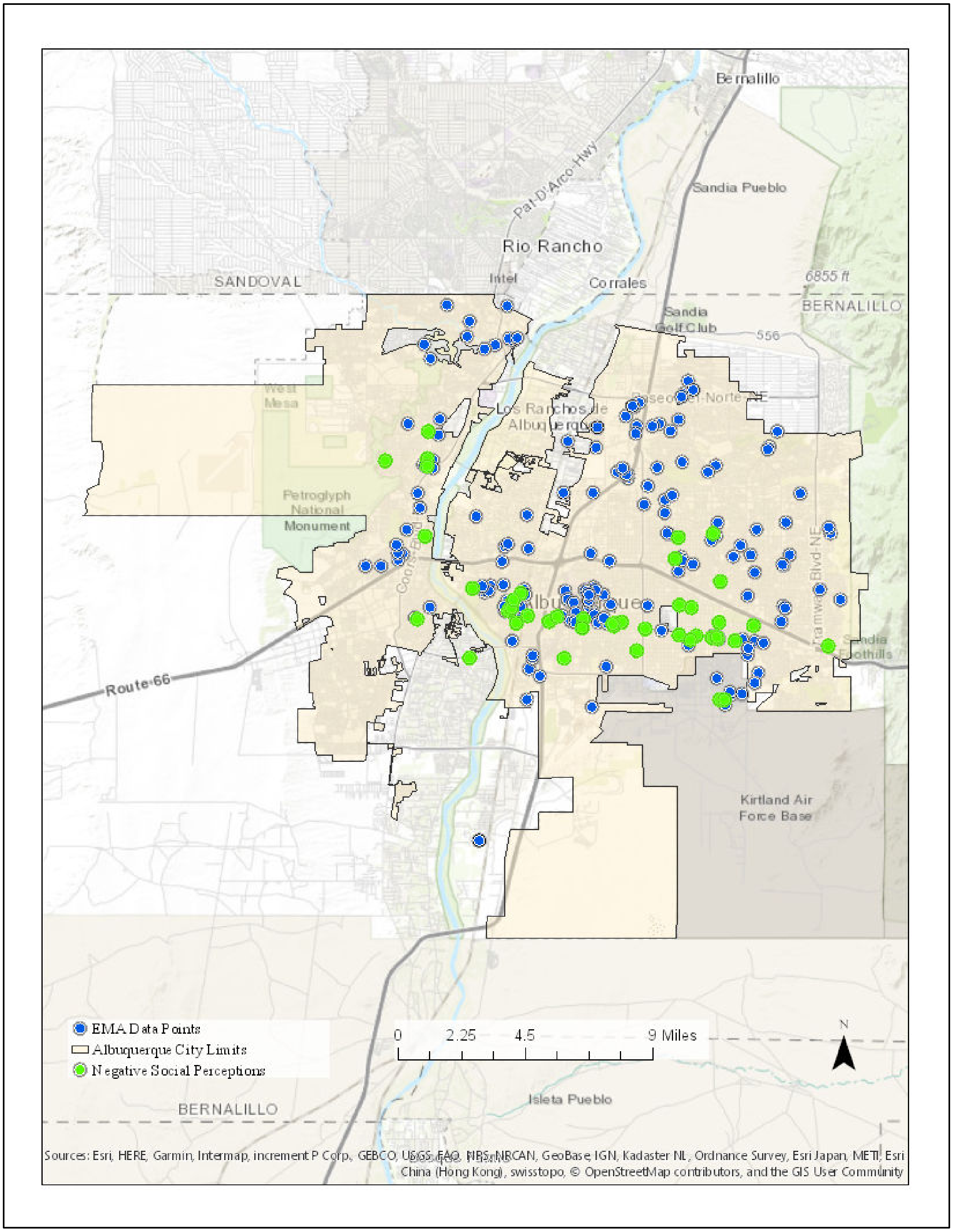
Feature Classification
Assign Loaded Area Features the Classification: Unknown Area Type
Assign Loaded Line Features the Classification: Unclassified Line Feature
Assign Loaded Point Features the Classification: Unknown Point Feature
Assign Loaded Lidar Samples the Classification: 0 - Created, never classified

☑ Include attributes from lines with coordinate data
☐ Column headers in first row of file (points/WKT only)
☑ Treat 3rd coordinate value as elevation. No Data = -999999
☐ Break Line/Area Features on Change in Field 3
☐ Break Field is Pen Up/Down (0/1)
☐ Create Areas from Closed Lines

Figure 5.17 Global Mapper generic text file import.

