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

May 5, 2023

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, case reporting and verification, community-based search, location tracking

Contents

List of Figures

Chapter 1

Introduction

1.1 Overview of the Current State of Technology

As defined by the International Commission on missing person, 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 missing persons globally are harder to discern. 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, there have been 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 and/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 process can be cumbersome and counter-intuitive, particularly in cases wherein the missing person is a child or a victim of an accident in which the case recording, investigation, and monitoring need to begin immediately (National Police Commission, 2016).

In response to this, the researchers aim to develop an application suite that leverages current technology to streamline the processes required in the filing, managing, and verifying of missing persons cases. The application suite is modeled after the AMBER alert system, such that it uses location and map services in both pushing area-wide alerts or notifications and also in tagging the missing person's last seen location on a map. As such, the application suite enables community cooperation in the search for missing persons through the 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: namely 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 persons cases has troubled not only the community but also authorities to develop effective measures and guidelines to quickly act on missing persons cases.

The dire situation of these missing persons 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 conflict in the community.

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 and required the user to fill out numerous forms (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, there is a need for a system that leverages the use of existing technology to streamline and accelerate the reporting and verification process, and notify users of PNP-verified missing persons cases in their area. The creation of the missing persons reports and alerts application suite, "HanApp", addresses the aforementioned issues by enabling users to file and receive updates on missing persons cases, get alerts and view PNP-verified missing persons cases, and also allowing the PNP to receive, manage, and verify missing persons reports online.

1.3 Research Objectives

1.3.1 General Objective

The general objective of this study is to develop an application suite with mobile and desktop interfaces for users who want to report and get updates on missing persons cases, users who want to be altered by and view verified missing persons cases in their area, and the PNP in order to expedite and simplify the filing, managing, verifying, disseminating, and resolving of missing persons cases. The application will be called "HanApp", a portmanteau of "Hanap" (Tagalog for "find") and App (for "application").

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 which securely stores and manages User and PNP accounts, login authorizations, and reports without the need for a server infrastructure or dependence on local data storage, allowing the application suite to be used in any operating-system compatible device.

3. To integrate geolocation services that allow for report routing so that reports are received and managed by the corresponding PNP account of the case's jurisdiction.

4. To implement location-based notifications so that users can receive alerts and view information about missing persons near their current location so they can assist in the search, thus promoting a community-based approach in resolving missing persons cases.

1.4 Definition of Terms and Acronyms

Child - refers to any person under the age of 18.

Absent Person - according to the PNP guidelines, absent persons are defined as those who are not in their domicile or place where they are supposed to be present in less than 24 hours, and whose families/relatives/significant others have no clue as to their whereabouts but there is no apparent risk and do not require police investigation.

Absent/Missing Person (A/MP) - in accordance with the PNP guidelines and the affixed checklist when receiving and processing missing person cases, the term (A/MP) can refer to the reported person whose whereabouts is unknown and whose classification as Absent or Missing is not yet finalized.

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 suite**", refers to the set of applications and its various interfaces developed in this study. The interfaces include:

- **Main (or User) App** - refers to the main application that has the features: reporting and getting updates on missing persons cases, receive alerts/notifications and view PNP-verified missing persons cases near their current location

- **PNP Admin App** - also referred to as "**PNP Application/App**", refers to the application interface that is accessed by the PNP in which they receive, manage, verify, and provide updates on received missing persons reports.

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

Missing person (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 suite.

PNP-verified report - refers to the report filed through the Main/User App that has been verified as complete and legitimate by the PNP precinct personnel through the PNP App.

Reportee - refers to the user filing the missing person CASE report via the main application, and is able to view the status of their reports through the main user application.

1.5 Scope and Limitations of the Research

HanApp primarily aims to streamline and hasten the filing and verification of missing persons reports to the PNP and create an intuitive platform for the alerting and dissemination of PNP-verified missing persons reports.

HanApp covers the following classifications of missing persons in accordance to the PNP Guidelines: absent person that has not been located 24 hours from their perceived disappearance, missing children, missing victim of natural calamities and human-induced disasters/accidents, and missing persons believed to be a victim of violence and crimes.

Furthermore, HanApp employs a serverless approach in its accounts and data management through Firebase, as well as online services such as Google Maps and other location based services, all of which rely on stable internet connectivity, be it through mobile data or Wi-Fi. As a result, HanApp is an exclusively online application.

Finally, in order to facilitate report filing and management, as well as alerting and disseminating PNP-verified missing persons reports, a serverless database powered by Firebase is used to store the missing persons reports. It should be noted that this is a separate and different database from the PNP's own "Missing and Found Persons" database (MFPD) and is exclusive to the application suite.

1.5.1 User and Information Access

The application will only have three interfaces which will cater to its two primary users — the general user, and the PNP personnel/helpdesk in charge of missing persons cases. Since this is an online-only application suite with a mobile interface for the main users and a desktop interface for the PNP, and the application's scope is limited to the Philippines, 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, or assist with search efforts for missing persons. They will only have access to information regarding their registered account, the updates provided by PNP on the report they filed, and the information included in the PNP-verified missing persons in their area.
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 or their devices.

Moreover, the application's administrators and developers will only have access to the serverless database containing the submitted reports and the (PNP and

User) account names and emails at most. As such, user passwords and user's current 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 missing persons reports, and will not be accessed by the application, the application's users, nor its administrators.

1.6 Significance of the Research

The research's application suite and its framework aim to consolidate and streamline the process of filing, managing, 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 to affected individuals whose loved ones have gone missing by allowing them to report and ultimately locate the whereabouts of these missing individuals with ease.

Moreover, the application could serve as a great tool to the PNP by making it faster and more convenient to receive, manage, and verify missing persons reports, thus allowing them to act faster on missing persons cases while simultaneously reducing the propagation of unverified reports. In addition, the data that will be generated from the application suite's usage by the public and the PNP can be used as reference regarding the statistics relating to missing persons in the country.

The focal point of this research paves the way for a systematic and technological breakthrough here in the Philippines through the development of an application suite that could potentially assist in resolving missing person cases by hastening the reporting and dissemination process.

Lastly, the novelty of this study could potentially provide a baseline framework

for similar studies impacting the creation of not only a more robust system for reporting missing persons cases in the country, but also the modernization of crime/incident reporting to the PNP through the use of online software rather than exclusively-in-person reporting.

Chapter 2

Review of Related Literature

2.1 Missing Persons Reporting and Finder 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). As the title suggests, the study proposed an application that allows users in India to directly report not only missing persons cases but also other crimes to the authorities through the app, with the goal of hastening the reporting process and, subsequently, solving the crime.

Moreover, their application features a "panic" feature to quickly record a crime and an "alert" feature to immediately inform authorities of wanted criminal sightings. The proposed app also allowed the user to get updates regarding their submitted case, and allowed authorities to track cases more conveniently. This aspect of the application is directly in line with the goals of the study and is proof of the viability and significance of the HanApp framework.

One key difference with this application compared to HanApp, 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 HanApp's framework expanded the scope by enabling and enforcing the dissemination of verified infor-

mation with regards to missing persons cases.

2.2 Mobile Applications for Alerting the Community Regarding Missing Persons

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 missing persons (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 missing persons. 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 alerting and disseminating of information regarding missing persons in order to encourage and widen the scope of community participation in resolving missing persons cases.

2.3 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 application suite, HanApp, 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 helped in the decision-making of the developers during the development of the application suite.

2.3.1 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, was a promising choice for the chosen serverless framework for HanApp. Knowing the benefits and how Firebase was implemented on mobile applications was crucial in the development of HanApp.

2.4 Location Services in Missing Persons Applications

2.4.1 Global Positioning System (GPS)

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).

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.

2.4.2 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.

2.4.3 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 provide proof of feasibility in using GPS, Google Maps, and similar technologies on how to approach the development of the app's location-based notification system for alerting users of PNP-verified MP cases near them.

2.4.4 Simplified User Experience and Interface Design for Crisis and Emergency-related Applications

User experience (UX) design is critical in developing effective methods for locating missing people. When someone goes missing, their loved ones may go through a painful and difficult situation. UX design may assist in the development of

tools and systems that make the search process simpler, more efficient, and less overwhelming. Hence, in addition to psychological and behavioral behaviors, the emotional experience of criminal behavior must be understood. There is an occurrence of the circumplex of emotions, which presents emotions in a circular order based on the aspects of arousal/non-arousal and pleasure/displeasure (Hunt, 2020; Suchana, Alam, Meem, Turjo, & Khan, 2021). Following this premise, the app will relatively be designed in a calm approach in order to mitigate individuals' emotional experiences during distress. The emotional needs of the users are the top-most priority and need to be taken into account for better user experience.

Designing intuitive and simple-to-use search systems is an important part of UX design in missing persons instances. Everyone should be able to use the platform, including those who are not tech-savvy or have low resources. Illustrations are critical in increasing the user experience of an app (Suchana et al., 2021). They can be used to convey important information, improve the app's aesthetic appeal, and give a feeling of consistency and identity. With that said, the outlined and rounded features of the material design hinges on the notion to help customers through the search process, thus, the design is basic, with clear directions and user-friendly navigation. Also, the color scheme follows a high-contrast theme that helps critical information stand out and be more easily identified. Indigo, a borderline shade of blue and purple, exudes a personality that is calm, trustworthy, and approachable in times of crisis which often results in reliability (Babich, 2017). Apart from that, this also caters users who are visually-challenged, especially color-blind users, to interact with the app without hassle. Deuteranomaly and Protanomaly are the common types of colorblindedness that impedes most people (Babich, 2017; National Eye Institute, 2019). Both of them make an individual unable to tell the difference between red and green, hence indigo was utilized to recuperate their needs.

