**BERA: BILAR EMERGENCY RESPONSE APPLICATION USING DYNAMIC CLUSTERING PROTOCOL**

A Thesis

Presented to the Faculty of the College

of Technology and Allied Sciences

BOHOL ISLAND STATE UNIVERSITY

Zamora, Bilar, Bohol

In Partial Fulfillment

of the Requirement for the Degree in Bachelor of Science in Computer Science

Armando C. Jumawid Jr.

Ronelyn G. Perocho

Mark Henry Perez

May 2024

# APPROVAL SHEET

**ACKNOWLEDGEMENT**

This book will not come to reality if not because of these people who truly supported us all throughout the making of this manuscript.

This work is a humble offering thanks and praises to our Almighty God, for giving the researcher the inspiration and for the good health, courage, strength, and enlightenment.

The researchers express their grateful thanks and appreciation to a number of people who had extended their immeasurable help and wholehearted cooperation.

**Engr. Max Angelo D. Perin**, Panel Expert, and **Mr. Rex Vincent D. Tejada**, **Mr. Joel A. Piollo**, **Nelia Q. Catayas, Ph.D**, Panelist, for sharing their ideas and for the sincere help that contributes towards the realization of this work despite all the trials encountered.

**Mr. Renante M. Avergonzado EdD**, DRRM Director - BISU Bilar Campus, for giving us the idea and information needed for the conduct of this study.

**Mr. Darrel A.** **Cardaña,** Thesis Adviser, for the untiring efforts and for the suggestions, advices, comfort, assistance, guidance, and encouragement swhich leads to the improvement of this manuscript.

**Mr. Joel A. Piollo**, Chairperson of ComSci Department, for his supervision, concerns, advices, and for the support and encouragement extended to the researchers.

**Nelia Q. Catayas Ph.D**, Dean of College of Technology and Allied Sciences, for her encouragement and support for our study.

The **respondents** who generously shared their time and efforts in answering our questionaires, enabling us to gather valuable data for our study.

To the **developer’s parents** and **siblings** for their unconditional love, unwavering support, and understanding throughout the academic endeavor.

The **developer’s loved ones**, **classmates**, **relatives**, and **friends**, we extend our thanks for their prayers, inspiring words, constant love, support, and motivation throughout the entire research process.

To all person whose names may not appear here but who in one way or another gave their valuable time and assistance to pursue the study. May the good Lord continue to fill you his immense love and abundant blessings and may your remarkable humility and service to others return to you a hundredfold.

We love you and thank you all beyond words.

**- The Researchers**

# TABLE OF CONTENTS

**Pages**

**TITLE PAGE……………………………...……………………………………………...i**

[**APPROVAL SHEET ii**](#_Toc164159719)

[**ACKNOWLEDGEMENT** **iii**](#_Toc164159720)

[**TABLE OF CONTENTS** **v**](#_Toc164159721)

[**LIST OF FIGURES** **x**](#_Toc164159722)

[**LIST OF TABLES** **xi**](#_Toc164159723)

[**LIST OF PREVIEWS** **xii**](#_Toc164159724)

[**ABSTRACT** **xiii**](#_Toc164159725)

Chapter

1

[Rationale 1](#_Toc164159726)

[Literature Background 3](#_Toc164159727)

[THE PROBLEM 10](#_Toc164159728)

[Statement of the Problem 10](#_Toc164159729)

[Objectives 11](#_Toc164159730)

[Scope and Delimitation 11](#_Toc164159731)

[Significance of the Study 13](#_Toc164159732)

[RESEARCH METHODOLOGY 14](#_Toc164159733)

[Conceptual Diagram of the Study 14](#_Toc164159734)

[Block Diagram of the Study 15](#_Toc164159737)

[Development Model and Approaches 16](#_Toc164159739)

[Requirements Planning 17](#_Toc164159741)

[User Design 17](#_Toc164159742)

[Construction 17](#_Toc164159743)

[Cutover 17](#_Toc164159744)

[Software Development Tools 18](#_Toc164159745)

[Environment and Participants 20](#_Toc164159746)

