HanApp: A Missing Persons Reporting and Tracking Application in the Philippines

A Special Problem
Presented to
the Faculty of the Division of Physical Sciences and Mathematics
College of Arts and Sciences
University of the Philippines Visayas
Miag-ao, Iloilo

In Partial Fulfillment
of the Requirements for the Degree of
Bachelor of Science in Computer Science by

GUIDES, Emmanuel Tarek Shayne
HISMAÑA, Nikko Gabriel
TORCULAS, Erru

Francis DIMZON
Adviser

December 23, 2022

Abstract

Incidents of people going missing is an ongoing problem in the country. Although Missing Persons cases typically occur after natural or man-made disasters, estimates and reports indicate that a sizable number of Missing Persons instances still happen outside of these settings with children and elderly people with cognitive disabilities being the most at-risk. Although the Philippine National Police (PNP) has created measures to address this problem, the reporting system established has not leveraged the use of technology to streamline the process.

This research aims to address these issues by developing an application that streamlines the Missing Persons case reporting and verification process, enables and encourages community-based Missing Persons search efforts, and allows users to track loved ones who are at risk of going missing. The application, "HanApp", will be developed using Flutter Framework for cross-platform compatibility, and shall utilize GPS technology, Google Maps Platform APIs, and Firebase serverless framework to achieve these features.

Preliminary results based on extensive review of related studies, applications, and documentations support the feasibility of developing the proposed framework and features of HanApp using the aforementioned technologies.

Keywords: Missing Persons, Philippine National Police, case reporting and verification, community-based search, location tracking, Flutter framework, GPS, Google Maps Platform APIs, Firebase serverless framework.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview of the Current State of Technology

As defined by the International Commission on Missing Persons, subjectively, anyone who is being sought by at least another person and whose location or whereabouts are unknown can be classified as a missing person (International Commission on Missing Persons, 2022). However, each country has their own standard or policies for legally defining a "missing" person, and as such, accurate statistics on the average rate of the disappeared globally are harder to comprehend. This, together with the unconfirmed number of the unreported cases of missing persons, people who voluntarily go missing, or even victims of disasters and conflict, brings to light how obscure and challenging a search for a missing person can be.

With this in mind, there have been some efforts globally to implement technologies and systems for organizing and solving this problem. The AMBER (America's Missing Broadcast Emergency Response) alert system, while not a standalone application, is a system that geo-locates and uses various forms of effective media such as smartphones, television, or radio to disseminate information about missing or abducted children in the United States in order to elicit citizen tips and responses in order to expedite the rescue of the said missing or abducted children (Griffin, Miller, Hoppe, Rebideaux, & Hammack, 2007).

NamUs, or the National Missing and Unidentified Persons System in the United States, is an integrated set of two databases: one for the information about unrecorded persons whose remains are inside the United States, and another for all profiles of missing persons and their information. Being publicly accessible for all entries and searches, the NamUs database has contributed to multiple projects, studies, works, and investigations for law enforcement (Murray, Anderson, Clark, & Hanzlick, 2018).

A push for a national Missing and Found Person Database (MFPD) in a 2016 Memorandum Circular from the National Police Commission in the Philippines has defined the database as a repository of all the names and relevant information about reported missing and found persons in the country. A website called the "Missing Person Website" was also defined in the memorandum, with the purpose of posting and displaying the name, picture, and other relevant information about missing and found persons (National Police Commission, 2016). The said database, however, is not open to the public unlike the NamUs. With further probing and scouring, the researchers have found that there is no clear, standardized, and modernized level of technology applied in the filing, dissemination, and searching for missing persons in the Philippines.

Currently, the standard procedure for filing missing persons (MP) cases in the Philippines is through the desk officers in their respective local police offices. This procedure can be both sluggish and counter-intuitive, especially in cases in which the missing person is a child or a victim of an accident, where the recording, investigation, and monitoring of the case needs to begin immediately (National Police Commission, 2016).

The researchers want to create an application that leverages current technology to streamline the processes required in the filing of missing persons cases and follows a similar formula to the AMBER alert system, which uses geo-location not only in the targeting of alerts but also in the tagging of the missing person's location, and through which it can both incite community levels of cooperation as well as dissemination of PNP-verified missing persons reports.

1.2 Problem Statement

The Philippine Statistics Authority (2021) have only reported missing persons statistics for cases involving natural and man-made disasters: 5,000 MPs in 2020 and 12,000 MPs in 2021 (Philippine Statistics Authority, 2021). On the other hand, the Philippine News Agency (2021) shared data from the DOJ which reported over 1.2 million reports of missing and/or exploited children in 2020, and 2.8 million reports in 2021 (Pulta, 2021). These, unfortunately, are only among the few publicly available numbers provided by national agencies with regards to MP cases in the Philippines.

Orion Support Incorporated (2015) estimated 35,000 MP reports each year in the country. With this, it relatively coincides with the rate of 1.7 persons per 1,000 Filipinos (Orion Support Incorporated, 2021). However, the obscurity of reporting these missing cases has troubled not only the community but also authorities to craft effective measures in finding them in a short span of time — mitigating any factors that will further put the individual in danger especially if they are part or is a person-at-risk-of-going-missing (PARGOM).

The dire situation of these MP cases causes distress to their beloved in which our culture is deeply ingrained to familiarity and closeness of kinship or other relationships in the same nature. Moreover, unresolved MP cases in the Philippines have drastic effects on the socio-economic health of the community, affecting not only the MP's immediate family but the missing individual's network and area. As MP cases remain unresolved, the sense of security in the area as well as the competence of local authorities are put into question, causing friction and restrictions.

Although the PNP adopted MC 2016-033 which established a unified Missing and Found Persons Database (MFPD) and the process for receiving and handling MP reports in order to handle this issue, these guidelines stipulated that MP case reporting can only be done in person in the station of the area where the disappearance occurred (National Police Commission, 2016). Requiring the MP case reportee to travel and file the case in-person takes away valuable time, and neglects to utilize existing technology to streamline the process and have them receive, verify, and act on the case immediately. Moreover, the PNP has advised the public against posting and sharing of unverified MP cases on social media (Madarang, 2022).

As such, the researchers aim to create an application that leverages the use of existing technology to streamline and accelerate the reporting and verification process, allow users to be notified of PNP-verified missing persons cases in their area, track the location of loved ones whom they consider as persons-at-risk-of-going-missing (PARGOM), and assist PARGOMs in getting help when lost.

1.3 Research Objectives

1.3.1 General Objective

The general objective of this study is to develop a mobile application that has multiple interfaces aimed at multiple target users, such as parents, their children, family members who are at risk of going missing (i.e. children or the elderly with dementia), the PNP, and many others, all in order to help file, verify, disseminate information, and know the whereabouts of missing persons.

1.3.2 Specific Objectives

This study specifically seeks:

1. To develop an application that would allow users to easily report missing persons to the PNP and receive updates on such reports, as well as a PNP counterpart application that will enable the PNP to quickly receive, verify, manage, and respond to reports.
2. To integrate a serverless database system for PNP-verified reports that enables location-based notifications so that app users can view information about missing persons nearby so they can assist in the search.
3. To integrate location and global position system (GPS) services so that users can check on the last-known location of a PARGOM, provided that the PARGOM has linked their account to the user's.
4. To develop the companion application with a simpler interface in which users who are identified as PARGOMs can link their account with another user

(e.g. child-parent) for simple and straightforward location sharing between said users.

1.4 Definition of Terms and Acronyms

Child - refers to any person under the age of 18

Database - also referred to as **"serverless database"**, refers to the application's own public database containing the reports filed to the PNP pending their verification, the PNP-verified missing persons reports which will be used to notify nearby users of missing persons cases in their area

HanApp - also referred to as "the application" or "proposed application", will refer to the application and its various interfaces being developed in this study. Interfaces:

- **Main (or parent) App** - refers to the main application that has the features: reporting and getting updates on missing persons cases, locating the user of the companion app
- **Companion App** - refers to the limited-feature application interface to be used by persons at risk of going missing (e.g. a child or a person with dementia)
- **PNP Admin App** - refers to the application interface that is accessed by the PNP in which they receive, manage, verify, and provide updates on filed reports

Located - will be used to denote that the missing person has been identified and reunited with their family or guardian.

Missing Persons (MP) - will refer to any person who is classified under the PNP guidelines as missing; an adult person who is missing is required under the guidelines to have not been located after 24 hours, whereas a child that has gone missing is immediately referred to as an MP.

Missing and Found Person Database (MFPD) - refers to the PNP's own private database where they manage Missing and Found Persons cases, and will not be accessible by the application.

Person-at-risk-of-going-missing (PARGOM) - refers to any person that the main app user deems to be at risk of going missing and is unable to find their way back, such as a child or an elderly person with dementia. If an adult PARGOM has not been located, they are technically not classified by the PNP as MP until 24 hours from perceived disappearance.

Reportee - refers to the person filing the missing person report via the application.

1.5 Scope and Limitations of the Research

The application primarily aims to streamline and hasten the filing and verification of missing persons reports to the PNP and prevent the dissemination of unverified missing persons reports which the PNP advises against.

The application, at least for its current proposed build, mainly focuses on the filing and locating of missing persons in scenarios not involving post-disaster search and relief operations (i.e. after an earthquake, typhoon, landslide).

Moreover, this application utilizes a serverless approach with the integration of Firebase, not to mention, services like Google Maps and other location based services all rely on stable internet connectivity, be it through mobile data or Wi-Fi.

Finally, in order to facilitate information dissemination of PNP-verified missing persons reports, and to notify and allow the community in assisting with the search for missing persons in their area, a public serverless database will be created storing the PNP-verified missing persons reports. It is important to note that this is a separate database from the PNP's own "Missing and Found Persons" database.

1.5.1 User and Information Access

The application will only have three interfaces which will cater to its main three users: the general user, the PNP personnel/helpdesk in charge of missing persons cases, and persons at risk of going missing (PARGOMs). Since this is a mobile application (with a desktop interface for the PNP) and the scope of the application is only national, only users in the Philippines with compatible smartphones and access to the internet will be able to use the app.

Due to the sensitivity of the data being handled, it's important to define the scope and limits of each user's access to address data privacy and security concerns.