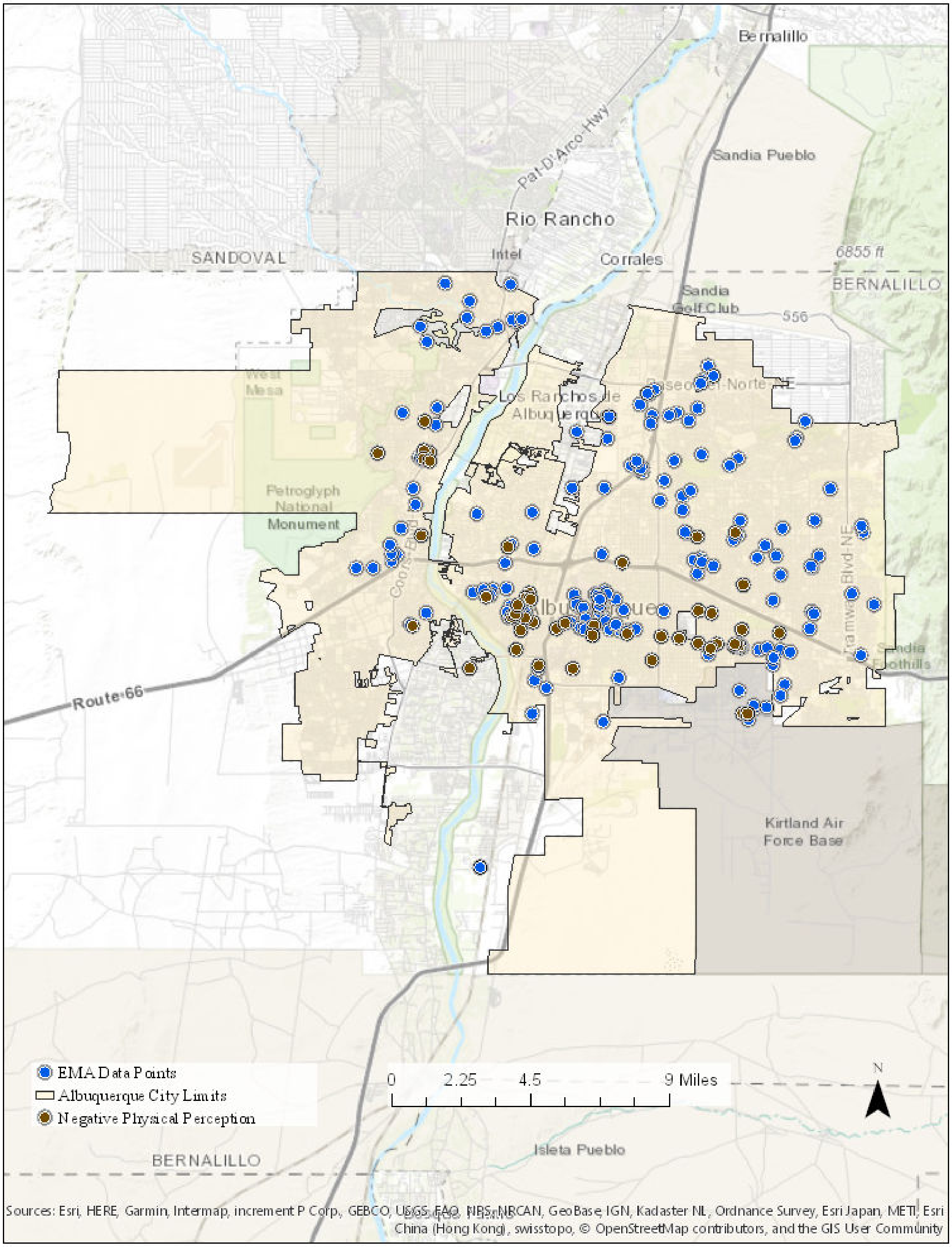
5.4.2 Applying Feature Data

Importing the text file through Global Mapper expedited the process to create point features with associated perception data that could be used for subsequent geospatial analysis. Since latitude and longitude was used, the points were generated with a geographic projection. The final step was to export the points as a shapefile for import into multiple geospatial platforms including ArcGIS Pro. Integrating this data into a GIS allows the user to visualize areas and times where participants avoid certain areas, and where they may respond with a negative view of these areas due to apparent physical and social disorders (Figures 5.18 – 5.20).