[Data Collection 20](#_Toc164159747)

[OPERATIONAL DEFINITION OF TERMS 23](#_Toc164159750)

2 PRESENTATION OF FINDINGS, ANALYSIS, AND INTERPRETATION OF DATA

[Existing Operation and Practices 24](#_Toc164159751)

[Needs of the Existing Operation 25](#_Toc164159752)

[Data Flow Diagram 27](#_Toc164159753)

[Event List Diagram 28](#_Toc164159755)

[Used Case Diagram 30](#_Toc164159760)

[Use Case Narrative 30](#_Toc164159762)

[Database Design 35](#_Toc164159770)

[Class Diagram 35](#_Toc164159771)

[Data Structure 36](#_Toc164159773)

[Program Hierarchy 38](#_Toc164159777)

[Functional Requirements 39](#_Toc164159779)

[Non-Functional Requirement 41](#_Toc164159780)

[Technical Requirements 48](#_Toc164159791)

[Minimum Hardware Specifications 48](#_Toc164159792)

[Screen Layout 49](#_Toc164159794)

[Economic Performance Evaluation 58](#_Toc164159795)

[Testing and Evaluation 59](#_Toc164159797)

[System Usability 59](#_Toc164159798)

[3](#_Toc163652550) SUMMARY OF FINDINGS, CONCLUSION AND RECOMMENDATION

[Summary of Findings 62](#_Toc164159800)

[Conclusion 63](#_Toc164159801)

[Recommendations 64](#_Toc164159802)

[REFERENCES 56](#_Toc163652554)

[**APPENDICES**](#_Toc163652555)

A. Letter of Intent [59](#_Toc163652556)

B. Letter of [Implementation 60](#_Toc163652558)

[C. System Usablity Questionnaire 61](#_Toc163652560)

D. [Interview Guide Questions 62](#_Toc163652562)

E. User’s Manual……………………………………………………………..63

F. Source Code………………………………………………………………...64

G. Researchers Biodata……………………………………………………..65

# LIST OF FIGURES

**Figure Page**

[1 Conceptual Diagram of the BERA Application 14](#_Toc164081584)

[2 Block Diagram of the Study 15](#_Toc164081585)

[3 RAD Development Model 16](#_Toc164081586)

[4 Contextual Diagram of the Present System 27](#_Toc164081587)

[5 Request Help Process 28](#_Toc164081588)

[6 Recording Requestor Data 28](#_Toc164081589)

[7 Locating the Requestors 29](#_Toc164081590)

[8 Generation of Reports 29](#_Toc164081591)

[9 Use Case Diagram of Bilar Emergency Response Application 30](#_Toc164081592)

[10 Class Diagram of the BERA 36](#_Toc164081593)

[11 Program Hierarchy 39](#_Toc164081594)

# LIST OF TABLES

**Table Page**

[1 Distribution of System Usability Respondents 21](#_Toc164156199)

[2 Interpretation Guide of the System Usability 21](#_Toc164156200)

[3 Send “Help” (Requestor) 31](#_Toc164156201)

[4 Manage Emergency Call (Requestor) 31](#_Toc164156202)

[5 Signing Up - User Registration (Responder/Requestor) 32](#_Toc164156203)

[6 Receiving Incident Notification and Dispatch (Responder/Admin) 32](#_Toc164156204)

[7 Account Management – (Administrator) 33](#_Toc164156205)

[8 Editing of Profile Information (Requestor/Responder) 34](#_Toc164156206)

[9 Logging Out (All) 34](#_Toc164156207)

[10 User Account Credential 37](#_Toc164156208)

[11 System User 37](#_Toc164156209)

[12 Incident Information 38](#_Toc164156210)

[13 Minimum Android Hardware Specifications 49](#_Toc164156211)

[14 Initial Investment and Annual Operating Costs 58](#_Toc164156212)

[15 System Usability Assessment Result N=10 60](#_Toc164156213)

# LIST OF PREVIEWS

**Table Page**

[1 Responder & Requestor/Witness Login Page 50](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157441)

[2 Splash Screen 50](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157442)