1. The general users refer to the general public who either wish to report a missing persons case, assist with search efforts for missing persons, or wish to keep track of and locate a loved one who is at risk of going missing. They will only have access to information regarding their registered account, the updates provided by PNP on the report they filed, PNP-verified missing persons in their area, and the location information of the PARGOMs linked to their account.
2. The PNP personnel/helpdesk in charge of missing persons cases and have access to their HanApp PNP account will only have access to information provided through reports, and will not be given access to any other information from the users.
3. The PARGOMs users are any persons who are at risk of going missing and may not have the capability to find their way back (e.g. children or those with cognitive disabilities such as dementia). They will only have access to their account information, their location, and the directory and location of the nearest PNP station or helpdesk.

Moreover, the application's administrators and developers will only have access to the serverless database containing the list of verified missing persons cases and their locations, and the account names at most, like user passwords, and location information will not be accessible by anyone apart from the users themselves.

The PNP's own private "Missing and Found Person Database" and all information therein will be separate from the serverless database to be used for storing

verified missing persons cases, and will not be accessed by the application, the application's users, nor its administrators.

1.6 Significance of the Research

The research's suggested mobile application and its framework aim to consolidate and streamline the process of filing and verifying missing persons reports, and also allow for an efficient way to disseminate PNP-verified missing persons reports. Thus, accelerating the process of resolving missing persons cases through quicker report verification, location-based alerts, and leveraging community-based search efforts.

This application and framework will be of great significance for protective services and individuals whose loved ones have gone missing by allowing them to report and locate the whereabouts of these missing individuals with ease.

In addition, the data that will be borne out of this research will be a pivotal factor in current events and statistics relating to missing persons. In fact, the focal point of this research paves the way for a systematic and technological breakthrough here in the Philippines to create a mobile application that will positively influence resolving missing person cases. Unequivocally, the novelty of this study provides a baseline framework for similar studies impacting the creation of a more robust system in the near future.

Chapter 2

Review of Related Literature

2.1 Mobile Applications for Persons at Risk of Going Missing (PARGOM)

As defined by the previous sections, any person that is experiencing trouble with navigating themselves can be categorized as a person at risk of going missing or PARGOM. This includes children, the elderly, or anyone experiencing medical, mental, or cognitive conditions that inhibit their ability to safely navigate back to their family or domicile.

A mobile alert application determined to engage community volunteers to help in locating missing persons with dementia called the "Community ASAP system" was developed and documented in a paper by Neubauer, N., Daum, C., Miguel-Cruz, A., and Liu, L.. The findings of the study's simulation of the Community ASAP system highlighted the importance of police services in these cases on account of their primary and direct involvement, and the effectiveness of community response and participation to the occurrence of missing persons with dementia. The approach also proves to be viable even for people who have no social media which is popular for being an accessible medium to disseminate information regarding MPs (Neubauer, Daum, Miguel-Cruz, & Liu, 2021).

"CoSMiC", a mobile application designed to crowdsource information about lost

and missing children in situ is another approach towards applying techniques and technologies in locating persons at risk of going missing. The main concern that the CoSMiC mobile application prioritizes to solve is with regards to the urgency and criticality that ensues whenever a child goes missing within a neighborhood. The application aims to digitize crowdsourcing for finding missing children through a landmark-based location history of the lost child that was chronologically and locationally procured within the network of crowdsourced information (Shin et al., 2014).

The urgency, reasoning, and crowdsourcing proposition and concerns by the above studies also reflect one of the main aims of the application being proposed; the care, prioritization, and the adaptation of approach towards PARGOMs.

2.2 Serverless Application Model

Serverless does not mean "server-less". Serverless Computing is an up-and-coming paradigm and framework for the development and deployment of multiple applications, now relying on services through the cloud. A recent shift of enterprise applications' architectures into containers, microservices, and serverless backend services has pushed the paradigm even more. Serverless platforms offer new features that make creating scalable microservices and applications easier and more cost efficient, promoting themselves as the next stage in cloud computing architectural evolution (Castro, Ishakian, Muthusamy, & Slominski, 2017). One example of an application development software used in a serverless framework is Google's Firebase, an example of BaaS (Backend-as-a-Service).

The proposed application also utilizes serverless computing as its paradigm. Knowing the general scenarios, perception, and support towards this new paradigm in software development is essential as it helps in the factoring and decision-making of the developers during the development of the proposed application.

2.2.1 Google Firebase

As defined by Khawas, C. and Shah, P. in their study titled "Application of Firebase in Android App Development-A Study" (2018), Firebase is one of the relatively new and even faster approaches towards handling large amounts of unstructured data as compared to the traditional Relational Database Management Systems (RDBMS) through developing serverless applications (Khawas & Shah, 2018).

In a paper by Hannula, T. (2021) titled "Unity mobile application with a serverless Firebase backend", where a lo-fi themed Android mobile application prototype was developed, both NoSQL database services that are Firebase Realtime Database and Cloud Firestore were both utilized in managing the backend of the application the author has developed. Both services have enabled the application to be simplified and streamlined as Firebase handles the maintenance of the data in the Realtime Database and Cloud Storage (Hannula, 2021).

As stated by the definition and example above, Google Firebase, an approach for serverless computing, is a promising choice for the chosen paradigm of the proposed application, serverless. Knowing the benefits and how Firebase was implemented on mobile applications is crucial in the development of the proposed application.

2.3 Global Positioning System and GPS Applications

The Global Positioning System (GPS) is, in itself, a United States-owned service that offers positioning, navigation, timing (PNT) services to its users. As it is free, open, and reliable, GPS has been used and integrated countless times on a myriad of applications, including ones that are implemented in mobile platforms (GPS.gov, n.d.).

A seamless application of the Global Positioning System to Android mobile phones was implemented in a paper titled "Abhaya: An Android App for the Safety of

Women" by Yarrabothu, R. S., and Thota, B. (2015). The paper describes an app called "Abhaya", where it employs a quick and easy-to-use alert button from the application integrated within the smartphone, in which a single tap can identify the location or place of the user through the use of GPS services and would thus then send a message comprising the location URL to all registered contacts of the user. Additionally, Abhaya also calls the first person in the registered contact list of the user and periodically sends an SMS every five minutes to the registered contacts until the stop button has been tapped (Yarrabothu & Thota, 2015).

A paper titled "Mobile phone application for reporting and tracking missing persons in Kenya ", by Elizabeth Mutisya for the Strathmore University (2017), describes developing a centralized system and a mobile application, that hopes to organize the rather inefficient process in Kenya with regards to addressing missing persons cases. The said application uses GPS to determine and tag the location of a missing person, given that missing person has the application on hand, as well as of the sightings reported to increase the effectiveness of the app. It also utilizes the currently centralized database, National Missing and Unidentified Persons System (NamUs) in Kenya to both counter-check, verify, and assist in finding these said missing persons. A web application was also developed in order to assist those who would still want to access the software but do not have the smartphone needed for it to run the application (Android). The application was developed incrementally, starting on a smaller scale and gradually increasing in complexity using the Agile methodology (Mutisya, 2017).

Mutisya, E. (2017) has proposed the utilization of the Global Positioning System (GPS) in the assistance in providing more details about the disappearance of a missing person. This approach has exemplified the fact that GPS services can be applied in finding missing persons. Not to mention, the development methodology used in the paper is Agile as well, which is similar to the proposed application's methodology, too. The Abhaya application (Yarrabothu & Thota, 2015) eventually provided the option for continuous tracking of the user or the missing person, provided that he or she has enabled the GPS in the proposed application, as periodically sending SMS instead of real-time tracking requires less processing power compared to the latter.

Overall, the utilization of the Global Positioning System (GPS) has provided an exemplary mechanism for tracking down missing persons by either identifying

the last location the person was found or a real-time continuous tracking. This employs heightened identification of an individual's whereabouts thus expediting the searching process.

2.3.1 Google Maps Platform

There have been numerous online mapping services available to the public, arguably the most popular of which is Google Maps. Google Maps, a web mapping service by Google, provides satellite imagery, street maps, real-time traffic conditions, and also route planning (Antony, 2021). Its closest competitor, Bing Maps, was developed by Microsoft, and a study showed that both provided near-similar accuracy in geocoding (Kilic & Gülgen, 2020). However, due to the fact that Google Maps is more feature-rich, and has over 1 billion active users monthly — which translates to frequent location mapping and verification, Google Maps is considered to be the better option (Lookingbill, 2019).

Furthermore, Google has made it possible for developers to integrate and utilize Google Maps in the development of applications that require maps and location data. This is offered through the Google Maps Platform which is a collection of APIs, SDKs, and tools to easily embed and allow data retrieval from Google Maps to applications (Google Developers, n.d.). In fact, there have been numerous studies showing the application of Google Maps and their open APIs.

In Ghana, Google Maps together with GPS was used to successfully map accurate digital postal addresses to a separate application (GhanaPostGPS) by overlaying it over Google Maps through their API (Gah, Katsriku, & Gyamfi, 2018). Another study utilized GPS, Google Maps, and GSM technology (for texting location details) for parents and school authorities to accurately monitor children's location in a timely manner which could be valuable if they go missing (Sunehra, Priya, & Bano, 2016).

Google Maps and the Google Maps Platform's open APIs and SDKs, together with GPS technology, makes it a viable technology to be used for developing apps for locating MPs in the country.

2.3.2 Location based notification and alarm applications

Technology and tools such as GPS and Google Maps (and the Google Maps Platform APIs), has made it possible to create applications to notify or alarm the user when they're at (or near) certain locations. In 2013, an Android application was developed using Google Places API (offered in the Google Maps Platform) and GPS called "GEO ALERT" which would alert a traveler when they're at a certain spot and it would show them a history of that specific location (Garg & Shukla, 2013). Another study proposed a location-based notification and alarm Android application utilizing GPS and LBS (location-based service), and it notified the user if a friend is nearby and alarms when the user enters a marked location on the map (Kanfade, Ambade, & Bhagat, 2018). More recently in 2022, a study titled "Travellert: A Location Based Alarm Application" developed an application using GPS and Google Maps Platform APIs and GPS which alerted a traveler when they were near their destination to avoid missing their stops (Buatag, Garcia, & Visto, 2022).