Furthermore, the app also focuses on the usability framework that leverages user experiences. Since the app focuses on a distress manner, the application expects to perform smoothly without consuming excessive phone resources and the content relevancy is determined by the app's goal and the information's proximity to the time and location of the hazard event. Tan, et.al. (2020) established the human-computer interaction frameworks that best suit in times of crisis which can be advantageous for user retention and usability. As a result, another crucial concern in missing persons UX design is ensuring that the platform provides useful and

up-to-date information (Tan et al., 2020). This information comprises the missing person's physical description, last known location, and any identifying features. It also contains information about the search activities, such as progress, fresh leads, and how individuals can help with the search.

Chapter 3

Research Methodology

3.1 Research Activities

3.1.1 Development Framework

The application suite 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 reporting and verification features are required before developing the feature that alerts and lets users to view PNP-verified cases near the user's location, and so on).

Due to this reason, the researchers have adopted 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 includes testing after each feature is built and integrated, and the system prototype was released only when all features have been built, integrated, and tested.

As seen in Figure 3.1, HanApp's system was developed by first designing the entire model, listing all the features, planning the order by which to develop the features so that they can be integrated, and then building and integrating the features in said order.

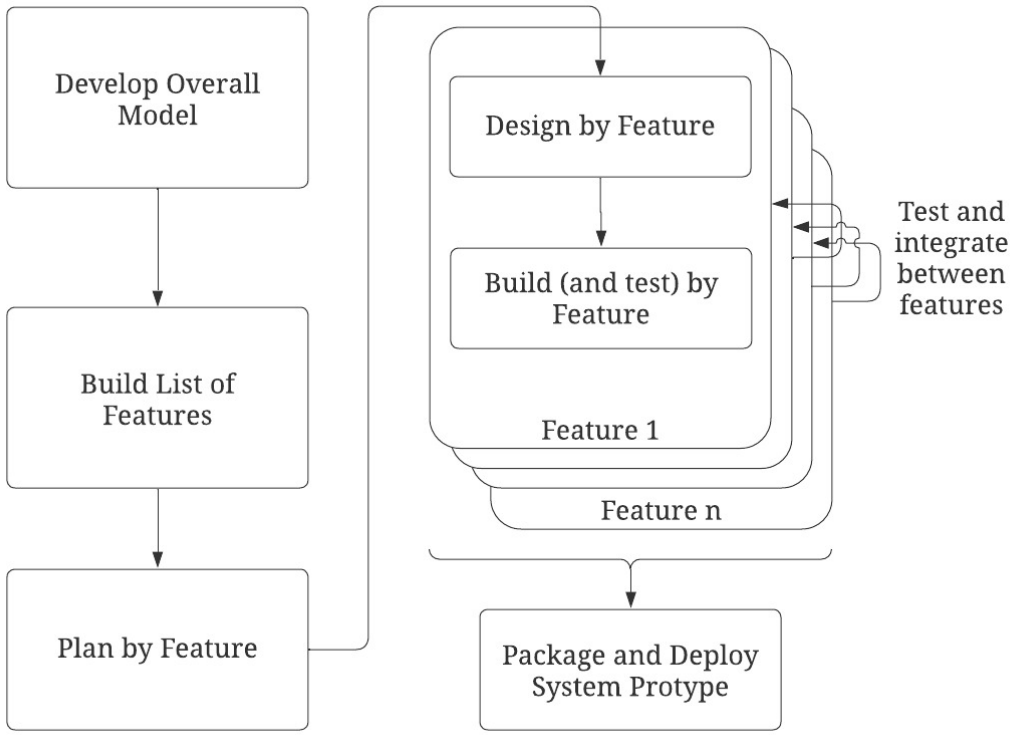


Figure 3.1: Modified Feature-Driven Development (FDD) Framework

3.1.2 Design, Building, Testing and Integration

Overall Model

A high-level overall model or framework of what the application suite should be able to do based on general use-cases was first developed during the study's proposal stage. It was there where the list of general features, to meet the general user requirements, were made. After which, the order of which each feature was to be designed, developed, tested, and integrated. This step ensured that each feature developed had their prerequisites met, facilitating the ease in integrating each of the application suite's features into an overall cohesive system.

Design and Development

From the identified features, a rough design of how each feature would look like and how they integrate with each other was made. After which, following the FDD approach, each feature was designed and consequently developed, with the

frontend (UI/UX) and backend (functions, data handling) codes of each feature being developed simultaneously.

Testing and Integration

During and after the development of each feature, the feature is user-tested in order to ensure that the features work as intended and any bugs are immediately fixed. For features that rely on previous features to work (i.e. user account creation must work before user login), extensive user testing was done to confirm that the integration between each feature worked as intended.

Due to both time and technical constraints, testing was done solely by user testing rather than written and automated tests. Once the current feature and its prerequisite features have been tested, the development of the next feature would commence.

Implementation

The final prototype is the fully functional application suite (including all the user interfaces) that meets the users' requirements and passes all the tests during development.

System Prototype

The final product of the study, the HanApp main user app and the PNP app, is made available to the intended users. In this phase, routine maintenance and regular performance monitoring, particularly in the backend (traffic, database health, storage usage, API usage), is required to keep the application running smoothly. Post-release support would also include bug reporting and fixes, as well as backend and frontend optimizations to enhance the application's efficiency and availability.

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.).

Github was utilized in the project for storing the source code of the application suite and for the source version control during development, as well as for the repository for the LaTeX files for the study itself.

Visual Studio Code

Visual Studio Code (VS 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).

Visual Studio Code and Android Studio were used simultaneously as the primary source code editor, with Android Studio's Android Emulator being used to run the debug builds of the application suite.

Android Studio

Android Studio is the official Integrated Development Environment (IDE) for developing Android mobile apps. It is based on the IntelliJ IDEA development tools and code editor. When compared to other IDEs, Android Studio is considered hefty; however, this is expected considering that it incorporates various integrations and add-ons like a flexible Gradle-based build system, built-in emu-

lators, code templates, extensive testing tools, and many others to guarantee that development is as interactive and fluid as possible (*Meet Android Studio*, n.d.).

Android Studio comes with an Android Emulator which was the primary tool used for running and testing the application interfaces, as well as in debugging.

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 have aimed to deploy the proposed application on both the mobile (Android) platform for the main users, and on the Web (desktop) platform for the side of PNP, Flutter is an outstanding choice out of all the available frameworks and languages.

Flutter (and the Dart programming language) was utilized in the development of all HanApp's interfaces. Flutter packages were also imported for ease of feature development.

Google Firebase

As Firebase has a variety of products and services that it provides, such as 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 Firebase is the sole and primary backend of the application suite. User authentication, database and data storage, and usage monitoring were all done under Firebase. Since Firebase is a serverless framework, utilizing Firebase as the application suite's backend allowed for development and testing without the use of a dedicated server infrastructure.

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. Apart from the Android Studio's Android Emulator, physical Android phones were utilized to test the debug and release builds of the application suite's main user mobile interface.

Laptop

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

3.2.3 Packages and Application Programming Interfaces (APIs)

Packages

Software packages are a group of software programs that can be downloaded as a bundle of related products and used in the development of software. They provide functionalities that are editable or customizable, in order to adjust to the specific requirements of organizations or developers that uses them (Jadhav & Sonar, 2009).

Packages in Flutter can be reviewed and downloaded from pub.dev, the official package archive for Flutter and Dart applications, which is also supported by Google (*pub.dev*, n.d.). Throughout the development of the applications, multiple verified packages from pub.dev have been used and are now essential to the functionality of the applications.

Firebase Packages. Multiple Firebase packages from pub.dev have been utilized in the development of the application to better integrate the serverless backend service (Firebase) onto the applications' interface and services. These packages are firebase_core, the overall prerequisite and helper package for all Firebase services in the Flutter application (*Firebase Core: Flutter Package*, 2023), firebase_auth, the package utilized in order to facilitate the registration, verification, log-in, and authentication persistence of users of the applications (*Firebase Auth: Flutter Package*, 2023), and firebase_storage and firebase_database for the cloud storage needed by the images utilized by the applications, and the real-time database (RTDB) used to save multiple data needed by the applications (*Firebase Auth: Flutter Package*, 2023; *Firebase Storage: Flutter Package*, 2023).

Location Packages. Packages were also needed in order to better facilitate and integrate maps and location services on the applications. These packages are google_maps_flutter, google_maps_flutter_web, and location. The aforementioned google maps packages are used on the mobile application and the web applications respectively, for they are needed in order to display and better blend the user experience for using google maps services on the applications' targeted platforms. Location package, on the other hand, was used in order to ask for user permission before the application requests their current device's location.

Data Persistence Package. Data persistence between views and states of the application is crucial in order to properly pass on volatile data from one page or interface, onto another within the application. For this purpose, the package shared_preferences was used. This package allowed the developers to save data within the application itself or from the databases into a key-value pair in the platform-specific persistent storage for simple data like Strings and Boolean values, among other types (*Shared Preferences: Flutter Package*, 2023).

Application Programming Interfaces (APIs)

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.).

Maps SDK for Android API. Maps SDK for Android is an API from the Google Cloud suite that adds maps functionality to Android apps and even on some embedded systems like Wear OS. This API enables Android-based hardware to use Google maps data, maps display, and maps gestures and responses. Additionally, it also provides some needed customizability on the maps interface by drawing polygons, lines, shortest paths, and customizable markers (*Maps Android SDK*, n.d.).

Maps JavaScript API. Maps JavaScript API is another API from the Google Cloud suite that enables maps functionality, customizability, and imagery for display on the web. The API features four map types, namely; roadmap, hybrid, terrain, and satellite (*Maps JavaScript SDK*, n.d.).

3.3 Application Requirements

3.3.1 Backend Requirements

Listed below is the overall structure of all connections and relationships among all data, interfaces, users, and the serverless service.

Serverless Architecture

The overall serverless architecture of the application suite and its system is portrayed in the diagram above. Firebase, as the serverless service being utilized, serves as the storage medium for all data being utilized in all of HanApp's interfaces.