[3 Sign Up Page & Forgot Password Page 51](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157443)

[4 Requestor/Witness Home Page 52](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157444)

[5 Map Page 52](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157445)

[6 Contact Page & Profile Page 53](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157446)

[7 Incident Dialog & Responder Homepage 54](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157447)

[8 Admin Login & Emergency History 55](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157448)

[9 Admin Map & Manage Requestor/Witness Accounts 56](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157449)

[10 Manage Responders 57](file:///C:\Users\ARA%20MAE%20JUMAWID\Downloads\Revised\BERA-CHAPTER-1-3-FINAL%20REVISION%20V1.3.docx#_Toc164157450)

# ABSTRACT

Nowadays, there are numerous catastrophic events, which affects a large number of people in the world including crises, fires, floods, road accidents, earthquakes, and terrorist attacks. The majority of public people use mobile and internet all day long. Emergency response systems play a crucial role in ensuring rapid and efficient assistance during critical situations. This research paper presents the development and evaluation of the application called “Bilar Emergency Response Application (BERA), using Dynamic Clustering Protocol” to enhance the effectiveness of emergency response operations. The proposed system leverages the ubiquity of smartphones and their inherent capabilities to facilitate real-time communication, coordination, and resource allocation among responders and affected individuals. Through dynamic clustering, BERA optimizes the allocation of resources based on the evolving nature of emergencies, ensuring timely assistance to those in need. The effectiveness of the application is evaluated through simulations and real-world scenarios, demonstrating its potential to significantly improve emergency response outcomes. BERA represents a promising advancement in the field of emergency management, offering a scalable and adaptable solution to address the challenges of modern-day emergencies.

Keywords: Bilar Emergency Response Application (BERA), ubiquity of smartphones, real-time communication, dynamic clustering

**Chapter 1**

**THE PROBLEM AND ITS SCOPE**

### Rationale

Emergencies are inherently unpredictable, occurring anywhere and anytime, necessitating rapid decision-making. This unpredictability can disrupt the usual chain of command, potentially leading to lapses in judgment and significant losses. The general public is not exempt from a various security threat, both internal and external. These threats can cause physical harm and even death. Among the causes of death are heart disease, traffic accidents, and death caused by criminal activities.

Building upon the insights gained from the global response to the COVID-19 pandemic, as highlighted by Erkhembayar et al. (2020), our research aims to further enhance emergency response mechanisms. The proactive measures taken by countries, such as Mongolia, underscore the importance of efficient emergency response frameworks, facilitated by legal structures like the State Emergency Committee and the Disaster Protection Law. Despite these frameworks, challenges persist in promptly reporting emergencies, especially in remote areas where victims may encounter difficulties in using traditional communication methods like phone calls or SMS.

Drawing from established research, such as the work of Khalemsky & Schwartz (2017) & McNab et al. (2009), which highlight the transformative impact of mobile emergency response applications, our study seeks to expand upon these insights. Khalemsky & Schwartz, (2017) demonstrated the substantial reduction in response times achieved through mobile emergency applications, resulting in faster medical assistance and improved patient outcomes. McNab et al. (2009) provided valuable insights into the design principles of these applications, emphasizing the importance of achieving optimal performance. In the context of our research, these studies underscore the potential of advanced technology in enhancing emergency services and response effectiveness

To address the challenge of improving emergency response times in remote areas, the implementation of this research can provide a centralized platform for rapid emergency calls, precise GPS location information, direct communication with local emergency services, and emergency requestor identification. This comprehensive approach enhances the region's emergency response capabilities and better protects its residents.

As technology advances, it provides alternatives, and solutions begin to emerge. One suggested technology is a personal emergency notification mobile application (Hong et al., 2017). Technologies like Bilar Emergency Response Application (BERA) are crucial in enhancing response times and effectiveness. Through this application, the public can effortlessly dispatch emergency messages to family, friends, or relevant institutions. The application aids in facilitating evacuation by providing information about the sender's location and the nature of the threat encountered. This application is anticipated to make a practical contribution to the community by offering a swift communication method during crucial situations. It is also expected to support existing emergency assistance services by providing valuable assistance to authorized agencies.