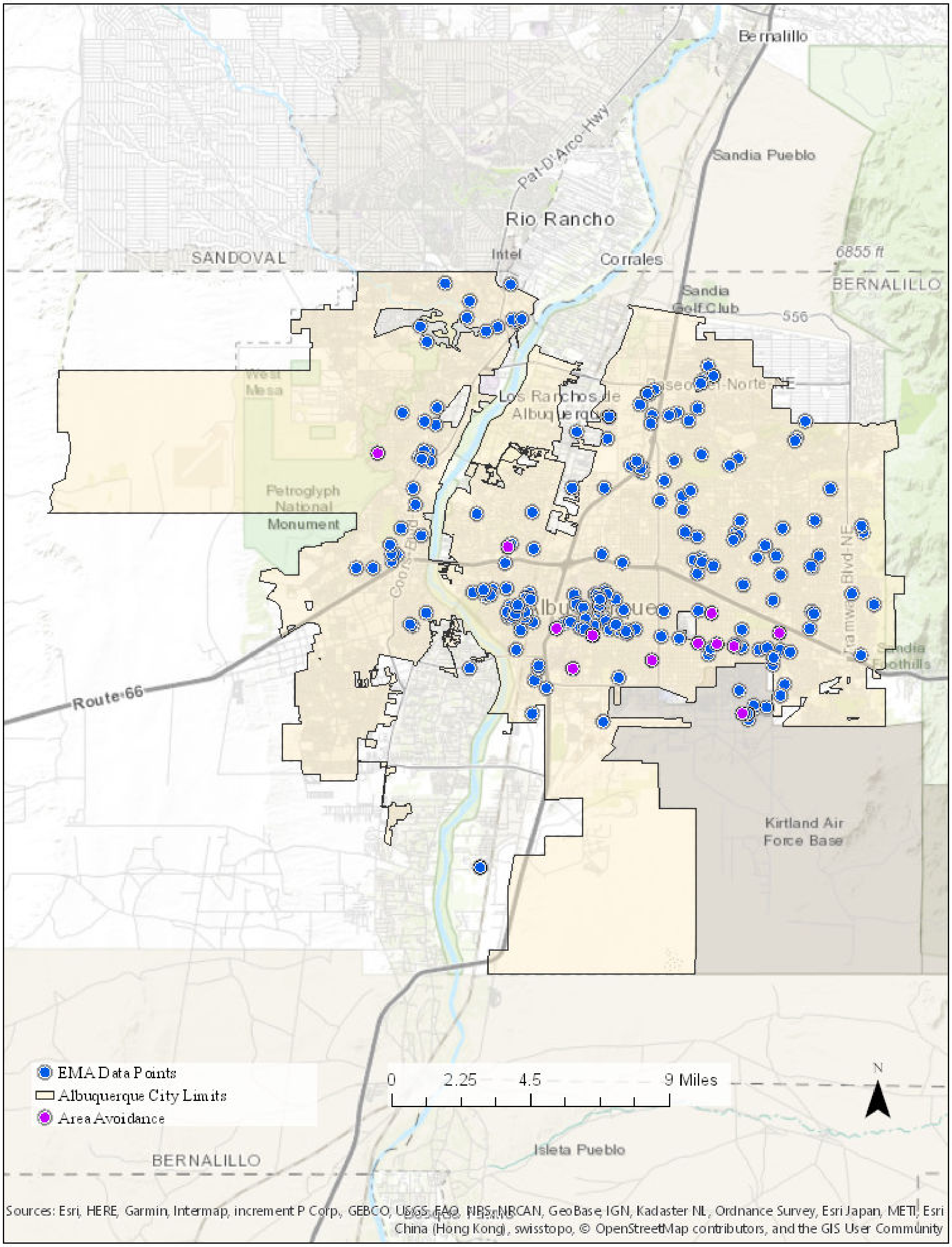
Sources: Esri, HERE, Garmin, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, IGN, Kadaster NL, Ordnance Survey, Esri Japan, METI, Esri China (Hong Kong), swisstopo, © OpenStreetMap contributors, and the GIS User Community

Figure 5.18 Areas where participants perceived social disorder



Sources: Esri, HERE, Garmin, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, IGN, Kadaster NL, Ordnance Survey, Esri Japan, METI, Esri China (Hong Kong), swisstopo, © OpenStreetMap contributors, and the GIS User Community

Figure 5.19 Areas where participants perceived physical disorder



Sources: Esri, HERE, Garmin, Intermap, increment P Corp., GEBCO, USGS, FAO, NPS, NRCAN, GeoBase, IGN, Kadaster NL, Ordnance Survey, Esri Japan, METI, Esri China (Hong Kong), swisstopo, © OpenStreetMap contributors, and the GIS User Community

Figure 5.20 Areas participants may avoid in the future

Chapter 6 Conclusions

The development and deployment for testing of the GeoBAT EMA mobile application provided a spatiotemporal element to personal perspective of safety data collection and met the project objectives required for mobile application functionality, data collection, and distribution. This chapter reflects on the objectives that were met, the challenges that were encountered, and the limitations of the EMA mobile application. Most importantly, this chapter reflects on the mobile application's future viability in tracking the fear of crime through dynamic EMA methods and the possible advantages of integrating the data collected into a GIS for comparison with real time or historical crime data.

6.1 Final EMA Mobile Application

User requirements were defined in Chapter 3 to determine the type of parameters required for users to successfully operate a mobile application that collected spatiotemporal data through event contingent and signal contingent protocols. All the basic requirements outlined to collect spatiotemporal and perception data from a mobile application platform were met. In addition, the requirements to store the collected data, and the development of a process to distribute to a GIS platform was achieved.

As mentioned earlier, the combination of the Android, Google Maps and Firebase API's opened the door for the development of a mobile platform that enabled users to authenticate their identity, receive notifications, browse through a map, select a location, answer four perception questions, and distribute this data to servers.

6.2 Development Challenges

Due to the extensive documentation that is available to Android developers, for Google Maps integration, and Google Firebase, there were minimal issues linking these platforms together. Although, linking these platforms together and deploying an existing application still took a great amount of time, due to the research involved to execute the required functionality, and overcoming the challenge of understanding code in documentation that was deprecated.