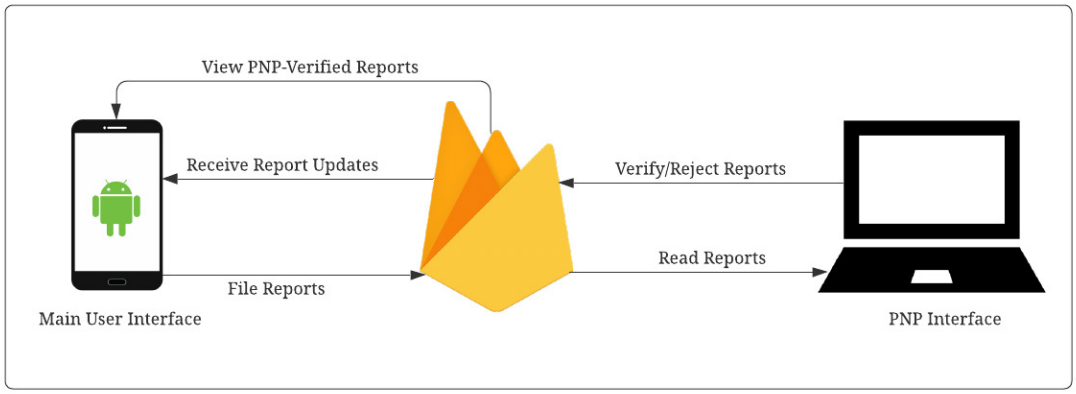


Figure 3.2: Client-Serverless Architecture with Firebase

Realtime Database structure for Main User and PNP profiles

Each profile type will be a node just under the root node as child nodes, namely, Main Users, 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

The 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

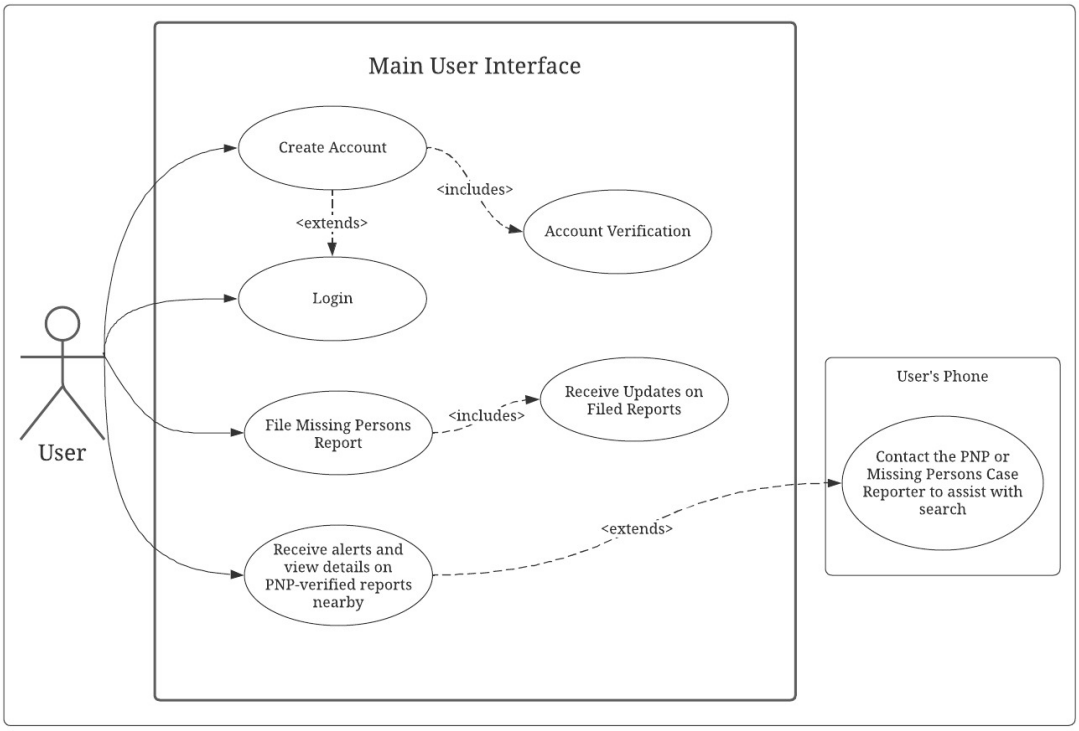


Figure 3.3: Use-Case Diagram for User (Main User Interface)

The User use-case diagram in Figure 3.3 illustrates all the possible tasks that a normal user could do within the main user 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 their email address, their account will be created. The main feature within the application 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.

PNP Admin App Interface

The PNP and PNP Admin interface use-case diagram is shown in Figure 3.4. 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 vicinity.

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 other user account . Once a PNP admin has logged in to the PNP admin interface, he or she, as an official, can then browse and either verify or reject missing person reports filed within their jurisdiction.

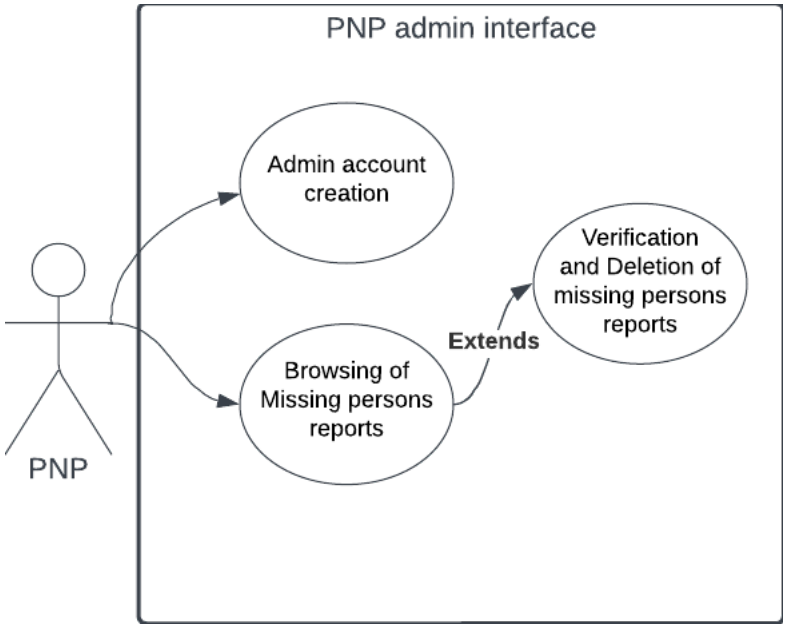


Figure 3.4: Use-Case Diagram for PNP (PNP Admin Interface)

3.3.3 Functional Requirements

The following are the user requirements, for both the main users and PNP users, identified during the design phase of the application suite's development in order

to meet the study's objectives.

User Registration

Main 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, users can then utilize and log in into the application.

PNP Admin Account Creation

For security reasons, creation of PNP Admin accounts will be highly regulated. As such, for the initial release build of the application suite, the PNP accounts are created upon the PNP's request to the developers to create their admin account.

As proof of concept and for testing purposes, the PNP Admin test accounts will be hard-coded by the developers to simulate the process of PNP requesting for an 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 where the reported person went missing has the closest vicinity to, and receive updates via the main app with regards to the status of the report.

With regards to reporting, it is of utmost importance that the report form pages of the main user interface should be intuitive, and be able to condense and simplify the forms required in the PNP's guidelines on reporting missing persons cases.

As seen in Figure 3.5 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 registered as already found or dead in the PNP's own Missing and Found Person Database (MFPD)), then the PNP can verify the report. If any updates are made, such as if the MP is found, it will

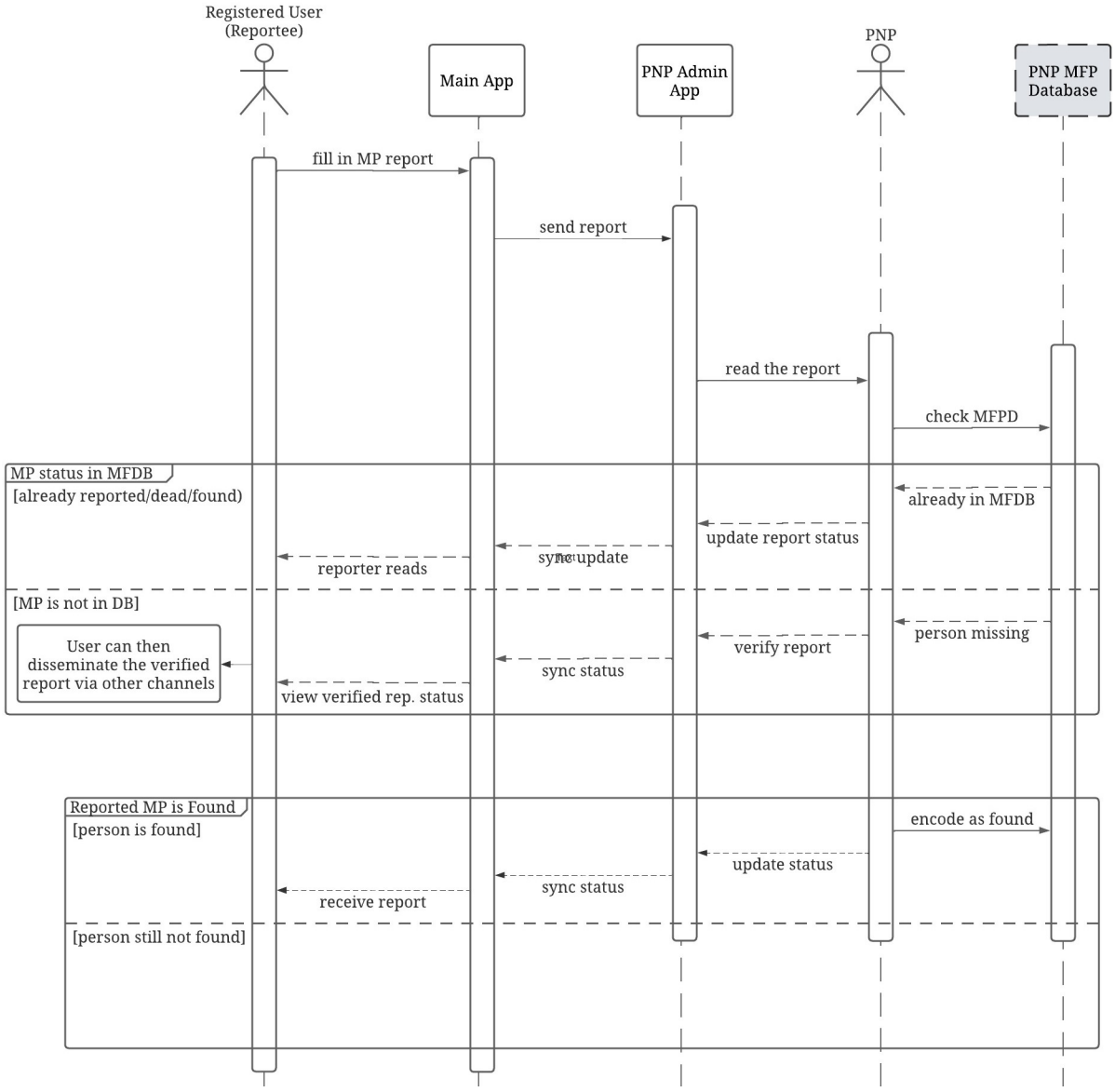


Figure 3.5: Sequence Diagram for Reporting, Updating, and Verification of MP cases