### Literature Background

According to Republic Act No. 10121, also known as the "Philippine Disaster Risk Reduction and Management Act of 2010," this legislation provides a robust legal and policy framework for disaster risk reduction and management in the Philippines. By adhering to the provisions of this act, the application aims to contribute to enhancing the country's disaster preparedness, response, and recovery efforts. It underscores the commitment to promoting resilience and reducing the adverse impact of disasters on communities and individuals, as envisioned in RA 10121.

Republic Act No. 10121, enacted on May 27, 2010, serves as the legal foundation for the Philippine Disaster Risk Reduction and Management System. The law outlines the country's commitment to addressing vulnerabilities, enhancing institutional capacities, and building community resilience to disasters and climate change impacts. It emphasizes a holistic and proactive approach, incorporating international principles, and aims to integrate disaster risk reduction into various aspects of development, governance, and peace processes. The Act underscores the importance of gender-responsive, community-centered, and environmentally sustainable disaster risk reduction and management practices.

The Senior Technology Exploration, Learning, and Acceptance (STELA) model, as proposed by Tsai et al. (2019) in their longitudinal randomized controlled trial published in Educational Gerontology (45(12), 728-743), aims to investigate the intricate dynamics of technology exploration and learning. As individuals age, the process of adopting new technologies becomes increasingly complex.

As we enter the era of globalization, technology usage optimization should be put into practice and continue to grow in the area of personal and public security. This is important because as we advance in the globalization timeline, the risks to our security in every area of our lives are at risk.

The Women's Design Service in London, UK, developed the "Making Safer Places" procedure in Bristol, Wolverhampton, London, and Manchester. The procedure uses a tool called the "fear-o-meter" to investigate the things that make women afraid (Whitzman, Legacy et al. 2013). The women conducted a survey of their domestic neighborhood, estate, and playground using highly participatory equipment in order to determine the aspects of the physical environment that need to be improved. For instance, rerouting roads, establishing fences, and identifying and moving vegetation.

Both personal and public safety are issues that could be addressed by further improving the current environment in our society. We cannot dispute that neglected infrastructure exists in both our urban and rural areas, and that these areas need to be improved in order to stop serving as magnets for criminal activity.

In Steurer's (2018) study on worldwide crime, homicide rates and burglary/housebreaking rates were analyzed across 198 countries, utilizing data from Knoema. The examination unveiled a diverse range of crime rates, prompting the application of logarithmic transformation to attain a more normalized distribution.

As they say, "Prevention is better than cure," so law enforcement should seriously think about using technology to deter crime. Given that crime prevention could be improved in an efficient and effective manner, this could help law enforcement officials in their duties.

Four stages of disaster management are typically recognized in the field: mitigation, preparedness, response, and recovery (Zlatanova et al., 1998). Preparation is mostly focused on routine planning within the emergency services and law enforcement (e.g., for emergency situations such as police, ambulance, fire). Every phase is interrelated and vital, but from the perspective of saving lives, the response and recovery phases are frequently regarded as the most important.

As a result, it is imperative that law enforcement respond to calls from crime victims as soon as they are received. In order to lessen the aftereffects of a crime towards the victim or victims who made the call, moments like these should be handled carefully and quickly.

The idea of an event timeline is one that is frequently applied in emergency response. This outlines the incidents, emergency calls, reactions, and other actions in chronological order. that take place throughout an event Sene (2008). Timelines are useful for post-event response assessment and can be made instantly available to help other responders comprehend the circumstances.

Clustering algorithms serve as essential tools in data analysis, enabling the automatic grouping of similar data points and unveiling patterns within large datasets. By categorizing data into distinct clusters, these algorithms simplify the interpretation of complex information, aiding researchers and practitioners in extracting meaningful insights. Their versatility extends to various domains, contributing to market segmentation, image analysis, biological data interpretation, and more, making clustering algorithms indispensable for effective data exploration and decision-making.