The existence of these studies and location-based notification and alarm applications do not only provide proof of feasibility in using GPS, Google Maps, and similar technologies for the proposed application to track PARGOMs, but could serve as a basis on how to approach the development of the app's location-based notification system for alerting users of PNP-verified MP cases near them.

2.4 Other Missing Persons finder mobile applications

The paper written by Desale H., Tavasalkar P., Vare S., and Shintre R. titled "Android Crime reporter and Missing Person Finder", details some key insights that can be extremely helpful for the development of the project (Desale, Tavasalkar, Vare, & Shintre, 2020).

What is ingenious about this application is that the idea of having a "panic button" within the application can be a very useful feature. It can both limit the delays in relying on another user to notify the police about another user's

becoming missing by having the said user notify the police right away. One key difference with this application compared to the proposed application, however, is that this only sends case reports and notifications directly to the police with the sole purpose of expediting the filing of these cases, whereas on the other hand, the team's proposed application's main goal is dissemination of information with regards to the missing person.

Studying the paper as reference is worthwhile as it was written with both straightforwardness in the functionality in mind. The said authors wrote about the integration of a 'panic button' in the mobile application, which, actually, has inspired the researchers to come up with a take of it as well in the proposed companion version of the application.

Chapter 3

Research Methodology

This chapter lists and discusses the specific steps and activities that will be performed to accomplish the project. The discussion covers the activities from pre-proposal to Final SP Writing.

3.1 Research Activities

3.1.1 Development Framework

The proposed application is an interconnected and interdependent system, such that one feature is either dependent on or a prerequisite of another (i.e. the registration feature is required before the reporting and verification feature; the main-companion app binding feature requires that the main and companion app interfaces have been built).

Due to this reason, the researchers will adopt a modified version of the Feature-Driven Development (FDD) agile framework. FDD approaches software development by developing an overall model, listing all features (and how they interact), planning each feature, then focusing on designing and building one feature at a time (ProductPlan, 2022). This modified version of FDD will include feature documentation after each feature is built and integrated, and the system prototype will be released only when all features have been built, integrated, and tested.

As seen in Figure 3.1, the system will be developed by first developing the entire model, list all the features, plan the other by which to develop the features so that they can be integrated.

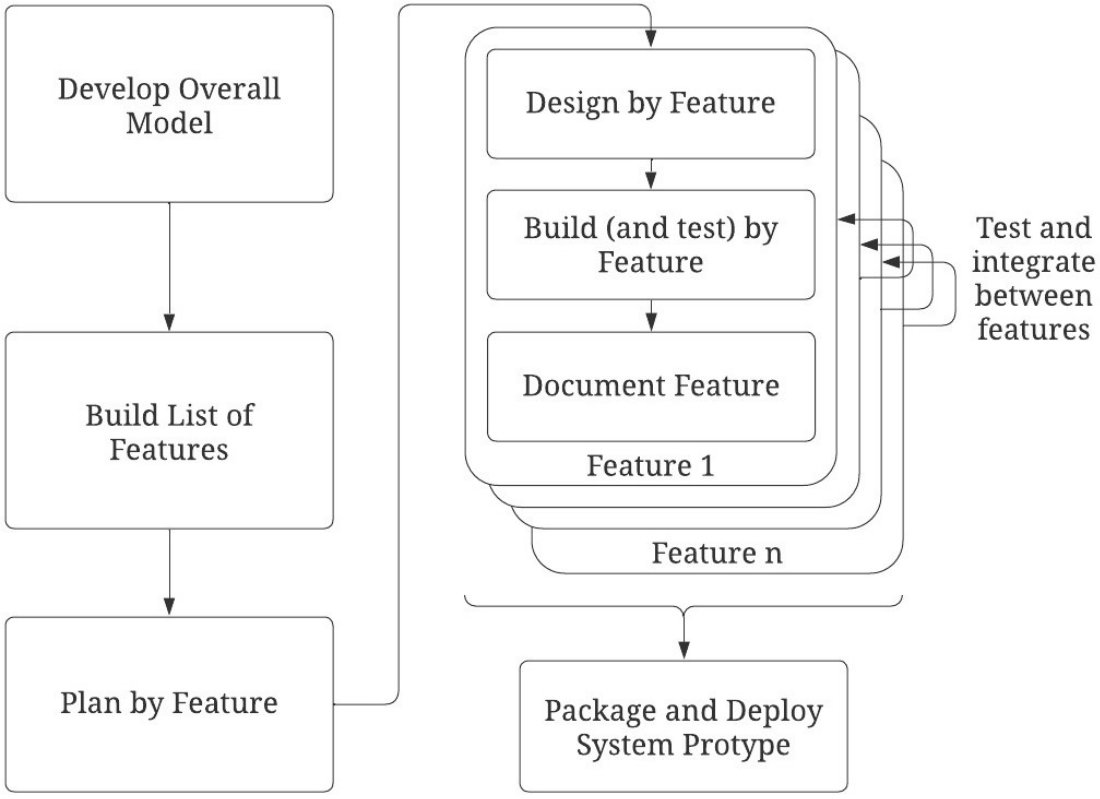

Figure 3.1: Modified Feature-Driven Development (FDD) Framework

3.1.2 Calendar of Activities

A Gantt chart showing the schedule of the activities should be included as a table. For example:

Figure 3.2 shows the Gantt Chart of all activities to be undertaken from the conception, to the development, and the analysis of the proposed application. Since the development framework used is FDD, development is scheduled by each feature.

The order of the features to be developed as per the table are as follows :

Activity	Sept	Oct	Nov	Dec	Feb	Mar	Apr	May
Identifying Problem	x							
Review Related Articles and Studies	x	x						
Formulate Solution and Overall Model		x	x					
Identify/Study Applicable Technologies		x	x	x				
Build List of Features/Requirements			x	x				
Plan order of Feature Development				x				
Design, Build, Document Feature 1					x			
Design, Build, Document Feature 2*					x			
Design, Build, Document Feature 3*						x		
Design, Build, Document Feature 4*						x		
Design, Build, Document Feature 5*							x	
Design, Build, Document Feature 6*							x	
Design, Build, Document Feature 7*							x	
Package and Deploy System Protoype								x
Testing Overall App System								x
Recording and Interpretation of Results								x
System-wide Documentation								x
Consultation	x	x	x	x	x	x	x	x

Figure 3.2: Timetable of Activities

- Feature 1 - Firebase serverless database backend
- Feature 2 - main app registration system
- Feature 3 - main app reporting and PNP Admin App verification feature
- Feature 4 - main app location-based notification feature for nearby PNP-verified reports
- Feature 5 - companion app and its “Find Me” feature
- Feature 6 - main-companion apps binding
- Feature 7 - main app “Find Companion” feature

3.2 Development Tools

3.2.1 Software

Github

GitHub is a web-based tool that utilizes Git, an open source version control that enables several users to make distinct modifications to applications or software simultaneously (*An introduction to GitHub*, 2020). GitHub is currently being used by over 94 million software developers, 4 million plus organizations, and has created over 330 million repositories for varied software (*About GitHub*, n.d.).

Visual Studio Code

Visual Studio Code is a compact yet capable source code editor for macOS, Windows, and Linux that runs on your desktop. It supports multiple programming and scripting languages like JavaScript, TypeScript, and Node.js, as well as a robust ecosystem of extensions for additional languages and runtimes (including C++, C#, Java, Python, PHP, Go, and.NET) (Microsoft, 2021).

Flutter

Flutter is a Google open-source framework used for creating attractive, locally built, multi-platform apps out of a single codebase. For rapid efficiency and performance on any device, Flutter code compiles to ARM or Intel machine code, as well as JavaScript. Dart, a programming language designed for speedy programs on any platform, powers it (*Flutter: Build Apps on any Screen*, n.d.). As the developers are aiming to deploy the proposed application on both the mobile (Android) platform for the main users and their companions, and on the Windows (desktop) platform for the side of PNP, Flutter is an outstanding choice out of all the available frameworks and languages.

Google Firebase

As Firebase has a variety of products and services that it provides, such as Firebase Cloud Messaging (FCM) for messages as well as notifications for Android, Web Applications, and iOS, Firebase Auth, which is a service that can authenticate users using only client-side code, Real-time Database, a NoSQL database service, and Firebase Storage, which is a file transfer service (Khawas & Shah, 2018), Firebase remains to be the most optimal choice for a serverless mobile application that may require the said services, such as with the proposed application.

Google Maps

Google Maps, one of the world's most influential applications (Mehta, Kanani, & Lande, 2019), provides multiple location services needed in the proposed application. This includes the pinging of the location of the user of the companion app, or perhaps even the pinpointing and updating of the location where a verified missing person was last seen.

3.2.2 Hardware

Android Phone

An Android phone is a type of smartphone that is operating using the operating system developed by Google, Android.

Laptop

The proposed application will be developed on laptop computers with the minimum specifications of an 8th generation Intel Core i3 CPU, and 8GB of RAM.

3.2.3 Application Programming Interfaces (APIs)

Geofencing API

The geofencing API lets the developers construct perimeters, also known as geofences, that encircle regions of interest. When a device crosses a geofence, your app receives a notice, allowing you to deliver a beneficial experience when users are nearby (*Geofencing API*, n.d.).

Geocoding API

The Geocoding API converts addresses directly into geographic coordinates, which may then be used to set markers on a map or position the map (*Geocoding API*, n.d.).

Distance Matrix API

The Distance Matrix API offers travel distances and times for a matrix of origins and destinations, with rows holding duration and distance values for each pair (*Distance Matrix API*, n.d.).

Maps Embed API

With the help of the Maps Embed API, developers may add a Street View panorama or interactive map to an application being developed through only using a straightforward HTTP request (*Maps Embed API*, n.d.).

Places API

The Places API is a service that employs HTTP requests to return information about locations. This API defines places as establishments, physical sites, or important points of interest (*Places API*, n.d.). This API will help the developers to pinpoint the location of the nearest and most logically sound police station, and

hence local PNP admin accounts, based on the location of the reports of missing persons.

3.3 Application Requirements

3.3.1 Backend Requirements

Listed below are the proposed and overall structure of all connections and relationships among all data, interfaces, users, and the serverless service. It is important to note as well that since this is still just the proposed architecture, the finalized form of it, and especially how data will be organized within Firebase, can be changed during the actual development if the developers ever find a more practical approach to it.