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 A/MP's last known location's nearest PNP station, as mentioned prior. As seen in Figure 3.5 , 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

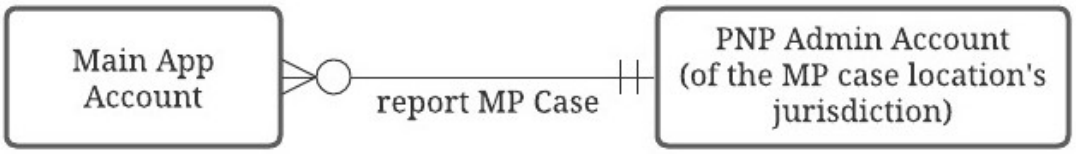


Figure 3.6: Entity Relationship Diagram of Main App Account Reporting to PNP Admin Account

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, etc.), 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 3.5.

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 should receive notifications about an MP case in their area. Users can tap on the marker where the MP was last seen to view the concise details of the MP in the report. As seen in Figure 3.7, all users should be able to view all PNP-verified reports, including those outside their radius, but will only receive notifications for cases within the user's radius.

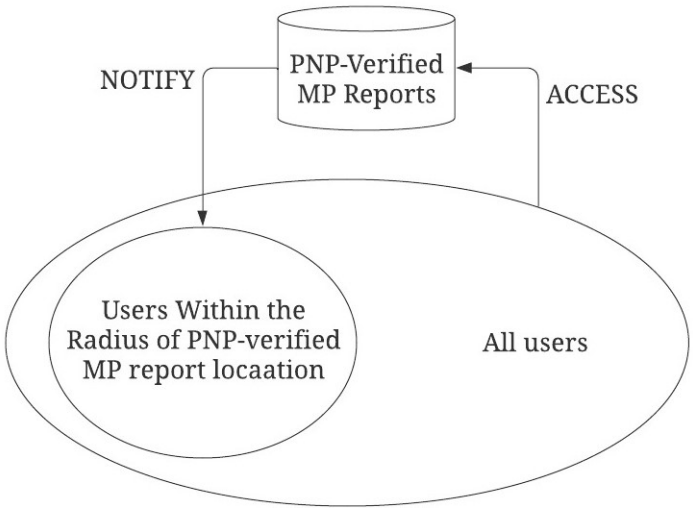


Figure 3.7: Diagram for PNP-Verified Reports Access and Notification System

Chapter 4

Results

This chapter presents the results of the study, specifically the HanApp application suite's various interfaces and screens, features, and backend framework implementation. As such, the figures shown herein are screenshots of HanApp's release build interfaces as well as the backend framework, particularly Firebase.

4.1 PNP Admin Interface

The PNP Admin interface focuses on the managing and administrative side of the applcation suite, providing the necessary access, features, and screens to manage all missing persons reports sent by users through HanApp's main user application interface.

4.1.1 PNP Account Creation and Login Screen

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, they 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 corresponding

admin accounts.

PNP Account Locations

For the proof of concept, four hard-coded PNP accounts, namely; PNP Miagao, PNP San Joaqin, PNP Jaro, and the National PNP were created by the developers directly through Firebase console's Authentication page. These accounts have the corresponding location (their "location data") of the PNP outposts in their locality (i.e. PNP San Joaqin account has the location of San Joaqin PNP outpost).

The PNP location data are utilized to filter out where the reports, based on their MP's last seen location, should be filtering to. However, it's important to note that the National PNP account will receive all reports regardless of distance, this is in order to cater to the reports filed outside of the vicinity of the Miagao, San Joaqin, and Jaro PNP accounts.

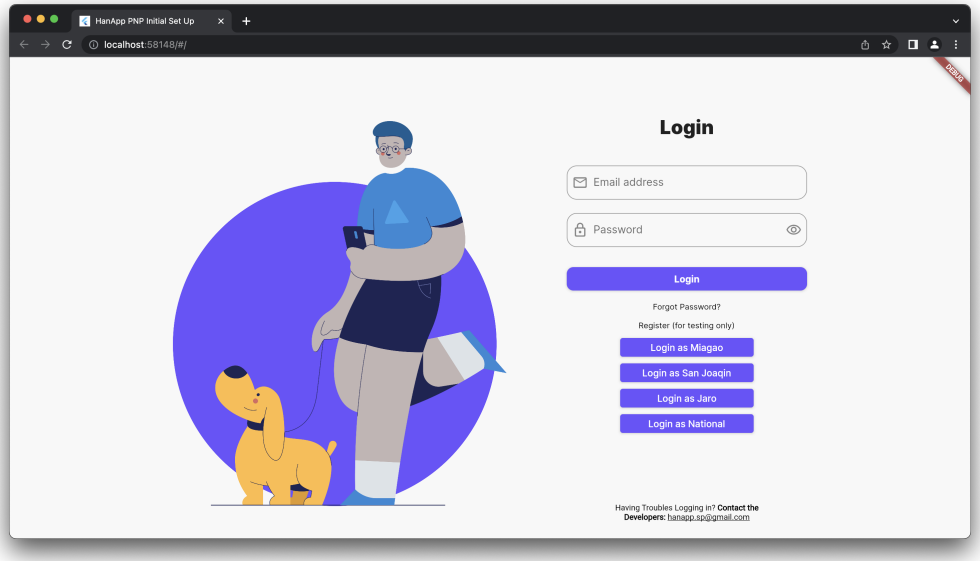


Figure 4.1: PNP Admin Login Page

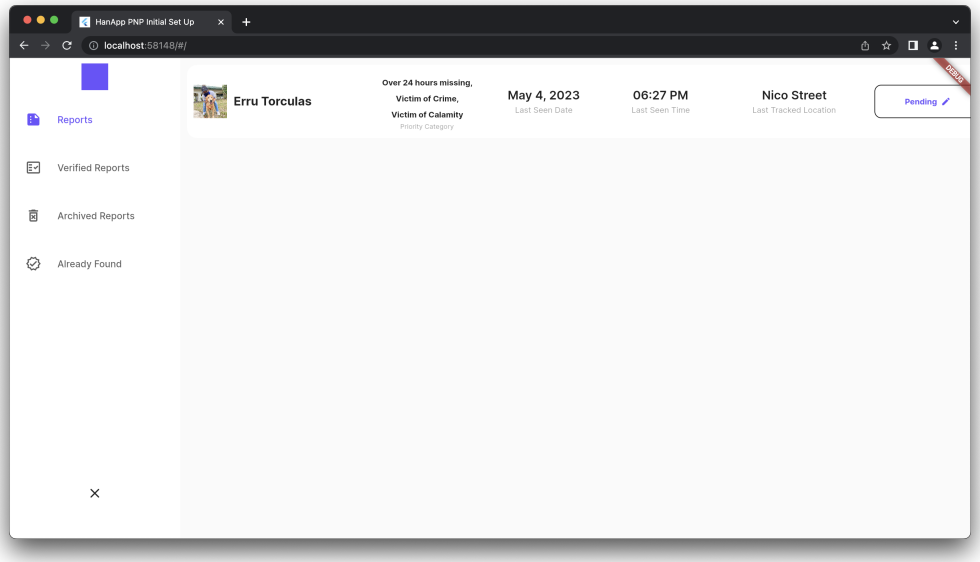


Figure 4.2: Admin Reports Management Suite View

4.1.2 PNP Admin Reports Management Suite

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, reports are correctly being filtered into their respective statuses onto the different menus as seen on the left-hand side of the interface (i.e. verified reports will be on the verified menu).