According to Xu and Wunsch's influential research on clustering algorithms, cluster analysis emerges as a crucial tool for comprehending unlabeled data, with their exploration of hierarchical structures and group formation emphasizing specific goals while acknowledging ongoing efforts to address associated challenges (Xu & Wunsch, 2005). In another study by Xu, D., & Tian, Y. (April 2015), an overview of commonly used clustering algorithms is presented, introducing their basic ideas, specifying sources, and analyzing the advantages and disadvantages of 19 selected categories, aiming to offer readers a systematic and clear understanding of this important data analysis method. A study by Na, S., Xumin, L., & Yong, G. (April 2010) explores the significance of clustering analysis in data mining, emphasizing the direct impact of clustering algorithms on results. It specifically discusses the drawbacks of the standard k-means algorithm, proposing an enhanced version that optimizes efficiency by utilizing a simplified data structure to store information across iterations, ultimately improving both the speed and accuracy of clustering.

Regarding our law enforcement's emergency responses, there is one thing we can all agree upon. In an emergency, emergency response should always occur as quickly as possible to protect victims of crime from additional harm from the already-occurring crime.

The following articles provide valuable insights and resources for creating and enhancing emergency response systems and applications, offering essential guidance for developers seeking to improve disaster preparedness and response capabilities.

Designing Mobile Applications for Emergency Response: Citizens Acting as Human Sensors, the authors conducted an investigation of emergency notification (EN) mobile applications, aiming to analyze their characteristics and practical usefulness. They used the Design Science Research (DSR) approach and identified that while generic social applications are commonly used for large-scale crises, specific EN applications are more effective for small-scale events. They also found that users prefer multimedia features over text and forms in such applications, suggesting the potential for improved usability and adoption in emergency situations.

The review of emergency response in disasters: present and future perspectives study conduct a systematic analysis of 3,678 publications (1970–2019) from the Web of Science to investigate the emerging field of emergency response research in disasters. The analysis employs bibliometric and social network analysis methods, revealing key research topics such as emergency response, simulation, optimization, emergency medicine, and education. The paper also identifies four primary research themes and highlights two research hotspots ("optimization" and "demand"), providing valuable insights and directions for future research in the field of emergency response.

Disaster management and emerging technologies: a performance-based perspective paper employs a systematic literature review (SLR) and VOSviewer software to analyze the impact of emerging technologies (ETs) on disaster management (DM) processes. It highlights the complexity and varying terminology in the DM field, emphasizing the importance of clarifying phases and roles. The study identifies key ETs, such as simulation, robotics, IoT, and social media, and their associations with different DM phases, emphasizing the potential of simulation for preparedness, robotics and IoT for response, and social media for performance measurement, management, and accountability. Additionally, the paper suggests future research directions and practical implications for enhancing DM performance using ETs.

User-Centered Design enhances the research by providing a fundamental framework for developing an effective emergency response application, such as "BERA." Examination of technology-driven strategies in "BERA" seeks to reveal the profound impact of user-centered design principles on its overall effectiveness. This approach ensures the app is meticulously tailored to meet user needs, especially in high-stress emergency scenarios.

The integration of a clustering protocol algorithm within the "BERA" emergency response application plays a vital role in enhancing situational awareness and response prioritization. By systematically identifying incident clusters and directing resources in a data-driven manner, the application not only reduces response times but also contributes to overall user safety.

There are numerous related applications that are running in different organizations and institutions. Among these significant studies where:

Citizen – This App provides COVID contact tracing, real-time safety alerts, and 24/7 assistance at your fingertips. In the midst of the pandemic, it helps track potential COVID exposures and offers key features like free at-home testing, crime alerts, police activity updates, breaking news videos, and safety alerts for loved ones. Citizen app is a powerful tool for personal safety and COVID-19 awareness, puts vital information and resources in your hands, ensuring you stay informed and prepared in challenging times.

Emergency + – This national app utilizes the GPS built into smartphones to display users' location coordinates. In the event of an emergency, when users dial Triple Zero (000), this app enables them to convey their precise location to the emergency call-taker. Additionally, the app provides information on other national numbers such as Crime Stoppers, Health Direct, and the National Relay Service. Furthermore, it includes built-in accessibility features that audibly describe on-screen content, allowing callers to use the app even without visual interaction. Vision Australia conducted a review of the app to ensure compliance with accessibility requirements.