Serverless Architecture

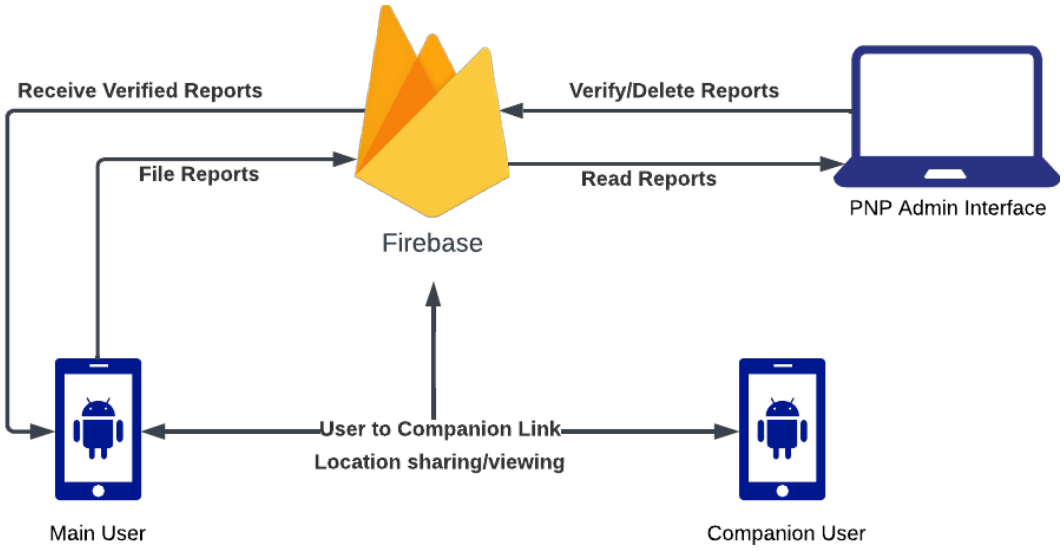

Figure 3.3: Client-Serverless Architecture with Firebase

The overall serverless architecture of the proposed application and its system is portrayed in the diagram above. Firebase, as the serverless service being utilized, will be the storage of all data being utilized on all three interfaces.

Realtime Database structure for User, Companion, and PNP admin profiles

Each profile type will be a node just under the root node as child nodes, namely, Main Users, Companions, and PNP Admins.

Realtime Database structure for sent reports

As Firebase's real-time database is structured as a tree, sent reports will be listed as child nodes of the user who sent them. This way, it will be easier to know who has filed what, and only the user who has filed an unverified report can directly see it. Once the PNP Admin interface has verified it, then that report will be displayed publicly.

Images and other media storage

A real-time database only functions on non-media data like text, integers, and location data; therefore, images used within the application interfaces (user image, missing person image) will be stored in Firebase's cloud storage.

3.3.2 User Interface Requirements

User (or Main) Interface

The User use-case diagram in Figure 3.4 illustrates all the possible tasks that a normal user could do within the application. User account creation will be done within the application itself, through Firebase auth's authentication service using a registered email address and password. After the user has registered and confirmed his/her email address, his/her account will be created. Within the proposed application, the user can do a variety of things, one of which is linking to a companion PARGOM by searching him/her within the application's search interface.

When linked up to a companion PARGOM, the User can prompt to ask the

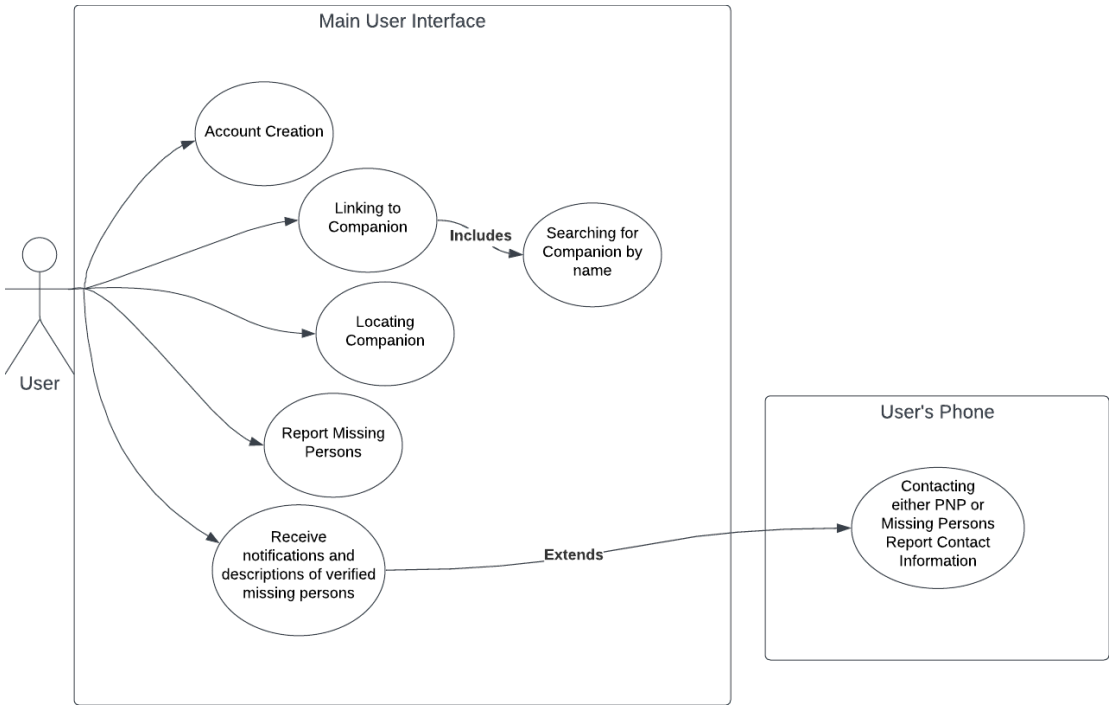

Figure 3.4: Use-Case Diagram for User (Main User Interface)

latest tagged location of the said companion. A main feature within the application, also, is with the reporting of missing persons. This is done through the application by filling out a form with the details required for the missing persons report, which would then be filed towards the most logically sound police-station. Users can also receive notifications with regards to the descriptions and features of any missing persons report that have been verified by the PNP.

Companion App Interface

The use-case diagram for PARGOMs and the companion interface of the application is seen in Figure 3.5. In principle, PARGOM user accounts are a miniaturized version of the base application. These user accounts are intended for users who are children, the elderly, or anyone else that can be considered as a person at risk of going missing. As such, every PARGOM, just like the User accounts, can also register the same way using Firebase auth authentication services.

Being a companion account, as well, allows PARGOM account users to receive

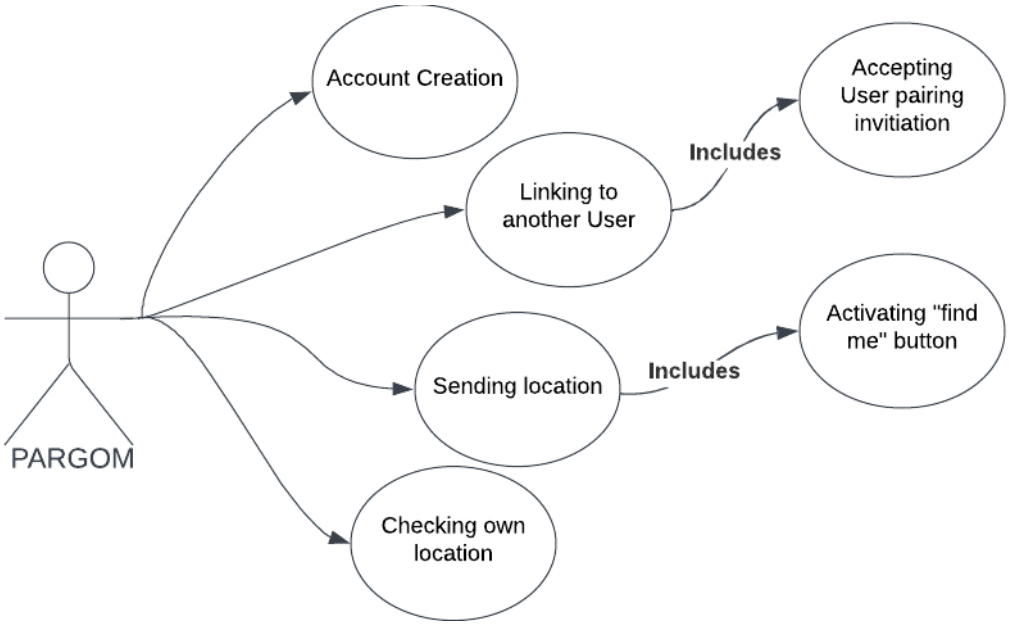

Figure 3.5: Use-Case Diagram for PARGOMs (Companion Interface)

invitations to becoming a companion of any Users they deem to be trustworthy of their location, for example, a child companion/PARGOM account holder to his/her User account holder parent. PARGOM accounts can also activate a “find me” button in which he/she can share his/her location directly with the User account it is connected to. Lastly, PARGOM accounts users can also check their own location within the app.

PNP Admin App Interface

The PNP and PNP Admin interface use-case diagram is shown in Figure 3.6. The PNP admin accounts are created in a different manner compared to the previously stated accounts. First, PNP admin accounts cannot be created through the PNP admin interface (e.g., a ”Register” option) in order to control and limit the number of admin users within a police station to only one, who can only handle missing persons reports within their jurisdiction.