Viewing Report Details

The report dialog box details the list of the necessary information that the reportee provided. Figure 4.3 illustrates missing person's descriptions, last seen location snapshot, and their picture that will be shown first to ease the process of identification. It's important to note that these are all the same and exact information required by the PNP when filing or verify a report of the missing person as per their guidelines on handling missing persons cases (National Police Com-

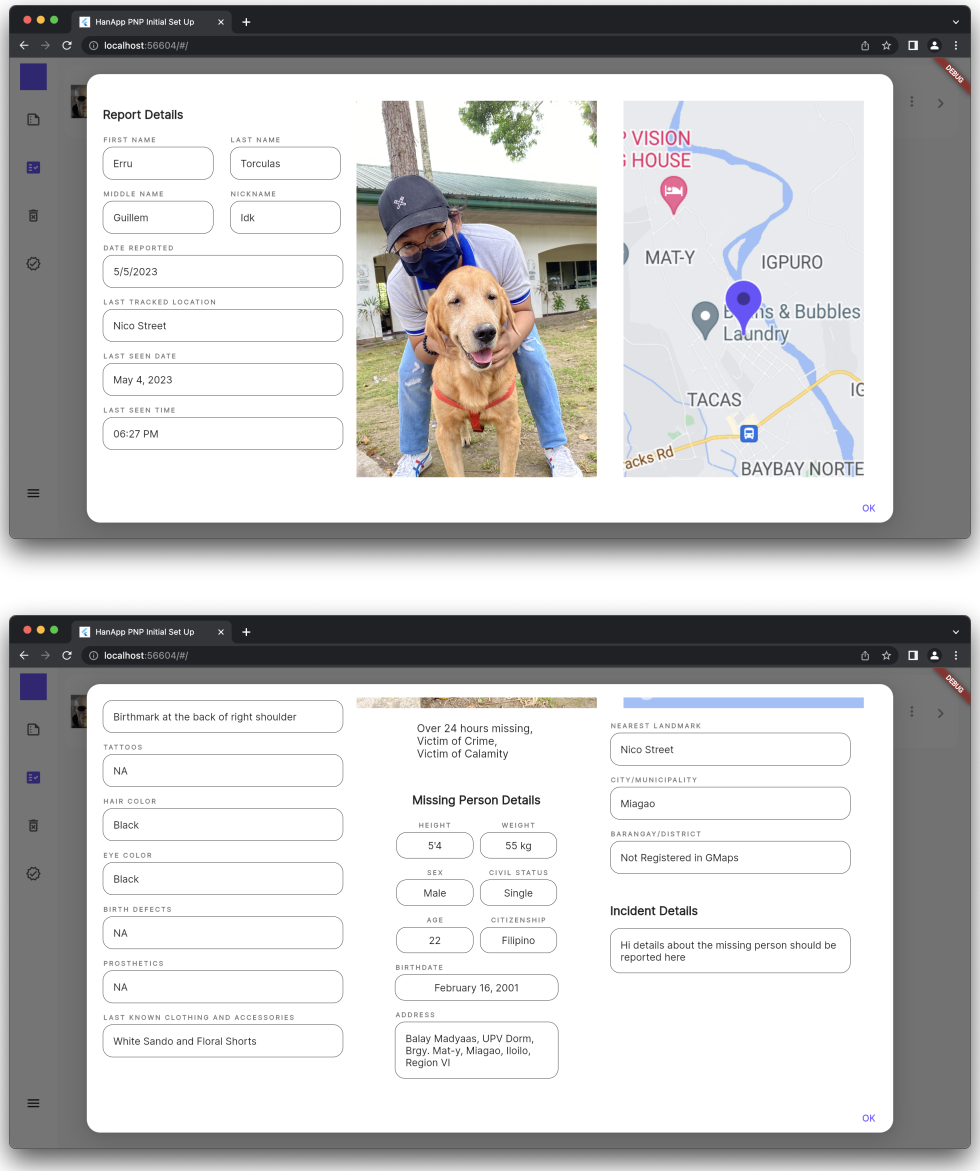


Figure 4.3: Dialog Box of Report Details

mission, 2016). But compared to the numerous forms required in their guidelines, the details are more compact and summarized for their convenience. The data can be retrieved from the database through shared preferences plugin used.

4.2 User or Main Interface

The main interface will be subdivided into three various parts to cover all the features that was built throughout the process.

4.2.1 User Account Registration, Authentication, and Login

Authentication is necessary for the user-side. Figures 4.4 and 4.5 are the Login and Register page, respectively. Input fields will be provided for the registration such as email address, full name, sex, birth date, and a numeric input field will be observed for the phone 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 register button will let the user account be registered and store in the database for authentication purposes.

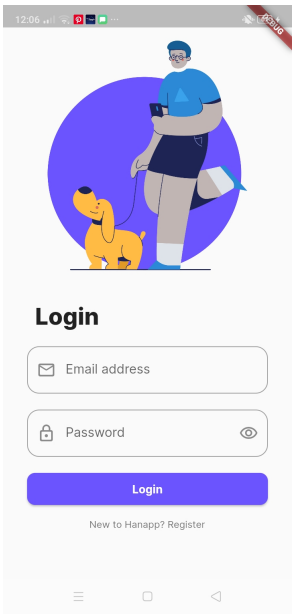


Figure 4.4: Login Page

In order to validate the authentication processes, use-case testing was initiated. The functionality of logging in meets the user requirement for a happy path since the expected output was observed. In order to mitigate any erroneous instances

of unexpected outcome, incorrect credentials was utilized and as expected, the application will deter these circumstances.

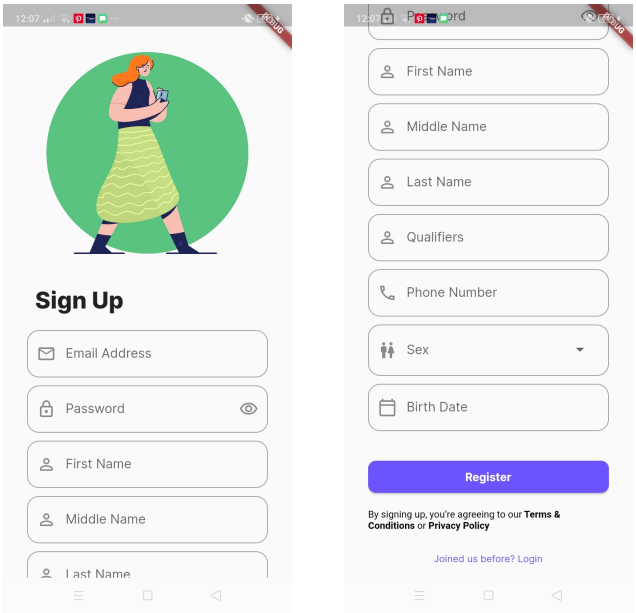


Figure 4.5: Sign Up Page

Account Verification

In order to make sure that newly registered accounts are of human-origin and are legit accounts of potential users, accounts need to be authenticated first before being able to login. Figure 4.6 illustrates the process of verification: first, after the user is registered, they will be redirected to a page where they will be given the choice to send email verification onto their newly registered email address for the application, or to redirect to the login page instead. If users select the option to send the verification email, they can then check their respective emails that ask for verification that, after clicking the link in the email, would mark their registered accounts as verified, allowing them to be able to login. If in some cases that users will opt-out of sending the verification email right after being registered, they still will be met with the same verification page if they try to login with a none email-verified account.

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:

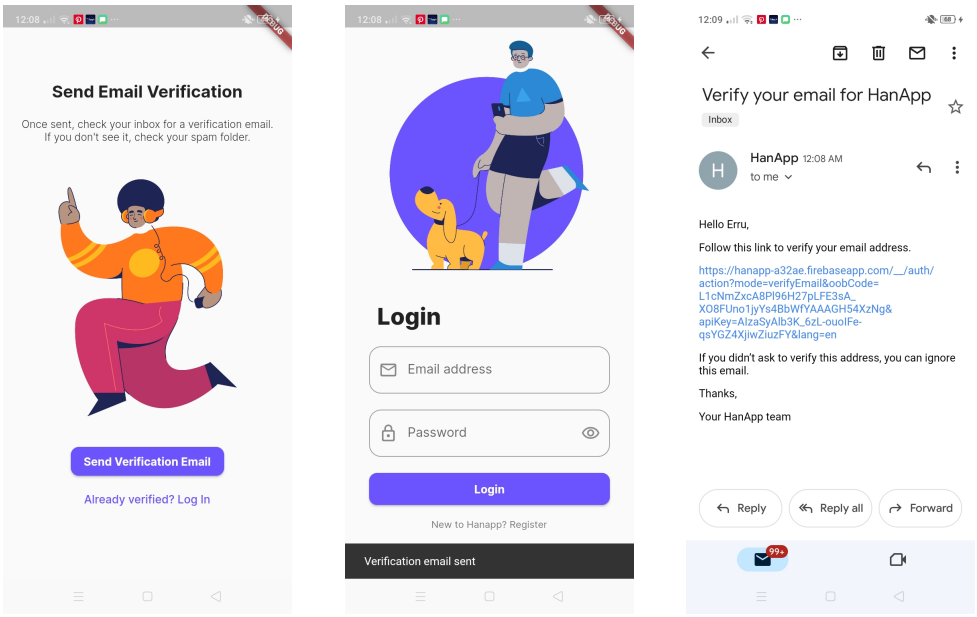


Figure 4.6: Account Verification Process

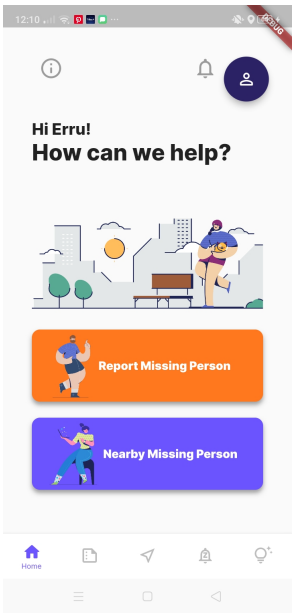


Figure 4.7: Home Page

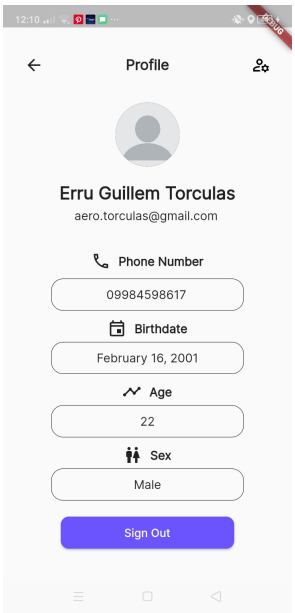


Figure 4.8: Profile Page

Report Missing or Nearby Missing persons as shown in 4.7. This will allow the user to optimally choose what course of action are they going to take on by presenting these call to action button first. A bottom navigation bar will be visible throughout the user's interaction of the app assuring that they can easily

navigate the entire features offered to them. The navigation bar will consist of five (5) sections, namely, Home, Reports, Nearby, Notifications, and Updates. Each section will be utilized to build the features of the app and functions specifically to the needs of the user. 4.8 illustrates the profile page that will consist of the details about the user that was already pre-filled on the registration part. Thus, it will display: the concatenated full name of the user, email address used in account registration, birth-date, age, and lastly sex. In addition, a sign out button will also be provided in case the user decided to log out from their account.