EchoSOS – EchoSOS was created to improve first contact and communication between people requesting help and the emergency services. EchoSOS is constantly being developed to face the challenges in rescue and health care and to improve communication and information exchange. EchoSOS was developed with the primary goal of enhancing the initial contact and communication between individuals seeking assistance and emergency services, and it continues to evolve to meet the ever-evolving challenges in rescue and healthcare, prioritizing the enhancement of communication and information exchange for more effective emergency responses.

SirenGPS – Dialing 911 from a mobile phone doesn't bring instant aid, because dispatchers need some location info to find you. SirenGPS (Android, iOS) puts them at the touch of one big red button. If your community subscribes to Siren 911, nearby first responders will receive your location and profile (emergency contacts, medical history, allergies and current medications, which you put into the app), improving your chance of being rescued in time.

## 

## THE PROBLEM

### Statement of the Problem

This study aimed to design and develop an application called BERA for Bilar Search and Rescue Unit (BISARU) in BISU to improve communication, coordination, and response times during emergencies.

Specifically, it seeks to answer the following questions:

1. What are the current processes for receiving emergency calls and tracking the location of emergency during response operations?

2. What features were essential in the development of the BERA?

3. Do we need to register in BERA?

4. How to design the application with the modules?

1. Emergency Notification
2. User Authentication and Role Management
3. Location Tracking
4. Clustering Protocol
5. Reports
6. What is the level of application acceptability regarding the usability of the emergency response application, as perceived by the target users?

### Objectives

The main objective of the study is to design an application called Bilar Emergency Response Application (BERA).

Specifically,

1. To develop an Emergency Response Application for BISARU in Bilar, Bohol.
2. To test and evaluate the system using System Usability Survey.
3. To implement and deploy the application called BERA in Bilar Bohol.

### Scope and Delimitation

The proposed application focused on the following modules:

* **Emergency Notification –** This module's scope involves defining how emergency alerts are received, processed, and communicated within the application. It may include features for reporting different types of emergencies and the level of detail provided in alerts.
* **User Authentication and Role Management –** secure access to the emergency response application, allowing users to register, log in, and recover passwords while maintaining security measures.
* **Location Tracking –** The scope of the location tracking module includes tracking and displaying the real-time location of emergency responders and individuals in distress, utilizing advanced GPS technology for accurate positioning. Additionally, users who have been registered within a 1 km radius of an incident will be located and displayed for efficient coordination and response.
* **Clustering Protocol -** Integrating a clustering algorithm within the Emergency Response Coordination, enhances situational awareness, and streamlines response prioritization. By systematically identifying incident clusters and directing resources in a data-driven manner, the application not only reduces response times but also contributes to overall user safety during emergency scenarios.
* **Reports** **–** This module's scope includes the generation of reports and analytics related to response times, resource utilization, and overall system performance. It defines the types of reports and the key performance indicators to be tracked.

The study may predominantly pertain to Bilar Bohol, Philippines, and might not be directly applicable to regions with markedly distinct geographic attributes, infrastructure, or emergency response requirements. The system will exclusively be available on platforms like Android mobile devices with provisions for offline functionality in areas with restricted connectivity. The application will solely perform functions such as incident reporting, location tracking, communication, mapping, user management, reporting, notification, and data storage, all aimed at augmenting emergency response capabilities within the confines of the municipality of Bilar.

### Significance of the Study

This project is to bring positive implications to the society in the issue of requesting for help during critical or emergency situations. In particular, the following entities may benefit from this study:

* Affected Individuals and Communities
* Researchers
* Future Researcher

## RESEARCH METHODOLOGY

### Conceptual Diagram of the Study

Figure 1 below represents the conceptual diagram of the study that represents the principle of input-process-output. Inputs are the incident reports of emergency requestor or wiitness collected by the administrator. The process involves incident reporting, location tracking, user authentication, clustering and resource allocation. The output includes reports detailing incidents and response times.