Local police stations can request PNP admin accounts from the developers in order for it to be recognized as an ”admin” account rather than any main user or PARGOM companion account. Once a PNP admin has logged in to the PNP

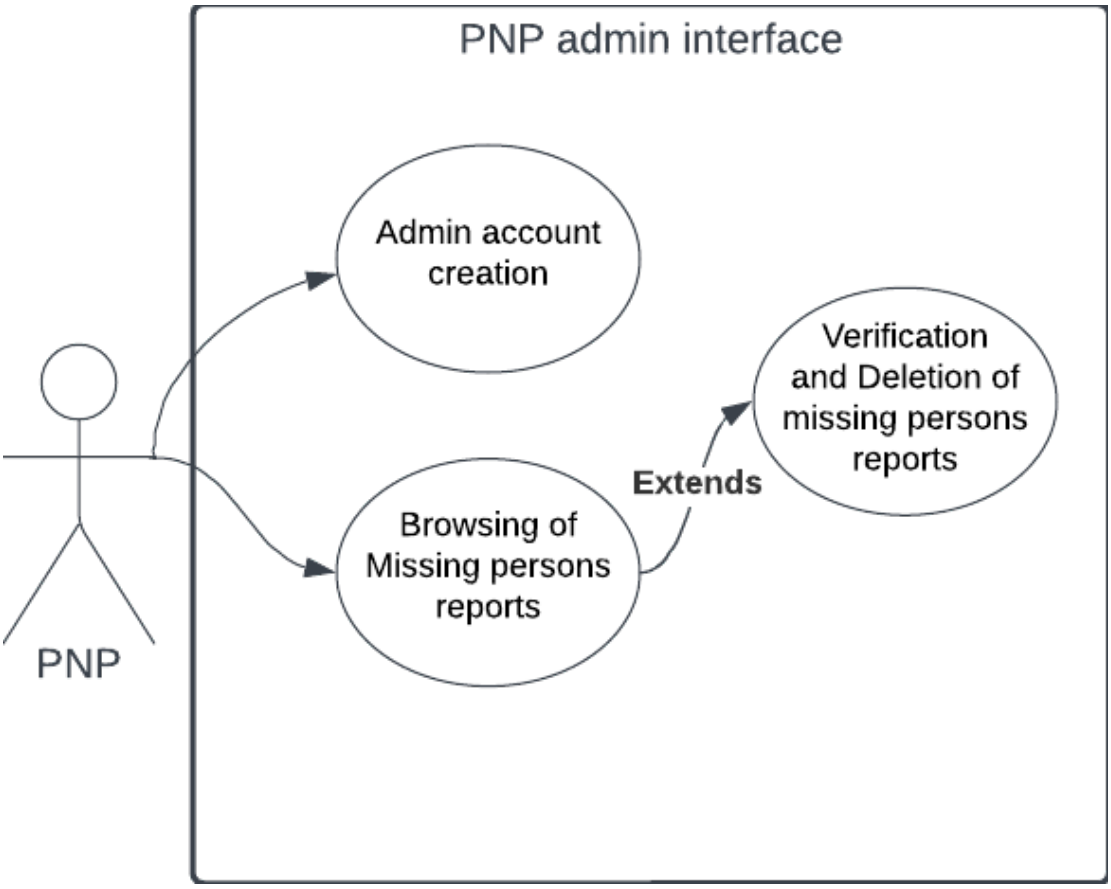

Figure 3.6: Use-Case Diagram for PNP (PNP Admin Interface)

admin interface, he or she, as an official, can then browse and either verify or delete missing person reports filed within their jurisdiction.

3.3.3 Functional Requirements

User Registration

Main and Companion application Users should be able to launch their respective application interfaces and register through the log-in and registration page in-app. Once verified and registered, both user types can then utilize and log in into the application.

PNP Admin Account Creation

PNP Admin accounts users should be able to contact the developers using the information displayed about logging in through the PNP Admin interface, in order for each PNP station to have a PNP Admin Account.

Reporting and Receiving Updates

General users should be able to fill out and send the MP case report form via the main app interface to the PNP station that has jurisdiction of the area where the reported person went missing, and receive updates via the main app with regards to the status of the report.

As seen in Figure 3.7 sequence diagram, reporting and receiving updates is very straightforward: the user (reportee) needs to fill out the MP report form, it's received by the PNP through their PNP Admin App interface, and if the person is indeed missing (and not yet reported, or already found, or dead), then the PNP can verify the report. If any updates are made, such as if the MP is found, it will be reflected in the user's app soon after.

It's also an important requirement that any reports made by users (reportee) are sent to the PNP Admin Account of the MP case location's jurisdiction (and not to all PNP Admin accounts throughout the country). As seen in Figure 3.7 , many users should be able to submit MP case reports of a specific location to the PNP through the application, but they will only be routed to the PNP Admin Account registered for said location.

Receive, Update, and Verify Reports

The PNP Admin should be able to receive reports from users through the PNP Admin app. PNP Admin should also be able to put updates on the report (i.e. MP case already reported, MP reported is already found, MP reported is already deceased), and also verify the report to state that the MP is categorized as missing. This can also be seen in the sequence diagram in Figure 3.7.

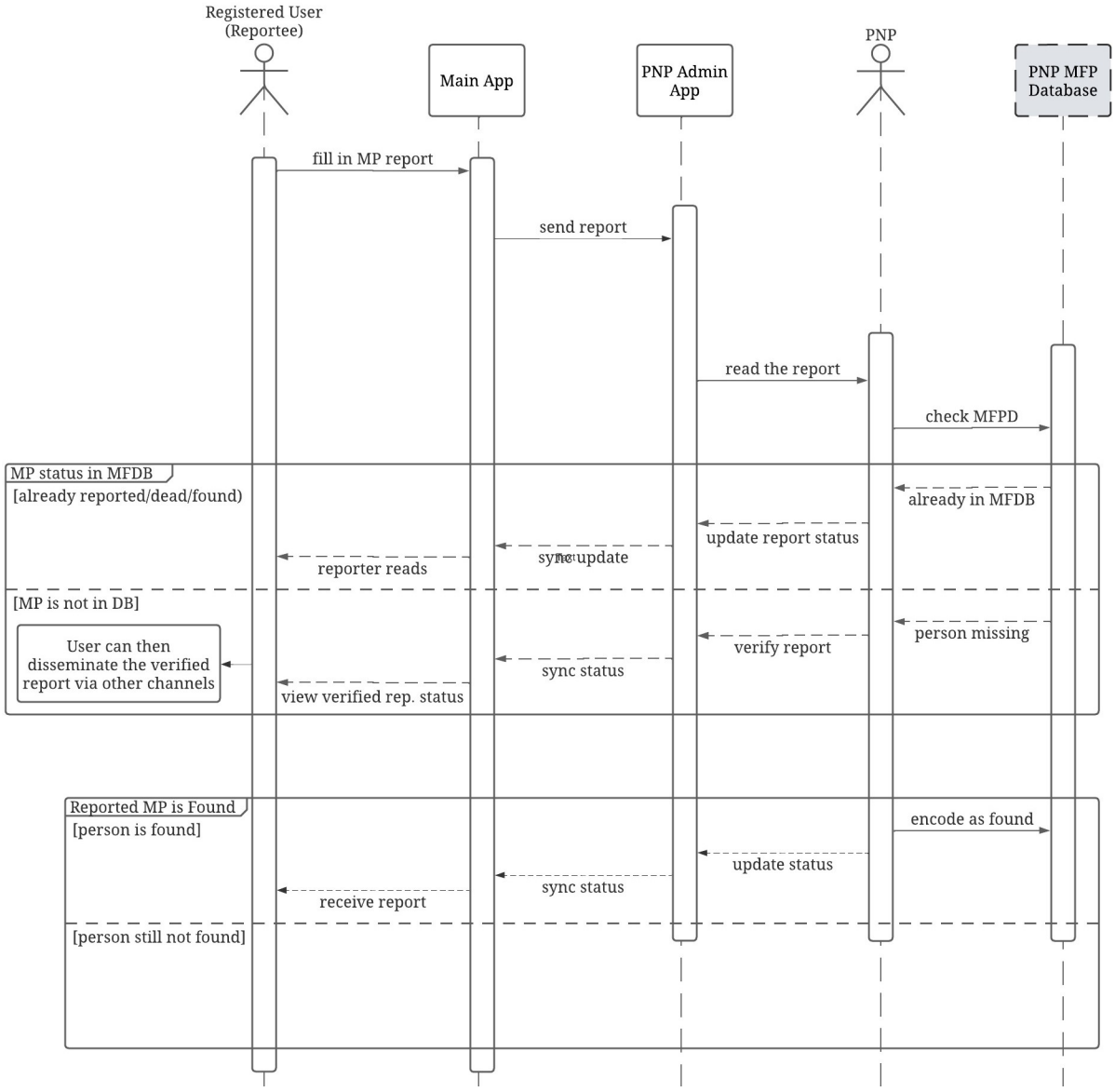

Figure 3.7: Sequence Diagram for Reporting, Updating, and Verification of MP cases

Moreover, as mentioned in the previous requirement and as seen in Figure 3.8 relationship diagram, the PNP Admin Account of a certain location should only receive (and be able to update/verify) MP reports from MP cases within the PNP station's jurisdiction.

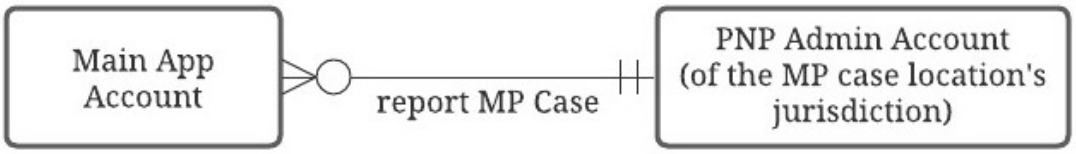

Figure 3.8: Entity Relationship Diagram of Main App Account Reporting to PNP Admin Account

Accessing PNP-Verified Reports and receiving Location-based notification for PNP-verified MP cases

Users in a certain radius of the PNP-verified MP case receive notification about an MP case in their area. Users can tap on the notification to view details of the MP and the contact number of the reportee or PNP station in charge.