4.2.2 Report Pages

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. It streamlines the reporting process done by the PNP and replicates their incident report, one that had to be done in written format in their respective outposts, so that verification is manageable (National Police Commission, 2016). 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 to disclose such as body marks, tattoos, and the like.

Report Page 1: Classifiers

On the first page of the reports section as seen in figure 4.9, the categorization of the missing person being reported is located. The reporting user will be met with checkboxes indicating that their person being reported is (1) a person missing due to natural calamities, (2) a person under the age of 18, and is therefore classified as a minor, (3) a person that has been missing for over 24 hours already, or lastly, (4) a person that is a victim of a crime such as, but not limited to, kidnapping, human trafficking, etc.

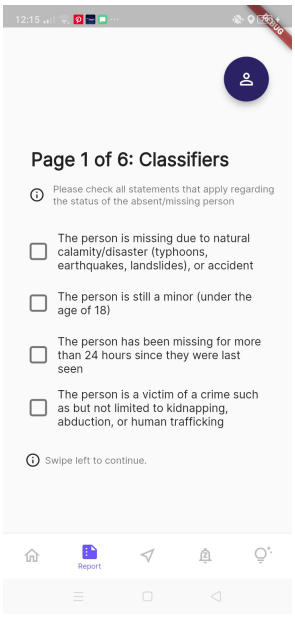


Figure 4.9: Report Form Page 1

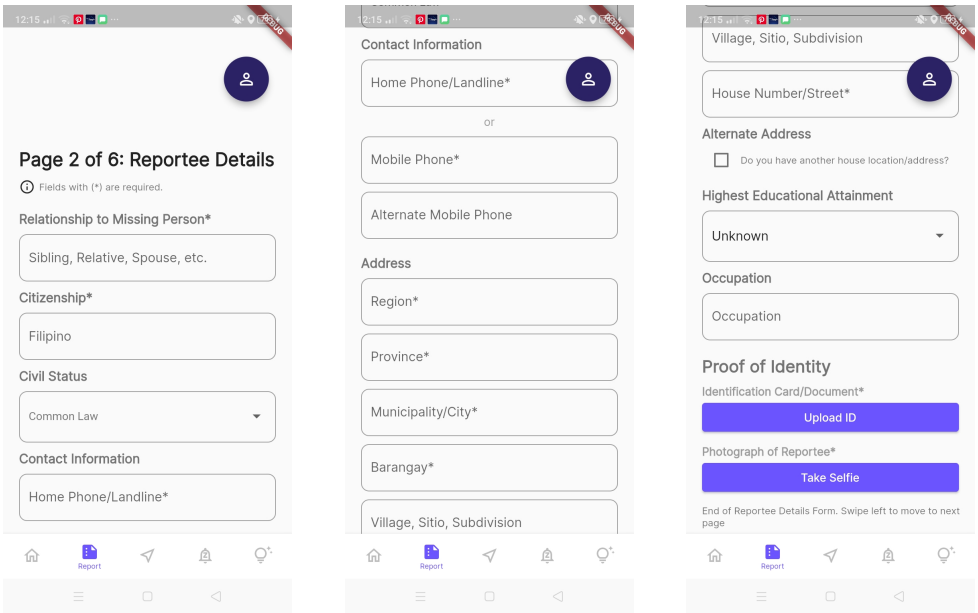


Figure 4.10: Report Form Page 2

Report Page 2: Reportee Details

On the second page of the reports page, as seen in figure 4.10, the reportee will be asked to fill out required information about them. This includes the reportee's

relationship to the missing person, contact information, address, citizenship, and the proofs of identity through a picture of his/her identification card (ID), and a selfie. The selfie serves as a secondary authentication measure in lieu of reporting remotely rather than in person, and also allows the PNP to verify if the user's face matches that of the photo in their submitted ID. The reportee's basic information such as name, birth-date, age, and sex are no longer required to be re-entered since they are now already found in the user's profile.

Report Page 3: Absent/Missing Person Details

The third page of the reports section, as seen in figure 4.11, requires the necessary information about the reported missing person such as their name, birth-date, sex, address, and contact information. It's important to note the specific usage of the term "Absent/Missing Person" as it is the same term used in PNP's required forms when handling missing persons cases, specifically "Annex C: Checklist for Absent/Missing Person".

Report Page 4: Absent/Missing Person Description

The fourth page of the reports section, seen in figure 4.12 still focuses on the information regarding the missing person. This time, however, are information relating to things that can most likely help strangers identify the missing person such as scars, marks, tattoos, hair color, eye color, prosthetic, birth defects, last known clothing and accessories, height, weight, blood type, medications, social media accounts, and of course, the last photograph of the missing person. Dental and Fingerprint records can also be added if the reportee wants to.

Report Page 5: Incident Details

The fifth page of the reports, as seen in figure 4.13 requires information about the incident relating the disappearance of the missing person and is based on the "Narrative of the Incident" section of the PNP's Incident Report Form which is used for reporting incidents including those of missing persons cases.

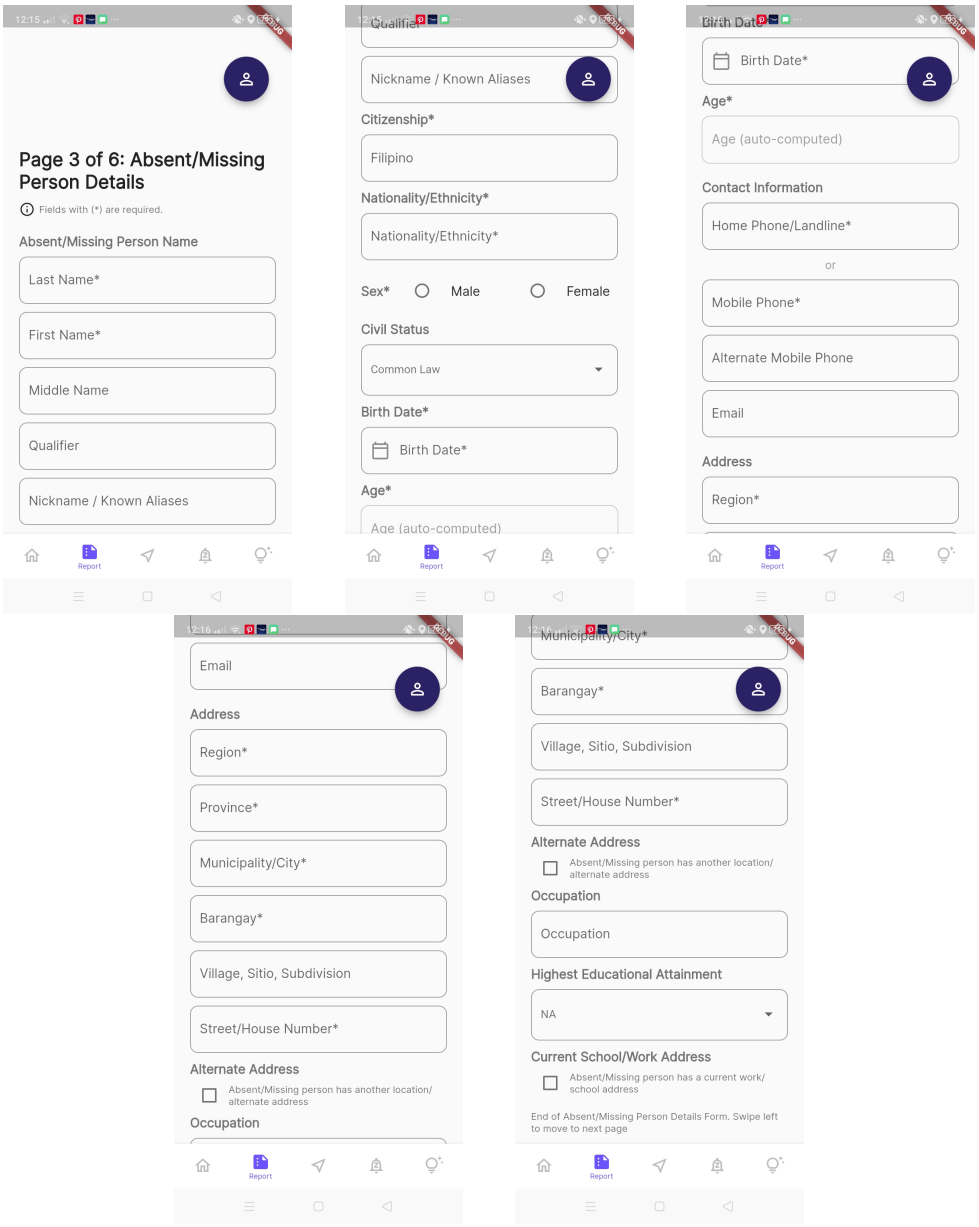


Figure 4.11: Report Form Page 3

The page requires information such as the report date, which is filled in as the current date, the last seen date, where users will be asked to select the date in a calendar when the missing person was last seen, the last seen location, where the users will be prompted to pin the location the missing person was last seen in a dialog that displays a map centered at the current location of the reportee, and the incident details, where users can input further details they deem as necessary