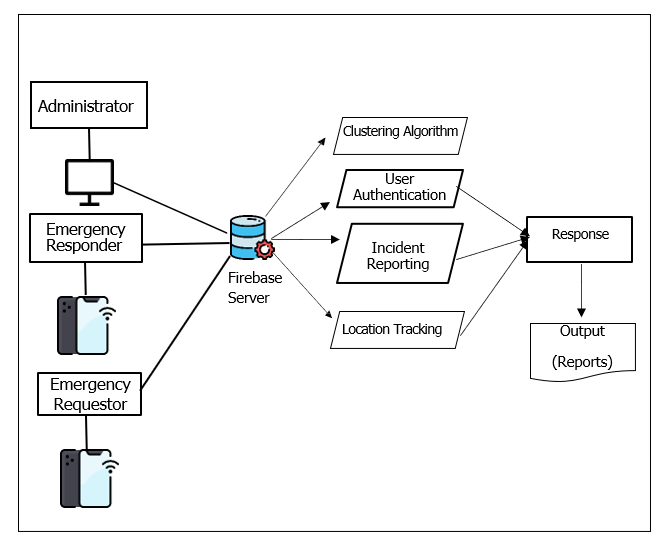


Figure 1. Conceptual Diagram of the BERA Application

### Block Diagram of the Study

Figure 2 shows the block diagram of the study Bilar Emergency Response Application Using Dynamic Clustering Protocol, Bilar, Bohol. It covers the specification of the basic functionality of the application that represents the work process of the Responders and Users. The other function of the app is the generation of reports.

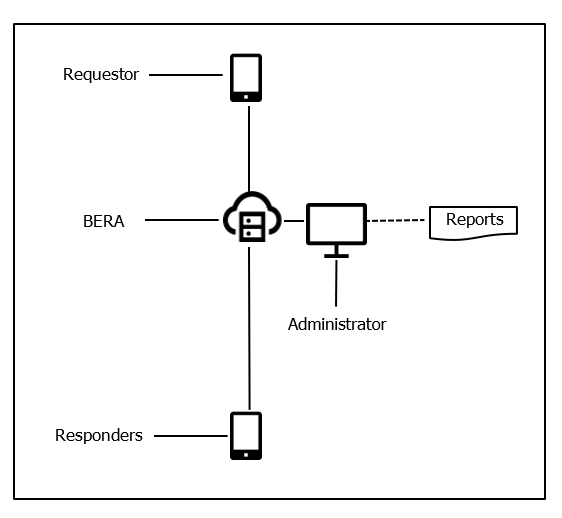
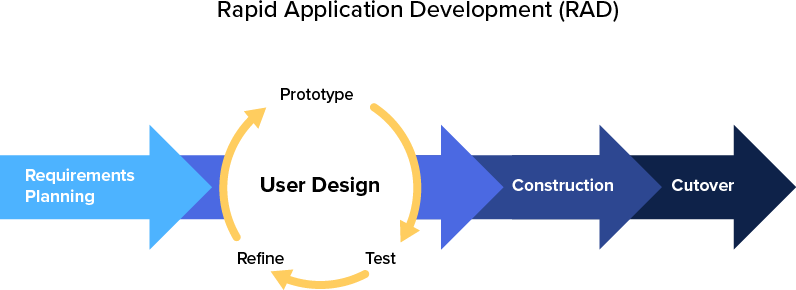


Figure 2.Block Diagram of the Study

### Development Model and Approaches

The researchers use a development strategy called Rapid Application Development (RAD) which puts an emphasis on immediate feedback and rapid prototyping over lengthy development and testing cycles. This ensures a quality and client-focused end product. Rapid Application Development or RAD means an adaptive software development model based on prototyping and quick feedback with less emphasis on specific planning. In general, the RAD approach prioritizes development and building a prototype, rather than planning. With rapid application development, developers can quickly make multiple iterations and updates to the software without starting from scratch. This helps ensure that the final outcome is more quality-focused and aligns with the end users’ requirements (KissFlow, 2022). Figure 3 shows the RAD (Rapid Application Development)



Adopted from KissFlow (2022), [https://kissflow.com/application- development/rad/rapid-](https://kissflow.com/application-development/rad/rapid-application-development/) [application-development/](https://kissflow.com/application-development/rad/rapid-application-development/)

Figure 3. RAD Development Model

**Requirements Planning**

The researchers and client cooperated in identifying and documenting the critical requirements for the emergency response system. This collaborative effort ensured that the development process aligns with the specific needs and objectives of the local emergency response ecosystem.