6.3 Project Limitations

Developing this mobile application to perform a comprehensive EMA on perceptions of safety fulfilled the base line requirements set forth for this project. However, a great number of limitations exist regarding mobile application platforms used, application features required to perform a robust survey, number of participants, data transfer, and API usage limits.

6.3.1 OS distribution limitations

As mentioned in Chapter 2, in the first quarter of 2018, there were approximately six million mobile applications distributed between Android and iOS platforms. Even though Android has a larger share, it is a limitation to distribute mobile applications developed specifically for crowdsourced perception studies to one platform.

The development of an EMA mobile application for this project on an Android platform limited access to testers, as most of the possible recruits for this study owned iPhones. Distributing to multiple platforms would have opened the door to additional testers, data points, and responsiveness to the configuration of this application.

6.3.2 Limitations in design

The design of most software, web, or mobile based applications go through multiple design and testing phases throughout their lifecycle to ensure the user experience and underlying functionality is optimal. In the case of this study, the time frame was limited, and testing was limited to iterative developer testing and one cycle of internal testing by a selected group of testers. Designing an EMA mobile application that is interactive and engages the user is important due to the requirements of the users to consistently interact and submit data. Once again, the application created for this project met the minimum requirements of functionality, but with further testing and development, additional features can be added to the mobile application to provide a better user experience, and a wider distribution of data that is relevant to an EMA based perception of safety study.

6.3.3 Data transfer limitations

A process was designed in this project for the developer to parse JSON data from Firebase Realtime database and export this data into a format that can be read into multiple GIS platforms. Even though the process was successful, it was limited due to its dependence on manual data entry on the developer side. The goal of future development work is to provide read functionality from the Firebase Realtime database to input real time data into a web or user created map once a user engages the Enter Results button in the Survey Activity. In other words, the main goal is to automatically place a point feature with associated attribute data on the map in conjunction with entries in the Firebase Realtime database.

6.3.4 Manual Notifications

Throughout the final testing phase, the process of sending notifications at time intervals set by the developer was enough to ensure all users received reminders to take the survey.

Firebase Cloud Messaging allowed the developer to send out notifications manually or set a specific time to send them out. Even though the process of sending out notifications was enough in this project, there were limitations that could be addressed in case of wider distribution. The main limitation was the manual distribution of notifications from the Firebase console. Each notification was sent at specific times from the Firebase console, which ties up developer resources. In the case of wider distribution, the process to overcome this limitation involves the integration of methods into the Main Activity to automate the times that notifications are distributed. Random generators or set times could be called to ensure participants receive the right amount of notifications and developers are not manually babysitting the console.

As mentioned in Chapter 5, all responses during the testing phase were based on signal contingent notifications and none were event contingent. Increasing the sample size would introduce situations where participants have the option to perform the survey at any time and provide data that reflects an individual's perceptions outside of the notification time frame.

6.3.5 API Usage Limits

The integration of Google Maps and Firebase API's was integral to the functionality of this mobile application and the viability of this EMA study. There were multiple API's and BaaS platforms to choose from including ArcGIS Runtime SDK for Android, Parse server, and Amazon Web Services (AWS) to name a few. However, in the case of this project, all Google platforms were chosen due to their flexible pricing structure and streamlined development process. As with most providers there were limitations to the amount of data usage for each platform. Due to the limited number of testers for this mobile application, usage fees were negligible, but the possibility of scaling to many users would change the pricing structure and plans.

6.4 Application Sharing

Although the application for this project was developed for internal testing within Albuquerque city limits, it is possible to distribute and use throughout the United States. The camera always drives to the device location, so data can be collected immediately from the area of interest. If the user gets lost, a Find My Location button is available to reorient the user. Wide distribution can be achieved by publishing the application in the Google Play Store from the internal testing track to the production track. Prior to production release additional beta testing and updates are required to ensure the user experience and functionality meet additional EMA study requirements. Sharing the application in the production process is not an arbitrary step, as extensive planning and resources are required to ensure the deployment of the application meets all release requirements. In addition, API usage limits would have to be assessed to provide the bandwidth to scale up to national level usage. The next section will cover the plans for future deployment and the resources required to accentuate this application to meet the challenges in collecting crowdsourced data for perception of safety research.

6.5 Future Work

This study introduced the development of an EMA mobile application that provided the functionality for users to record their perceptions of safety at their chosen location in real time. The collection of dynamic data and its integration into a GIS platform provided an opportunity to analyze user perceptions of safety within their surrounding physical and social environments. The development of the EMA mobile application was the first step, however there are additional steps required to fully recognize the potential of this application and its contribution to subsequent GIS analysis in tracking participant perceptions of safety.

6.5.1 Integration with iOS

As mentioned earlier in this chapter, development in a single OS limited the capability to recruit additional testers. Adding the GeoBAT application to the iOS platform would generate an additional pool of testers and allow more participants to engage once the application goes to production release. One of the main solutions to this issue is the integration of the Google Flutter SDK into the development environment. Flutter is Google's mobile application SDK that is built on the Dart programming language. Dart provides the ability to customize, build, and publish for both iOS and Android. Flutter is free to use, can be integrated into the Android Studio environment, is a faster way to build mobile applications, and can reach more users with a single investment (Flutter, *"FAQ"*, 2018).