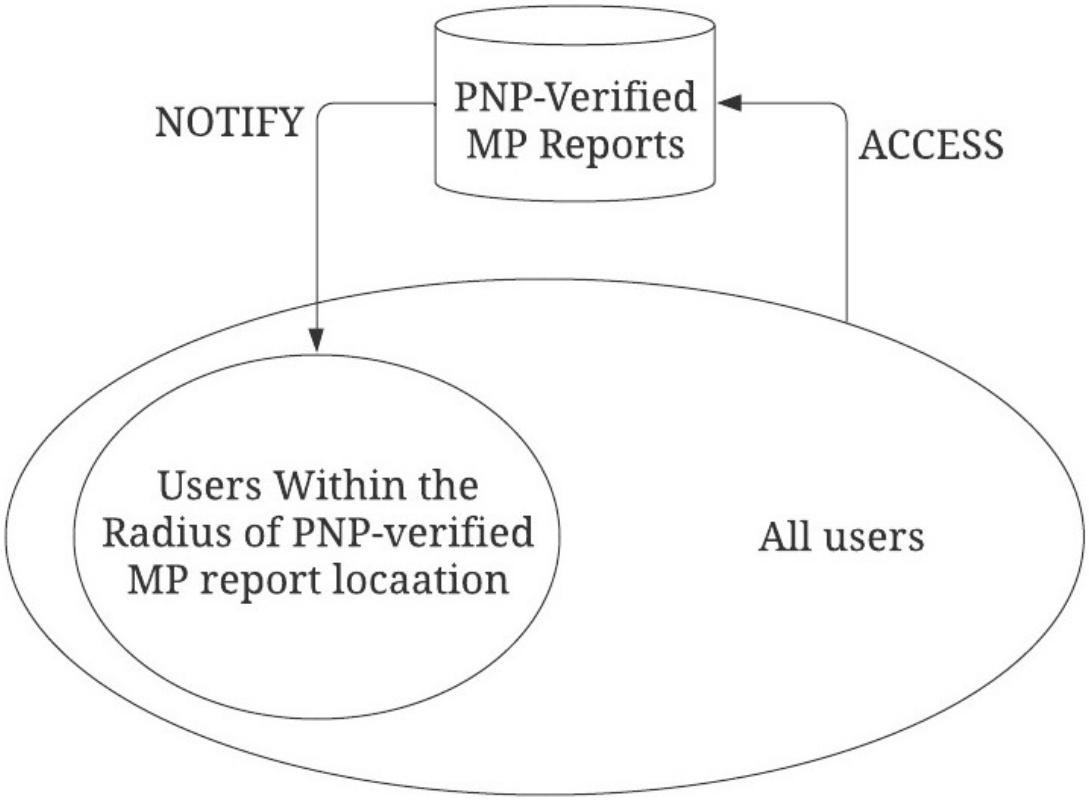

Figure 3.9: Diagram for PNP-Verified Reports Access and Notification System

As seen in Figure 3.9, although only users in the radius of the PNP-verified MP re-

port location will be notified, all users throughout the country will be able to view the PNP-Verified reports all over the country in order to leverage the community-based approach and cover a wider range for searching MPs.

For ease of use, the main app will be designed wherein whenever a user checks on PNP-Verified MP Reports, the first results should be of MP cases within or near their area, and should be provided the option to filter cases by location.

User-companion binding

Users and PARGOMs should be able to bind their accounts via the main app and companion app so that users can keep track of the companion's location.

As seen in 3.10 sequence diagram, the binding process is done through a simple "request-and-accept" system. The main user should be the one to initiate the binding process by entering the unique username of the companion account, and send the request. Afterwhich, the companion user should be able to either decline or accept the request for binding. Once accepted, the main and companion accounts will be binded. The purpose for this approach is to ensure that not any user could just bind their account to a companion.

It's also important that the system supports multiple account binding. As seen in Figure 3.11 relationship diagram, a companion account should always be bound to a main account for one of the main features (Find Companion) to work. Multiple main accounts can bind to multiple companion accounts (i.e. father and mother, binding their main accounts to their children's companion accounts).

User App - Find Companion Feature

Users connected to a companion app (companion user) should be able to check the most recent location of the companion they are linked to. This is done within the main user interface. Companion users, upon connecting to a main User for the first time should be informed that the User they are linked to has access to their most recent location.

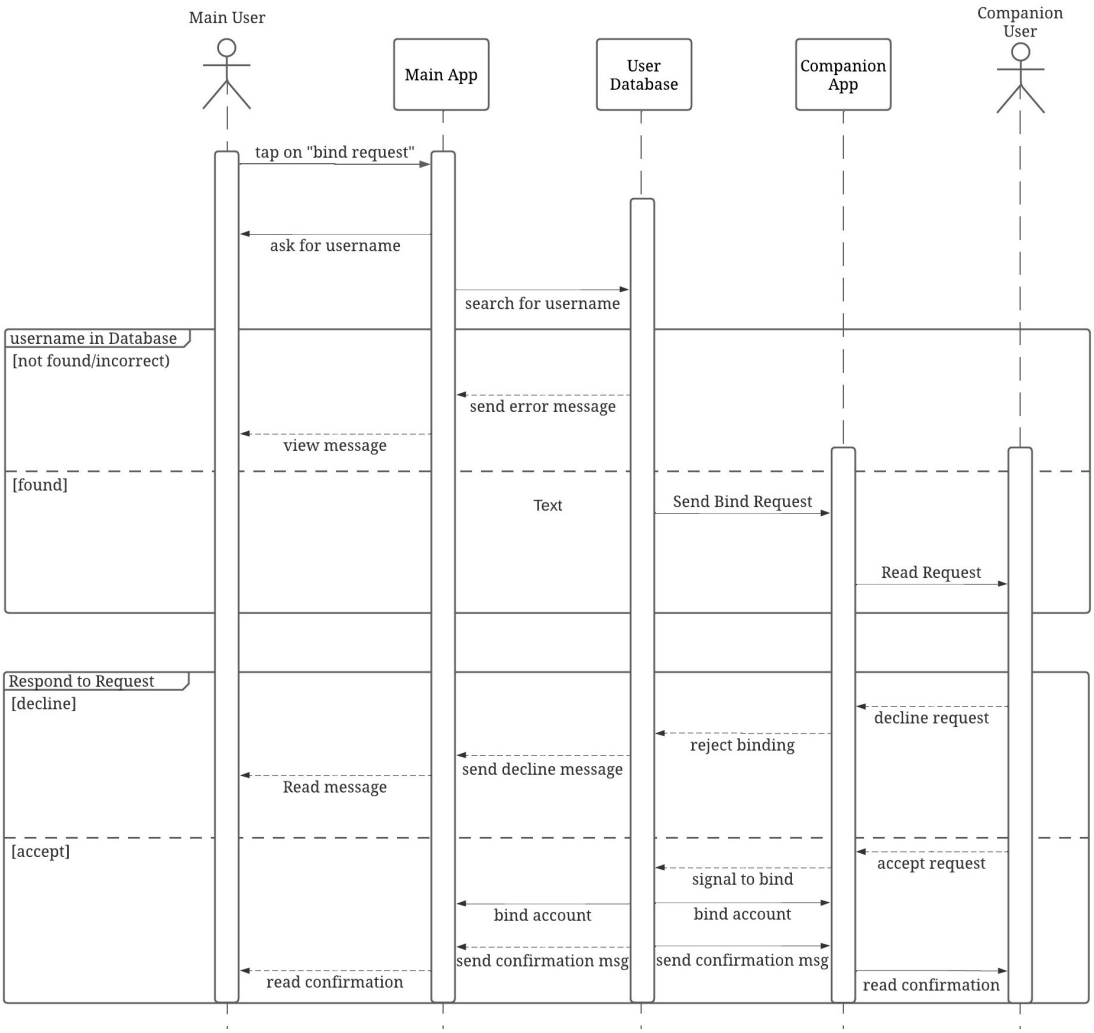

Figure 3.10: Sequence Diagram for Main-Companion Account Binding

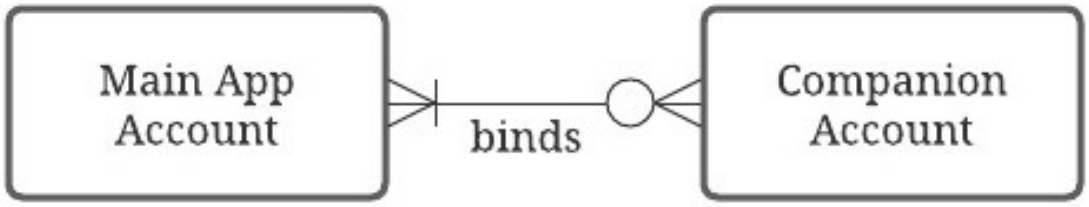

Figure 3.11: Entity Relationship Diagram of Main App Account and Companion Account

Companion App - Find Me

Companion application users should be able to use a “Find Me” button within the application in order to signal the linked User account of the companion’s current

location. The “Find Me” feature should also display the nearest PNP station or help desk, and display their contact information for easy calling.

Chapter 4

System Prototype

4.1 User Interface

The mid-fidelity wireframe of the application's features will be subdivided into three interfaces, namely PNP Admin, User or Main, and Companion.

4.1.1 PNP Admin Interface

The PNP-side interface will focus on the managing and administrative side of the product. Figure 4.1 Login page of the PNP-side app will authenticate the user to enforce access in the reports management system of the missing persons. Input fields are provided for username and password for PNP's credentials. When the PNP Admin user is having troubles with regards to authentication or registration, he/she would need to contact the developers directly, either to get an admin account for their respective PNP outpost, or to reset the password of their said admin accounts.