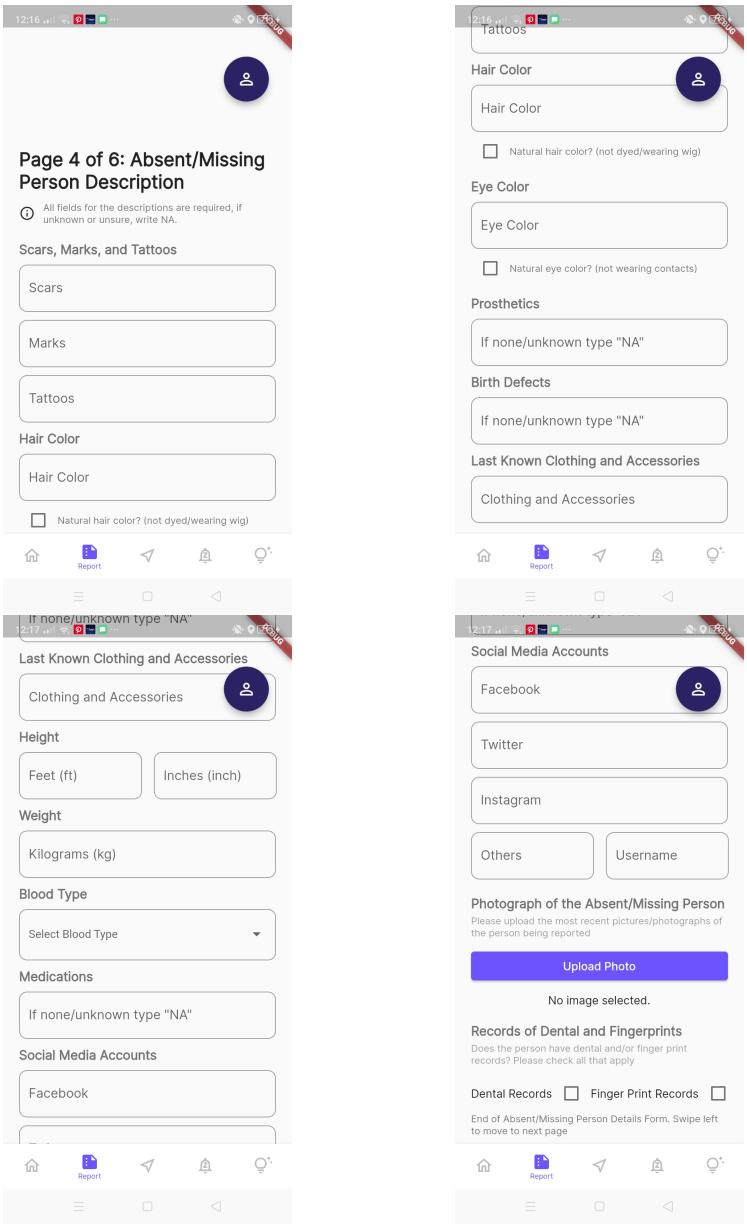


Figure 4.12: Report Form Page 4

in order to aide in the finding of the missing person.

Report Page 6: Confirmation and Authorization

The sixth, and the last page, as seen in figure 4.14 asks for the confirmation and authorization of the user to finally submit the report that they have filled

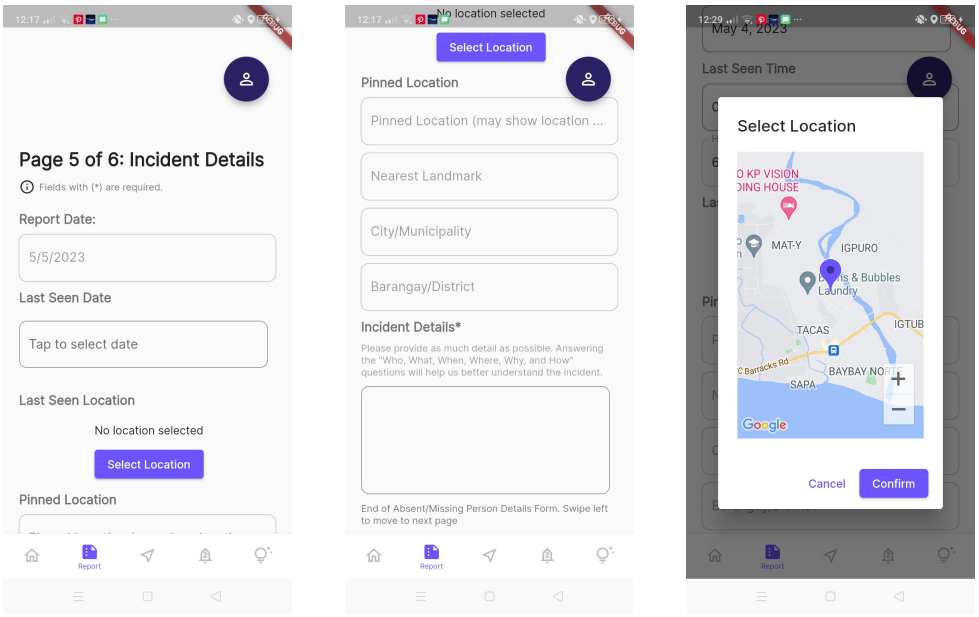


Figure 4.13: Report Form Page 5

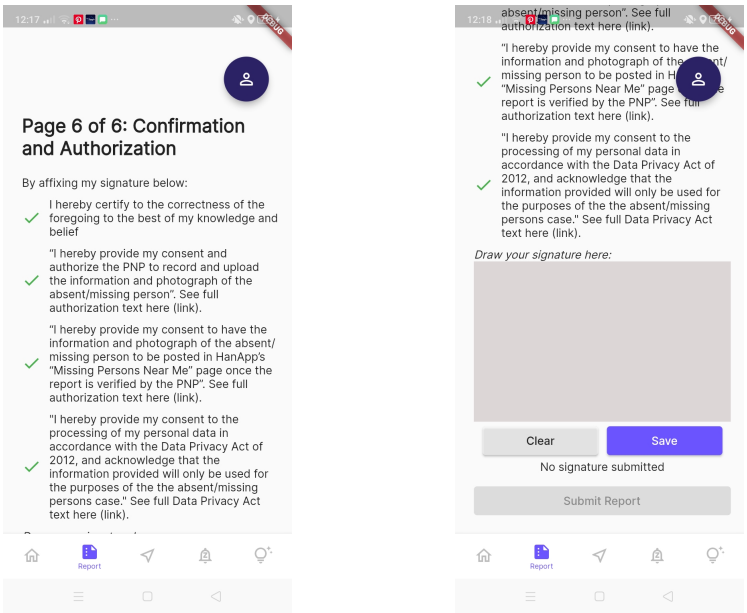


Figure 4.14: Report Form Page 6

out. The page would require the signature of the reportee on the signature pad provided citing that, by signing the report, they agree to have the information submitted published and adhere to the rules and responsibilities that come with them submitting the report to the PNP interface for verification. After which,

the users can then submit the form to PNP.

4.2.3 Nearby Page

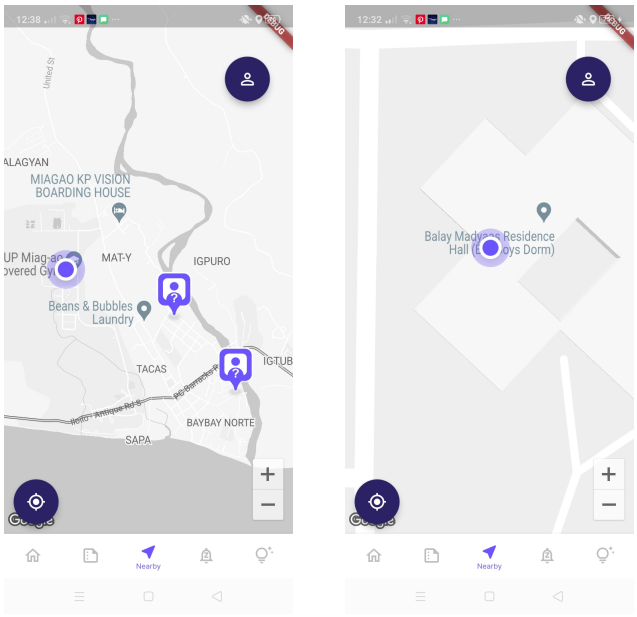


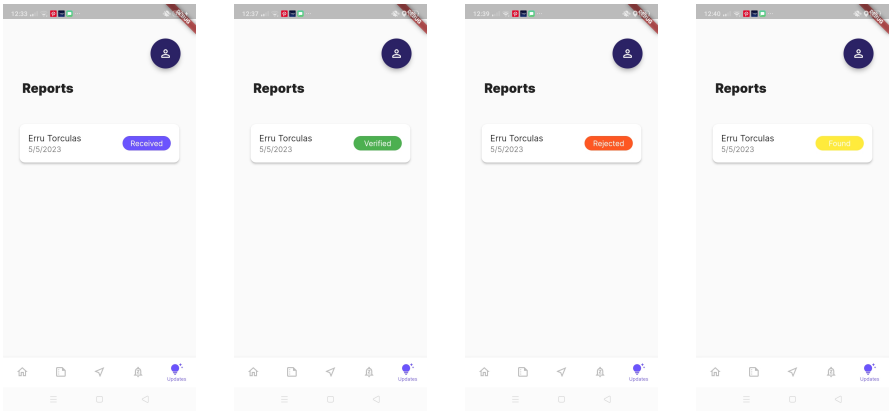
Figure 4.15: Missing Persons and User Location in Nearby

Figure 4.15 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. Only the verified reports that was identified by the PNP through their reports management suite will only be shown. This will allow reports to be validated thoroughly on their end. Apart from that, the user has the liberty to navigate the map, may it be street view or city-wide view by pinching the screen. Also, users can zoom in and zoom out depending on their preference. Landmarks and familiar streets will be shown to guide the users easily on the whereabouts of the missing person. Considering, for example, that a 5-km radius will be visible, it will then encircle the last place the MP was seen. Another thing that the application also wants to note is how the user can tap on the MPs icon to see the details of descriptions pertaining to the individual. Importantly, there is an additional feature that permits the user to find the nearest route going to the last seen location of the missing person by clicking the 'right turn' icon that will redirect them to Google Maps app, given that they have it on their phone installed. Through all of this, it will

enable a swift identification of the person and give the authorities a leeway to make necessary course of action to find these reported missing persons.