The release of this application to a wider audience would result in a larger data sample to compare a user's perception of safety to actual crime data. Through a production release, recruitment could be expanded to a wider demographic base and population segments that commute extensively, such as Uber and UPS drivers. Expanding the sample would provide greater insight on the user's fear of crime and its impact on the study area.

6.5.2 Expansion of mobile application functionality

A main component missing in the final release for internal testing was an option for the user to enter demographic data immediately after authentication. The integration of demographic information into collected data sets provides an informational link that could be used in a GIS to compare with other data sets. The idea behind this option is to provide the user with a template to enter information regarding age range, gender, economic background, and race. This data would be attached to the generic user ID and applied to their survey responses to gain a broader perspective of user perceptions. For example, comparisons could be made between participants who report at different times or provide different reactions within the same area.

Survey responses were limited to four questions to focus on a user's perception of safety at the time of the survey. This includes a user's perceptions of the physical and social environment, and their perceptions of the people in their general vicinity. Expanding these questions to include perceptions and actions taken will help to assess dynamic and static perceptions of safety.

As mentioned earlier in this chapter, the addition of automated notification parameters, and the extraction of collected data points linked directly to a web or user map would add value to this application.

6.5.3 Future GIS Analysis with EMA data

The availability of an EMA mobile application to collect data that is exportable to a GIS provides geospatial analysts with the information required to analyze and compare data in a spatial context. The focus of this study was to collect real time data from mobile application users and track their perceptions of safety in real time. Tracking in real time provides analysts with the tools to study the user's feelings at the time of collection. As discussed in chapter 2, prior studies would track perceptions statically and would result in measurements that were likely to become equated with a perceived risk based on subjective probability rather than a reflection of actual experience (Jackson 2015).

The dynamic collection of data, and the possibility of syncing this data to a web map, would provide analysts with the tools to track this data in relation to other dynamic data sets, such as daily crime reports. This type of analysis could be performed on the street level, as both data sets are collected on a micro-scale. On the other side of the scale, the distribution of dynamic data sets from this EMA mobile application could be compared to micro-scale discrete data sets, such as U.S. Census block data, American Community Survey (ACS) data, and local police area commands. The integration of census and ACS data would supply analysts with the tools to compare EMA user responses with demographic data to gauge if bias is a factor in the user's avoidance and fear responses. Police commands can also be integrated to determine which areas have the highest rate of negative perceptions to help law enforcement determine where to focus priorities for fear reduction and updated planning. Table 2 provides a list of additional data sets enough for this project's study area.

Table 2. Additional data sets

Layer	Source	Availability
ABQ Crime Mapping Data	APD/Crimemapping.com	Open source data/updated daily/six-month time frame
APD Area Commands	CABQ Maps Open Data Portal	Open source
Bernalillo County Census Tracts (2010)/American Community Survey (2017)	https://www.census.gov/cgi-bin/geo/shapefiles/index.php?year=2017&layergroup=Census+Tracts https://www.census.gov/programs-surveys/acs/	Freely Available

The tools analysts require to produce results depends on the data sets used and the type of study performed. Within the ArcGIS Pro environment, the Mapping Clusters toolset can be used to identify the locations of statistically significant hot spots, cold spots, and spatial outliers (ArcGIS Pro 2018). Identifying hot spots through hot spot analysis and density-based clustering can provide analytical data to end users to identify areas where individuals perceive crime to be prevalent. These statistically significant spatial clusters can also be compared to actual crime clusters within the same time period to determine if fear of crime perceptions match with actual crimes.

Finally, since the survey questions are organized on a one to five scale, with a balance between positive and negative perceptions, it would be beneficial to use a Likert scale to scale the responses provided in the mobile application. Averages of personal perceptions could be generated to provide a general overview of positive and negative responses for each question provided. This information can be used to determine which parts of a geographical area contain a higher level of positive or negative perceptions in relation to varying input from respondents with different demographic backgrounds. This data would be available for city leaders and law enforcement to assist with determining priority areas for fear reduction and changes to city planning initiatives.

References

Android Developers. 2018. “Use testing tracks to get invaluable early feedback from users.” Google Play. Last modified May 22. Accessed July 20, 2018. https://developer.android.com/distribute/best-practices/launch/test-track

Android Developers. 2018. “Request App Permissions.” Google Play. Last modified November 21. Accessed September 15, 2018. https://developer.android.com/training/permissions/requesting

Android Developers. 2018. “Android Studio.” Google Play. Last modified November 8. Accessed September 15, 2018. https://developer.android.com/studio/intro/

Android Developers. 2018. “Distribution Dashboard.” Google Play. Last modified October 27. Accessed October 5, 2018. https://developer.android.com/about/dashboards/

Android Developers. 2018. “App Manifest Overview.” Google Play. Last modified November 8. Accessed September 10, 2018. https://developer.android.com/guide/topics/manifest/manifest-intro