Figure 4.2 is the management suite for the PNP to access reports. The Reports page will have a scrollable list that will consist of the various details such as MP's name, status of the report, time and date reported, and the location of MP. As for the status, a dropdown menu will be used to categorize each report namely, Received, Already Found, Verify Report, and Reported. For better management,

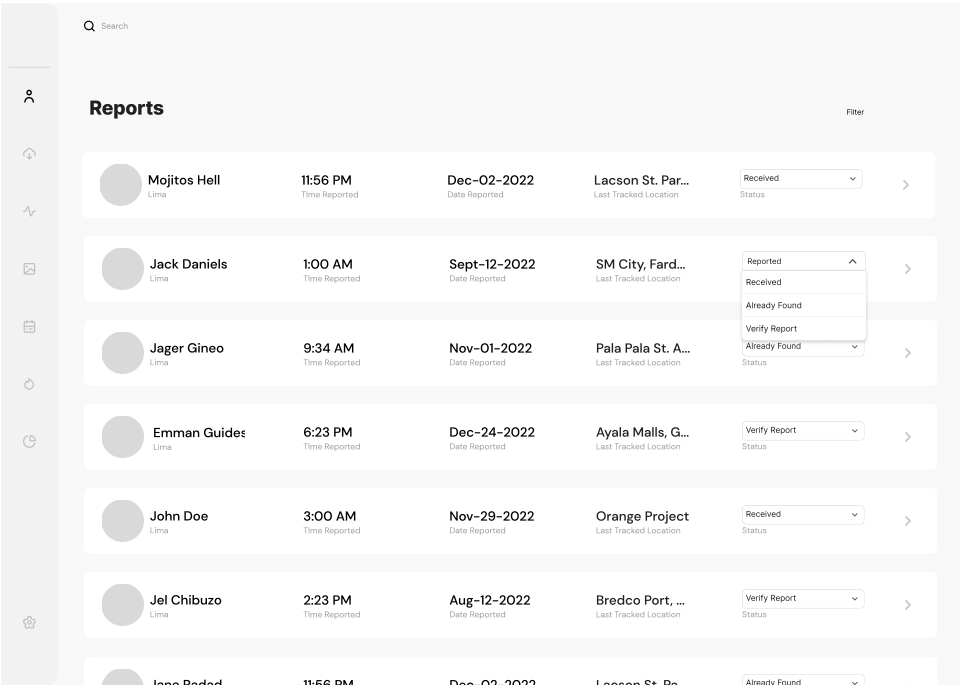

Figure 4.1: Desktop View for Login Page

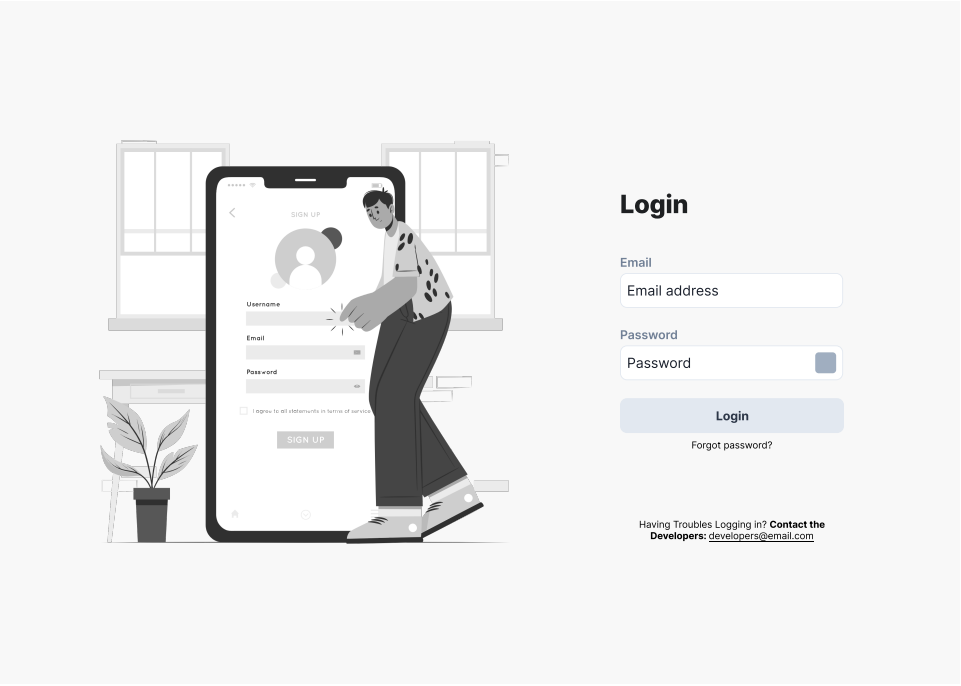

Figure 4.2: Desktop View for Reports

a Filter button located at the top-right will be available to ensure that every report is being noticed and filtered out.

4.1.2 User or Main Interface

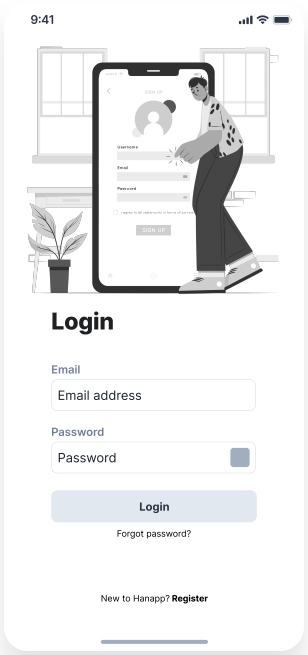

Figure 4.3: Login Page

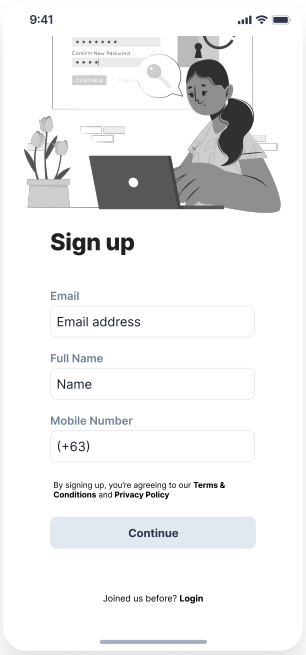

Figure 4.4: Register Page

Authentication is necessary for the user-side. Figures 4.3 and 4.4 are the Login and Register page, respectively. Input fields will be provided for the registration, username and full name, a numeric input field will be observed for the mobile number. To ensure that the user understands the Terms and Conditions and Privacy Policy, a hyperlinked text can be used to redirect them for their perusal. Then a continue button will let the user proceed to the next required details such as reportee's identification for easy reporting of missing persons later on.

Moreover, when user authentication is completed. The login button will let them proceed to the Home page of the application where the user will have two options: Report Missing or Locate Missing persons. A bottom navigation bar will be visible throughout the user's interaction of the app assuring that they can easily

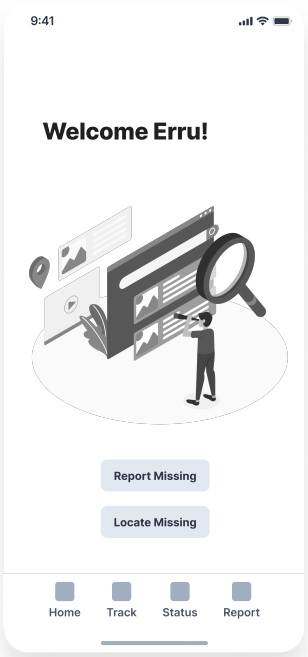

Figure 4.5: Home Page

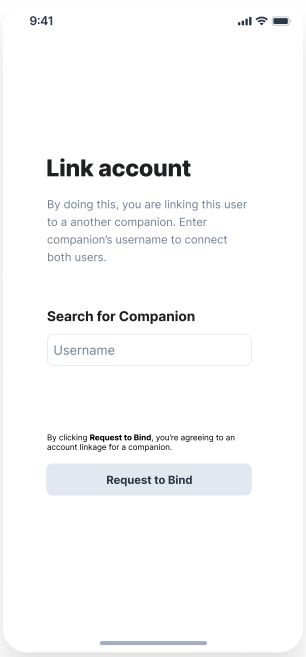

Figure 4.6: Link Account Section

navigate the entire features offered to them. The navigation bar will consist of four (4) sections namely, Home, Track, Status, and lastly, Report. Each of these sections have distinct functionalities.

Importantly, the user-side application will be able to link user's app to a companion app. Figure 4.6 searches unique companion's username to bind both account for easy linkage of current location and for tracking purposes.

Figure 4.7 is the interface of tracking missing persons near the user. A location marker will guide the current or last seen whereabouts of a missing person. Apart from that, the user has the liberty to navigate the map, may it be street view or city-wide view by clicking the "person" icon at the lower right of the screen. Also, users can zoom in and zoom out depending on their preference. Considering, for example, that a 5-km radius will be visible, it will then encircle the last place the MP was seen.

As for the push notifications of incoming verified missing persons, Figure 4.8 will prompt an alert message that will shortly disrupt the app's activity. This way,

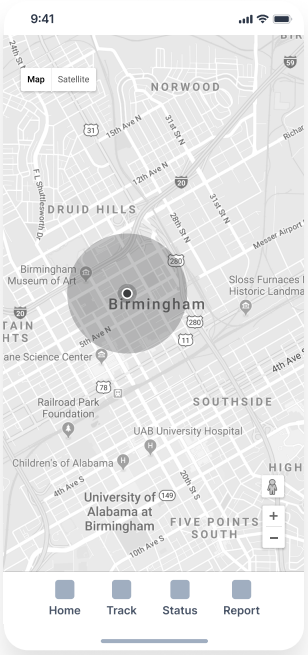

Figure 4.7: Tracking MP Location

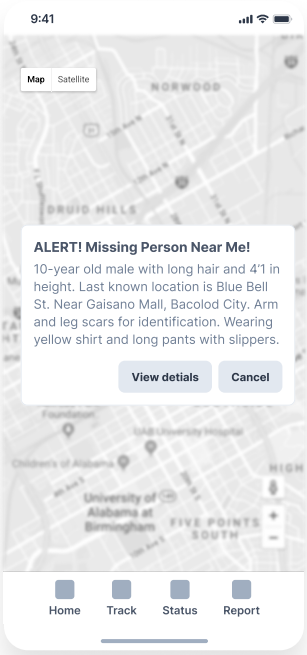

Figure 4.8: Push Notification Alert

the user's attention is engaged and focused on the notification for a short period of time. The message will be a short description of the person in case there are sightings of them around the area. To give the user some leeway, they will have two options whether to view the details of the report or cancel the alert.

One of the features of the app is the consolidating all the missing reports in one view. Basically, this Figure 4.9 interface is a scrollable view for all Missing Persons Near Me feature which also redirects to Figure 4.7 allowing maximization of the map for easy tracking. It would allow easy navigation of all missing reports and have a glimpse of any updates pertaining to the location that the MPs were seen.