4.2.4 Updates Page

Figure 4.16 demonstrates how the user can visibly see the status of their reported missing persons through the Updates section. For instance, when the user submits their report, the PNP has the liberty to change the status of their report given the information that was forwarded to them as shown in Figure 4.2. Not only that, but they will also inspect and verify these as it is sensitive and crucial at this point to confirm whether it needs urgent attention or not (reports with incomplete details would be a factor here). The update of the status is real-time, so when the PNP change its status to a different one, it will automatically change in the user side. The update tags of the reports will consist of four (4) statuses to choose from, namely, Received, Verified, Rejected, or Found. All of these are directly shown to the Updates section of the user's account.



(a) Received (b) Verified (c) Rejected (d) Found

Figure 4.16: Updates Page with Status of Report

4.3 Backend Framework - Firebase

Firebase handles both account management and data management. As such, the following sections will discuss and show the backend framework at which user accounts and reports data is stored and managed.

4.3.1 User Accounts Authentication and Storage

As seen in Figure 4.17, user accounts are stored in Firebase's Authentication page. It shows the user account's unique UID, email, date created, and last signed in page. Firebase also handles the authentication service which was shown in the user account authentication section above.

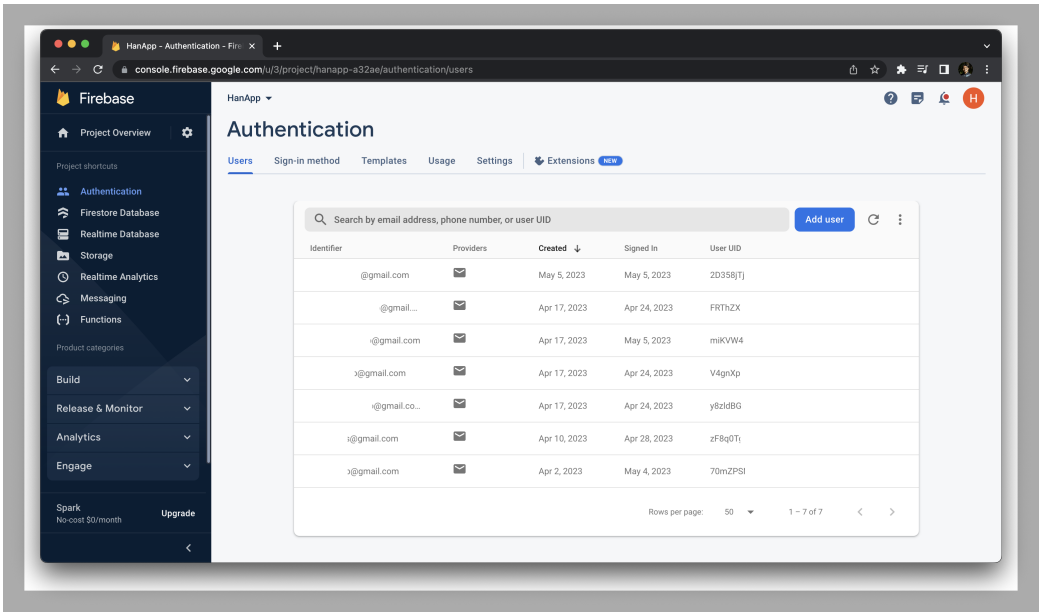


Figure 4.17: Authentication Database

4.3.2 Realtime Database (RTDB) for User Data and Reports

All user data (for main and PNP accounts) and their reports are stored in the RTDB as seen in Figure 4.18. There are three main "nodes" storing the Main Users user data, PNP Accounts user data, and Reports data. The Reports node stores and organizes all reports that have been submitted through the application under each user.

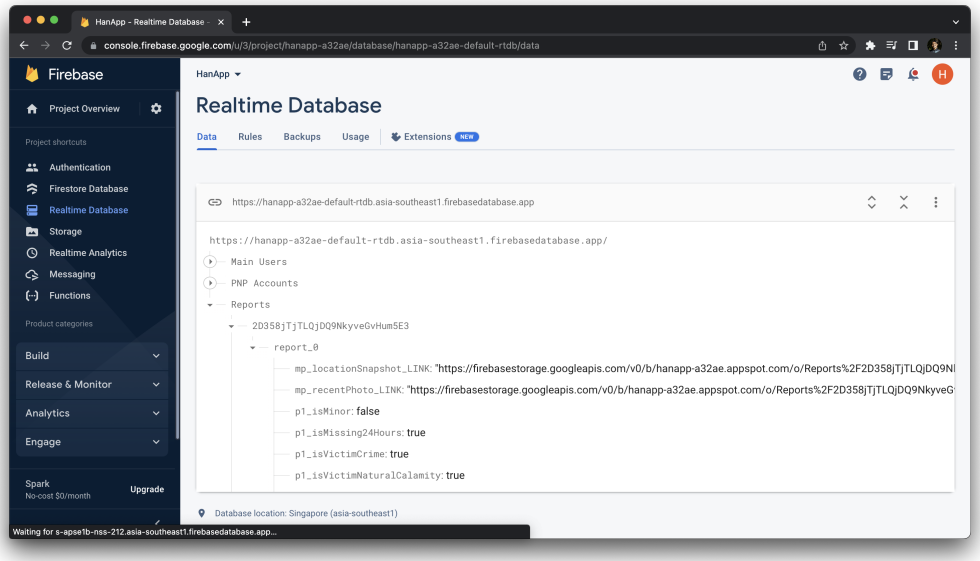


Figure 4.18: Realtime Database User Data and Reports

Firebase Storage for Images Submitted in Reports

Figure 4.19 contains all the images submitted in the reports including the reportee's signature. As seen in the header address line, images are organized under a "Reports" node, under each "User" node, and under each "Report Number" node. This setup allows for a more organized way of storing images submitted for easier retrieval.

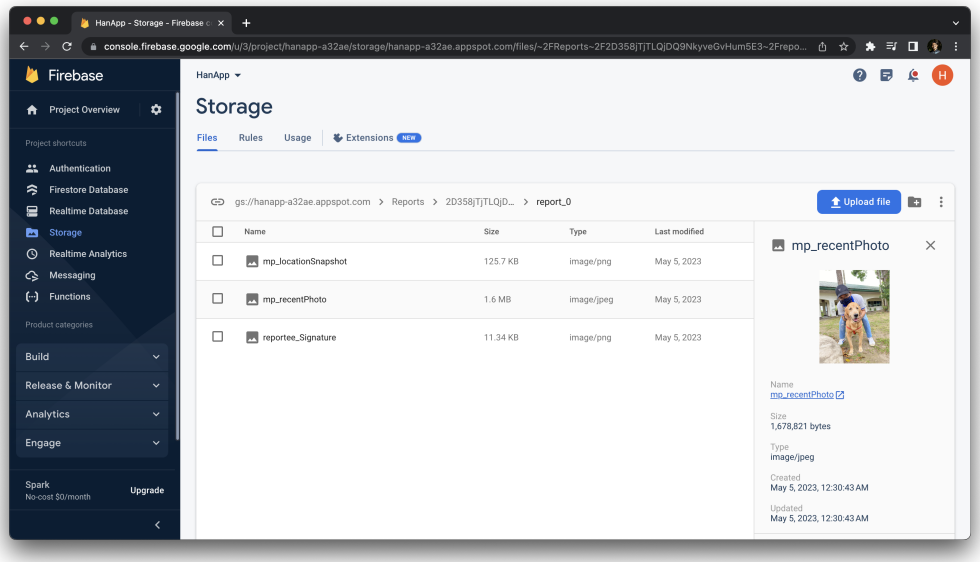


Figure 4.19: Firebase Storage

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

Babich, N. (2017, Jul). *The Most Important Color In UI Design.* Retrieved from https://babich.biz/blue-color/

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.

Firebase auth: Flutter package. (2023, May). Retrieved from https://pub.dev/packages/firebase_auth

Firebase core: Flutter package. (2023, May). Retrieved from https://pub.dev/packages/firebase_core

Firebase storage: Flutter package. (2023, May). Retrieved from https://pub.dev/packages/firebase_storage

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

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.

Hunt, D. (2020). Applying the criminal narrative experience framework to missing children. *Journal of Investigative Psychology and Offender Profiling*, *18*(1), 18–34. doi: 10.1002/jip.1567

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/

Jadhav, A. S., & Sonar, R. M. (2009). Evaluating and selecting software packages: A review. *Information and software technology*, *51*(3), 555–563.

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 android sdk. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/android-sdk/overview

Maps javascript sdk. (n.d.). Google. Retrieved from https://developers.google.com/maps/documentation/javascript/overview

Meet android studio. (n.d.). Retrieved from https://developer.android.com/studio/intro

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 Eye Institute. (2019). *Color Blindness.* U.S. Department of Health and Human Services. Retrieved from https://www.nei.nih.gov/learn-about-eye-health/eye-conditions-and-diseases/color-blindness

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/images/Memorandum%20Circulars/MC%202016-033%20GUIDELINES%20ON%20THE%20RECORDING%20MONITORING%20AND%20INVESTIGATION%20OF%20MISSING%20AND%20FOUND%20PERSONS%20AND%20IDENTIFICATION%20OF%20HUMAN%20REMAINS.pdf

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

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

pub.dev. (n.d.). Retrieved from pub.dev

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

Shared preferences: Flutter package. (2023, Mar). Retrieved from https://pub.dev/packages/shared_preferences

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).

Suchana, K., Alam, S. M., Meem, A. T., Turjo, M. D., & Khan, M. M. (2021). Development of user-friendly web-based lost and found system. *Journal of Software Engineering and Applications*, *14*(10), 575–590. doi: 10.4236/jsea.2021.1410034

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).

Tan, M. L., Prasanna, R., Stock, K., Doyle, E. E., Leonard, G., & Johnston, D. (2020). Modified usability framework for disaster apps: A qualitative thematic analysis of user reviews. *International Journal of Disaster Risk Science*, *11*(5), 615–629. doi: 10.1007/s13753-020-00282-x

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).