Android Developers. 2018. “Permissions Overview.” Google Play. Last modified November 20. Accessed September 8, 2018. https://developer.android.com/guide/topics/permissions/overview#permission-groups

Android Developers. 2018. “Get the last known location.” Google Play. Last modified April 25. Accessed September 6, 2018. https://developer.android.com/training/location/retrieve-current

Android Developers. 2018. “Platform Architecture.” Android Software Stack. Accessed November 9, 2018. https://developer.android.com/guide/platform/

ArcGIS. “World Street Map.” Last modified November 29, 2018. Accessed October 4, 2018. https://www.arcgis.com/home/item.html?id=3b93337983e9436f8db950e38a8629af

ArcGIS Pro. “An overview of the Mapping Clusters toolset”. Last modified November 2018. Accessed January 10, 2019. http://pro.arcgis.com/en/pro-app/tool-reference/spatial-statistics/an-overview-of-the-mapping-clusters-toolset.htm

Ballesteros, Jaime, Bogden Carbunar, Mahmudar Rahman, Naphtali Rishe and S.S. Iyengar. 2014. "Towards Safe Cities: A Mobile and Social Networking Approach.” IEEE Transactions on Parallel and Distributed Systems 25, no. 9 (September): 2451-2462. Accessed June 12, 2018. DOI: 10.1109/TPDS.2013.190

Barocas, Solon and Andrew Selbst. 2016. “Big Data’s Disparate Impact” *California Law Review* 104, no. 3 (June): 671.

Bendler, Johannes, Tobias Brandt, Sebastian Wagner, and Dirk Neumann. 2014. "Investigating Crime-to-Twitter Relationships in Urban Environments – Facilitating a Virtual Neighborhood Watch.” Paper presented at proceedings of the European Conference on Information Systems (ECIS), Tel Aviv, Israel, June 9-11. Accessed June 10, 2018. http://aisel.aisnet.org/ecis2014/proceedings/track11/10

Braga, Anthony A., Andrew V. Papachristos, and David M. Hureau. 2014. "The effects of Hot Spots Policing on Crime: An updated Systematic Review and Meta-Analysis." *Justice Quarterly* 31. No. 4 (July): Accessed July 12, 2018. DOI: 10.1080/07418825.2012.673632

Breetzke, Gregory D and Amber L. Pearson. 2014. "The fear factor: Examining the spatial variability of recorded crime on the fear of crime." *Applied Geography* 46 (January): 45-52. Accessed June 14, 2018. DOI: 10.1016/j.apgeog.2013.10.009

Burke, Lora, Saul Shiffman, Edvin Music, Mindi Styn, Andrea Kriska, Asim Smailagic, Daniel Siewiorek, Linda Ewing, Eileen Chasens, Brian French, Juliet Mancino, Dara Mendez, Patrick Strollo, and Stephen Rathburn. 2017. "Ecological Momentary Assessment in Behavioral Research: Addressing Technological and Human Participant Challenges." *Journal of Medical Internet Research* 19, no. 3 (March): E77. Accessed July 14, 2018. http://search.ebscohost.comlibproxy2.usc.edu/login.aspx?direct=true&db=lls&AN=122323755&authtype=sso&custid=s8983984

Callanan, Valerie J. 2005. *Feeding the Fear of Crime: Crime related Media and Support for Three Strikes.* New York, NY: LFB Scholarly Publishing LLC.

Chataway, Michael L., Ross Coomber, and Christine Bond. 2017. "The geography of crime fear: A pilot study exploring event-based perceptions of risk using mobile technology." *Applied Geography* 86 (September): 300-307. Accessed June 15, 2018. DOI: 10.1016/j.apgeog.2017.06.010

Clark, Julie. 2003. "Fear in Fear-of-crime." Psychiatry, Psychology and Law 10, no.2 (June): 267-282. DOI: 10.1375/pplt.2003.10.2.267

Doran, Bruce J. and Melissa B. Burgess. 2012. *Putting Fear of Crime on the Map: Investigating Perceptions of Crime Using Geographic Information Systems.* New York, NY: Springer Science+Business Media, LLC.

Ferguson, Andrew G. 2017. *The Rise of Big Data Policing.* New York, NY: New York University Press.

FBI News. 2018. "2017 Preliminary Semiannual Crime Statistics Released." Federal Bureau of Investigation. Last modified January 23. Accessed June 14, 2018. https://www.fbi.gov/news/stories/2017-preliminary-semiannual-crime-stats-released

Firebase. 2018. "Pricing Plans." Accessed November 12, 2018. https://firebase.google.com/pricing/

Firebase. 2018. "Structure your database." Last modified October 25, 2018. Accessed September 28, 2018. https://firebase.google.com/docs/database/web/structure-data

Flutter. 2018. "FAQ." Accessed November 26, 2018. https://flutter.io/docs/resources/faq

Garrigos-Simon, Fernando J., and Ignacio Gil-Pechuan. 2015. *Advances in Crowdsourcing.* Edited by Sofia Estelles-Miguel. New York, NY: Springer International Publishing, 2015. Golbeck, Jennifer. 2015. *Introduction to Social Navigation* Boston: Elsevier Science.