Obviously, the Report page is one of the integral parts of this application that would allow users to give necessary information in the event of a person missing. Figure 4.10 streamlines the reporting process done by the PNP and replicates their incident report so that verification is manageable. Input text fields will be used throughout this page. Accompanying it with dropdown menus for categorical information such as Gender, and Blood type. Physical indicators will have mostly text/paragraph input fields for elaborations and additional details that are needed

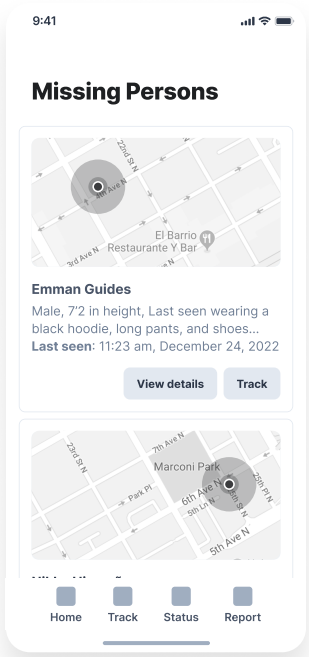

Figure 4.9: Missing Persons Near Me

to disclose such as body marks, tattoos, and the like.

4.1.3 Companion-side

Figure 4.12 visualizes the interface for the "Find me" button. This will be the main feature of the companion-side app in which it will be centered at the home page after the login. In the event that the user is missing, the companion user can click the "Find Me" button to let their paired User know their location, in case of an emergency.

After the "Find Me" button is activated, a map locating the user's real-time location as shown in Figure 4.13 will be shown. Importantly, the companion can further leverage this by clicking nearby police stations within boundaries in order for them to be able to know the nearest Police Station they can possibly go to. Location markers will be visible in the map as long as it's the nearest police station from the user's mark. Thus, companion users can easily contact the provided details of protective service and call or text message as shown in Figure

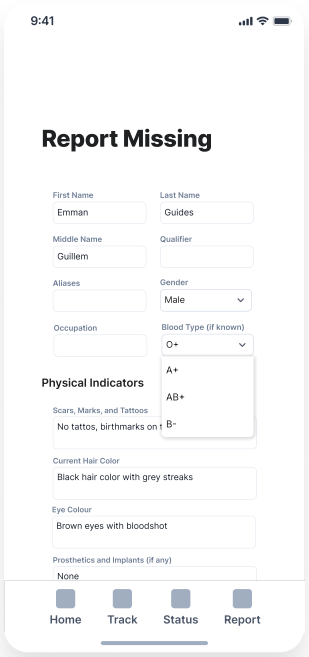

Figure 4.10: Report Missing Section

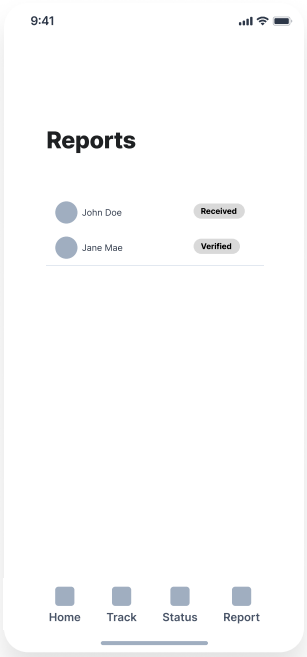

Figure 4.11: Status Reports Section

4.14 Get Help Section.

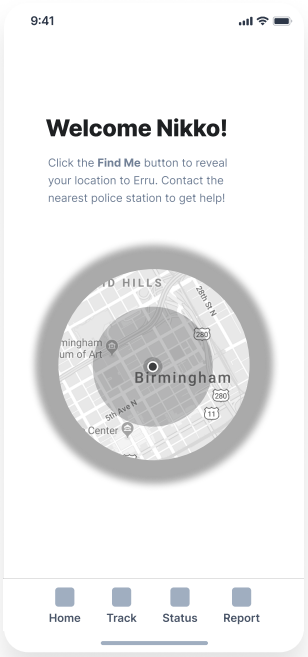

Figure 4.12: Find Me

Figure 4.13: Show Location

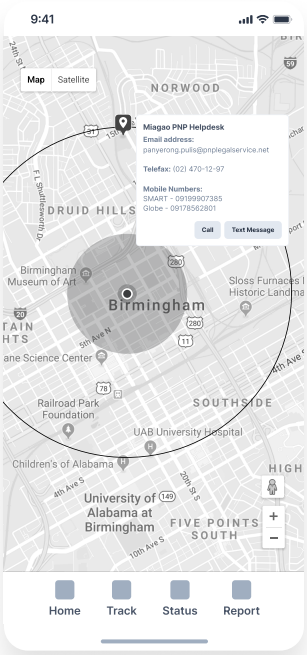

Figure 4.14: Get Help

References

About github. (n.d.). Retrieved from https://github.com/about

Antony, B. (2021). *A study on usage of google maps by travellers in madurai city.* Retrieved from https://www.ijires.org/administrator/components/com_jresearch/files/publications/IJIRES_1779_FINAL.pdf

Buatag, R., Garcia, P. M., & Visto, R. I. (2022, Jul). *Travellert: A location-based alarm application.*

Castro, P., Ishakian, V., Muthusamy, V., & Slominski, A. (2017). Serverless programming (function as a service). In *2017 ieee 37th international conference on distributed computing systems (icdcs)* (pp. 2658–2659).

Desale, H., Tavasalkar, P., Vare, S., & Shintre, R. (2020). Android crime reporter and missing person finder. *Int Res J Eng Technol (IRJET)*, *7*(03), 2395–0056.

Distance matrix api. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/distance-matrix/overview

Flutter: Build apps on any screen. (n.d.). Retrieved from https://flutter.dev/

Gah, S. K., Katsriku, F., & Gyamfi, N. K. (2018). Using gps and google maps for mapping digital postal address (ghanapostgps). *Communications on Applied Electronics*, *7*(13), 15–20.

Garg, D., & Shukla, D. A. (2013). geo alert-a location based alarm system using gps in android. *International Journal of Multidisciplinary in Cryptology and Information Security*, *2*(3).

Geocoding api. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/geocoding/overview

Geofencing api. (n.d.). Google. Retrieved from https://developers.google.com/location-context/geofencing

Google Developers. (n.d.). *Google maps platform.* https://developers

.google.com/maps/faq#:~:text=The%20Google%20Maps%20Platform%20is,Maps%20JavaScript%20API. Google.

GPS.gov. (n.d.). *The global positioning system.* Retrieved from https://www.gps.gov/systems/gps/

Griffin, Miller, Hoppe, Rebideaux, & Hammack. (2007). A preliminary examination of amber alert's effects. *Criminal justice policy review*, *18*(4), 378–394.

Hannula, T. (2021). Unity mobile application with a serverless firebase backend.

International Commission on Missing Persons. (2022). *Who are the missing?* Retrieved from https://www.icmp.int/the-missing/who-are-the-missing/

An introduction to github. (2020, Jun). Retrieved from https://digital.gov/resources/an-introduction-github/

Kanfade, M. M., Ambade, S. D., & Bhagat, A. P. (2018). Location based notification system. In *2018 international conference on research in intelligent and computing in engineering (rice)* (pp. 1–6).

Khawas, C., & Shah, P. (2018). Application of firebase in android app development-a study. *International Journal of Computer Applications*, *179*(46), 49–53.

Kilic, B., & Gülgen, F. (2020). Accuracy and similarity aspects in online geocoding services: A comparative evaluation for google and bing maps. *International Journal of Engineering and Geosciences*, *5*(2), 109–119.

Lookingbill, A. (2019). *Google maps 101: how we map the world.* Google.

Madarang, C. R. S. (2022, Jul). *Public urged to be skeptical of unverified posts on cases of missing persons.* Retrieved from https://interaksyon.philstar.com/trends-spotlights/2022/07/07/221520/public-urged-to-be-skeptical-of-unverified-posts-on-cases-of-missing-persons/

Maps embed api. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/embed/get-started

Mehta, H., Kanani, P., & Lande, P. (2019). Google maps. *Int. J. Comput. Appl*, *178*(8), 41–46.

Microsoft. (2021, Nov). *Documentation for visual studio code.* Author. Retrieved from https://code.visualstudio.com/docs

Murray, Anderson, Clark, & Hanzlick. (2018). The history and use of the national missing and unidentified persons system (namus) in the identification of unknown persons. In *New perspectives in forensic human skeletal identification* (pp. 115–126). Elsevier.

Mutisya, W. E. (2017). *Mobile phone application for reporting and tracking missing persons in kenya* (Unpublished doctoral dissertation). Strathmore University.

National Police Commission. (2016). *Guidelines on the recording, monitoring and investigation of missing and found persons, and identification of human remains.* Camp Crame, Quezon City Philippines. Retrieved from https://www.didm.pnp.gov.ph/index.php/2-uncategorised/55-4th-police-expert-dispatch

Neubauer, N., Daum, C., Miguel-Cruz, A., & Liu, L. (2021). Mobile alert app to engage community volunteers to help locate missing persons with dementia. *Plos one*, *16*(7), e0254952.

Orion Support Incorporated. (2021, Jun). *Philippines Missing Persons Investigation.* Retrieved from https://www.osi.com.ph/commercial-investigations/philippines-missing-person-investigations/

Philippine Statistics Authority. (2021). *Number of deaths, missing persons and directly affected persons attributed to disasters per 100,000 population.* Retrieved from https://openstat.psa.gov.ph/PXWeb/pxweb/en/DB/DB__3I__G13/0013I3D1311.px/table/tableViewLayout1/?rxid=8fcf1969-b128-48d1-ab99-45bbd52dd512

Places api. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/places/web-service/overview

ProductPlan. (2022). *What Is Feature Driven Development (FDD)? — Definition.* Retrieved from https://www.productplan.com/glossary/feature-driven-development/

Pulta, B. (2021). *Missing, exploited children reports double in 2021: Doj.* Retrieved from https://www.pna.gov.ph/articles/1163963

Shin, H., Park, T., Kang, S., Lee, B., Song, J., Chon, Y., & Cha, H. (2014). Cosmic: Designing a mobile crowd-sourced collaborative application to find a missing child in situ. In *Proceedings of the 16th international conference on human-computer interaction with mobile devices & services* (pp. 389–398).

Sunehra, D., Priya, P. L., & Bano, A. (2016). Children location monitoring on google maps using gps and gsm technologies. In *2016 ieee 6th international conference on advanced computing (iacc)* (pp. 711–715).

Yarrabothu, R. S., & Thota, B. (2015). Abhaya: An android app for the safety of women. In *2015 annual ieee india conference (indicon)* (pp. 1–4).