Gassner, David. 2016. “Android Development Essential Training: Create Your First App.” Lynda, November 30, 2016. Accessed February 3, 2018. https://www.lynda.com/Android-tutorials/Android-App-Development-Essentials-Create-Your-First-App/518053-2.html

Gassner, David. 2015. “Android App Development: Google Maps.” Lynda, August 18, 2015. Accessed February 4, 2018. https://www.lynda.com/Android-tutorials/Adding-Google-Maps-Android-Apps/414068-2.html

Goodchild, Michael F. 2004. “Social Sciences: Interest in GIS Grows.” 26 (Spring): 1-4. http://www.esri.com/news/arcnews/spring04articles/social-sciences.html.

Google Maps Platform. 2018. “Pricing and plans.” Accessed November 12,2018. https://developers.google.com/maps/previous-pricing

Google Maps Platform. 2018. “Map Objects.” Last modified November 26, 2018. Accessed September 25, 2018. https://developers.google.com/maps/documentation/android-sdk/map

Hackernoon. 2017. “Introduction to Firebase.” GeekyAnts, December 28, 2017. Accessed November 2, 2018. https://hackernoon.com/introduction-to-firebase-218a23186cd7

Hale, C. 1996. “Fear of Crime: A Review of the Literature.” *International Review of Victimology* 4, no. 2 (January): 79-150. DOI: 10.1177/026975809600400201

Hamilton, M. F. Salim, E. Cheng, and S. Choy. 2011. “Transafe: A Crowdsourced Mobile Platform for Crime and Safety Perception Management.” *ACM SIGCAS Computers and Society* 41, no. 2 (December): 32-37. Accessed June 10, 2018. http://dx.doi.org/10.1145/2095272.2095275

Hipp, John R. 2010. “Resident Perceptions of Crime and Disorder: How Much is “Bias”, and How Much is Social Environment Differences?” *Criminology* 48, no 2 (May): 475-508.

Intravia, Jonathan, Kevin T. Wolff, Rocio Paez, and Benjamin Gibbs. 2017. “Investigating the relationship between social media consumption and fear of crime: A partial analysis of mostly young adults.” *Computers in Human Behavior* 77 (December): 158-168. Accessed June 18, 2018. DOI: 10.1016/j.chb.2017.08.047

Jackson, J. 2015. “Cognitive closure and risk sensitivity in the fear of crime.” *Legal and Criminological Psychology* 20, no 2 (September): 222-240. Accessed August 5, 2018. DOI: 10.1111/lcrp.12031

Jefferson Brian J. 2018. “Predictable Policing: Predictive Crime Mapping and Geographies of Policing and Race.” *Annals of the American Association of Geographers* 108, no. 1 (January): 1-16. Accessed July 12, 2018. DOI: 10.1080/24694452.2017.1293500

Kwan, Mei-Po. 2000. “Interactive geovisualization of activity-travel patterns using three-dimensional geographical information systems: a methodological exploration with a large data set.” *Transportation Research Part C* 8, no. 1: 185-203. Accessed January 12, 2019. DOI: 10.1016/S0968-090X(00)00017-6

Krill, Paul. 2017. "Apple's Swift is losing developers to multiplatform frameworks." InfoWorld, October 6, 2017. Accessed November 10, 2018. https://www.infoworld.com/article/3231664/application-development/apples-swift-is-losing-developers-to-multiplatform-frameworks.html

KOAT. 2018. "Already short-staffed APD asks for exemption from border deployment." KOAT News. Last modified April 5. Accessed June 15, 2018. http://www.koat.com/article/officers-who-serve-in-national-guard-could-be-deployed/19697308

Lane, Jodi, and James W. Meeker. 2003. "ethnicity, information sources, and fear of crime." *Deviant Behavior* 24, no. 1 (January): 1-26. DOI: 10.1080/10639620390117165

Masden, Christina, Catherine Grevet, Rebecca Grinter, Eric Gilbert, and W. Edwards. 2014. "Tensions in Scaling-up Community Social Media: A Multi-Neighborhood Study of Nextdoor." Paper presented at SIGCHI Conference on human factors in computing systems, 26 April. Accessed June 6, 2018. DOI: 10.1145/2556288.2557319

Melo, Alysson. 2016. "Firebase vs. Parse Server." Back4App Blog, March 23. Accessed July 19, 2018. https://blog.back4app.com/2016/06/15/firebase-parse/

Mennis, Jeremy, Mason Michael, Andreea Ambrus, Thomas Way, and Kevin Henry. 2017. "The spatial accuracy of geographic ecological momentary assessment (GEMA): Error and bias due to subject and environmental characteristics." *Drug and Alcohol Dependence* 178, (September): 188-193. Accessed July 10, 2018. http://dx.doi.org/10.1016/j.drugalcdep.2017.05.019

McCall, Jeffrey M. 2007. *Viewer discretion advised: taking control of mass media influences.* Lanham, Maryland: Rowman & Littlefield Publishers, Inc.

McNaughton-Cassill, Mary E. 2017. "We're consuming too much media. It's time to detox our brains." Recode, February 23, 2017. Accessed June 15, 2018. https://www.recode.net/2017/2/23/14669710/reprogram-brain-media-overload-political-fake-news

O'Connell, Michael. 1999. "Is Irish Public Opinion towards Crime Distorted by Media Bias?" *European Journal of Communication* 14, no. 2 (June): 191-212. Accessed June 18, 2018. doi: 10.1177/0267323199014002003

Perry, Walt, Brian McInnis, Carter C. Price, Susan C. Smith, and John S. Hollywood. 2013. *Predictive Policing: the role of crime forecasting in law enforcement operations*. Santa Monica: RAND Corporation.

Razip, M.M., Abish Malik, Shehzad Afzal, Matthew Potrawski, Ross Maciejewski, Niklas Yun Jang, David S. Elmqvist, and David S. Ebert. 2014. "A Mobile Visual Analytics Approach for Law Enforcement Situation Awareness." Paper presented at 2014 IEEE Pacific Visualization Symposium, Yokohama, Japan, March 2014. Accessed June 6, 2018.
DOI: 10.1109/PacificVis.2014.54

Rees-Punia, Erika., Elizabeth D. Hataway, and Jennifer L. Gay. 2018. "Crime, perceived safety, and physical activity: A meta-analysis." *Preventive Medicine* 111 (June): 307-313. Accessed June 14, 2018. DOI: 10.1016/j.ypmed.2017.11.017

Ross, Catherine E. and John Mirowsky. 1999. "Disorder and Decay: The Concept and Measurement of Perceived Neighborhood Disorder." *Urban Affairs Review* 34, no. 3 (January): 412-432. DOI: 10.1177/107808749903400304

Sakip, Siti Rasidah Md., Noraini Johari, and Mohd Najib Mohd Salleh. 2013."Perception of safety in gated and non-gated neighborhoods." *Social and Behavioral Sciences* 85 (September): 383-391. Accessed June 14, 2018. DOI: 10.1016/j.sbspro.2013.08.367

Scarborough, Brittney K., Toya Z. Like-Haislip, Kenneth J. Novak, Wayne L. Lucas, and Leanne F. Alarid. 2010. "Assessing the relationship between individual characteristics, neighborhood context, and fear of crime." *Journal of Criminal Justice* 38, no. 4, 819-826. Accessed June 18, 2018. DOI: 10.1016/j.jcrimjus.2010.05.010

Smyth, Neil. 2017 *Android 3.0 Development Essentials.* Columbia, SC: Payload Media, Inc.

Solymosi, Reka., Kate Bowers, and Taku Fujiyama. 2015. "Mapping fear of crime as a context-dependent everyday experience that varies in space and time." *Legal and Criminological Psychology* 20, no.2 (September): 218-221. Accessed June 15, 2018. DOI: 10.1111/lcrp.12084

Statista. 2018. "Number of mobile phone users worldwide." The Statistics Portal. Last modified 2018. Accessed June 7, 2018. https://www.statista.com/statistics/274774/forecast-of-mobile-phone-users-worldwide/

Statista. 2018. "Number of apps available in leading app stores as of 1st quarter 2018." The Statistics Portal. Lat modified 2018. Accessed June 7, 2018. https://www.statista.com/statistics/276623/number-of-apps-available-in-leading-app-stores/

Sui, Daniel, Sarah Elwood, and Michael Goodchild. 2013. *Crowdsourcing Geographic Knowledge.* New York, NY: Springer Netherlands:Dordrecht.

World Population Review. "Albuquerque, New Mexico Population 2018." Last modified June 12, 2018. Accessed June 16, 2018. http://worldpopulationreview.com/us-cities/albuquerque/

Udacity. 2017. "Firebase in a weekend: Android". Accessed September 12, 2018. https://www.udacity.com/course/firebase-in-a-weekend-by-google-android--ud0352

U.S. Department of Justice. Civil Rights Division. 2014. "Letter to Richard J. Berry, Mayor of Albuquerque" by Jocelyn Samuels and Damon P. Martinez.

U.S. News and World Report, "New Mexico's Exodus Problem." Last modified August 7, 2017. Accessed June 16, 2018. https://www.usnews.com/news/best-states/articles/2017-08-07/why-new-mexicans-are-fleeing-the-state

Viswanathan, Priya. 2018. “iOS App Store Vs. Google Play Store for App Developers.” Lifewire, August 27, 2018. Accessed November 18, 2018. https://www.lifewire.com/ios-app-store-vs-google-play-store-for-app-developers-2373130

YouTube. 2017. “How to Use Android Firebase.” CodingWithMitch, November 15, 2017. Accessed September 25, 2018. https://www.youtube.com/playlist?list=PLgCYzUzKIBE_cyEsXgIcwC3P8ipvlSFd_

YouTube. 2017. “Google Maps & Google Places Android Course.” CodingWithMitch, October 3, 2017. Accessed August 29, 2018. https://www.youtube.com/playlist?list=PLgCYzUzKIBE-vInwQhGSdnbyJ62nixHCt