**User Design**

The researcher developed the user design through numerous iterations. Customers are shown iterations of the system to make sure their needs are met. Developers work quickly to produce prototypes with a variety of features and functions rather than adhering to a strict set of specifications. These prototypes are then presented to BISARU for them to determine their preferences and dislikes.

**Construction**

Since the majority of the issues and changes were addressed during the thorough iterative design phase, the researchers can now construct the final iteration of a module or the system itself. This phase is broken down into several steps: preparation for rapid construction, program and application development, coding, unit, integration, and system testing.

**Cutover**

This is where the finished product goes to launch. BISARU and its crew can now use the newly released application. Data conversion, final tests and training will proceed right after. Still, the researchers and the client continue to look for bugs and problems that immediately needs addressing until finalization will complete.

### Software Development Tools

The following are the tools used in the development of Bilar Emergency Response Application using Dynamic Clustering Protocol of Bohol Search and Rescue Unit (BISARU) in Bilar, Bohol.

1. **Android Smartphone (API 21)** – The mobile device will be the main source of communication between the requestor and the responder. It is used to set up and use the **Emergency SOS** feature, which can help you call for help, share your location with your emergency contacts, and record video in an emergency situation. Also, this device will instantiate communication to the Firebase server via API key.
2. **Google Maps (v111.0.5563.8)** - A service offered by Google that provides access to map data, APIs, tools, and libraries. Integrating Google Maps Platform allows emergency responders to see user location, plan optimal routes, and gain situational awareness around incidents.
3. **Visual Studio Code (v1.75)** – Visual Studio supports web and mobile development; it allows us to create responsive and cross-platform applications. Develop web-based dashboards for real-time monitoring or mobile apps that can be used by responders in the field. It is the IDE used by the researchers to develop Bilar Emergency Response Application.
4. **Flutter (SDK 3.3.5) –** An open-source, cross-platform Software Development Kit framework used by the researchers to develop the application. Flutter supports offline functionality, allowing the application to continue functioning even when an internet connection is unavailable. This is crucial for emergency response scenarios where network connectivity may be unreliable.
5. **Dart (SDK 2.18.2) –** The programming language used in Flutter framework in developing cross-platform applications. Dart is used to support asynchronous programming, which is crucial for handling tasks such as fetching real-time data, making API calls, and processing background tasks.
6. **Git (v2.93.1) –** A DevOps tool used for source code management. It is used to track changes in the source code and allow multiple developers to work together. It allows non-linear development through branching. This is used by the researchers to manage the source code, collaborate effectively, and ensure a robust and organized development process.
7. **Firebase (v12.7.0) –** A Google-backed *BaaS* (Backend as a Service) application. The chosen backend service by the researchers. Firebase provides tools for tracking analytics, reporting, app crashes and creating market and product experiments.

### Environment and Participants

The study was conducted at BISARU located in the municipality of Bilar, Bohol. It has an estimated of 5-10 emergencies per month and working forces which consist of paramedics, emergency medical technicians (EMTs) and staff.

The participants of the study included the staff, paramedics and emergency medical technicians (EMTs). They gave their thoughts by rating the application usability questionnaires. BISARU provided us with the data and helped the development of the system successful.

### Data Collection

A letter of permission was put together to show support and make sure that the proposed system would be tested and put into use. Before starting to build the application, we talked to the staff and owner to find out what exactly was needed and what was missing. This helped us understand how the Business Information System Application Requirements and Understanding (BISARU) workflows work, which became the foundation for setting the necessary standards and requirements for the application.

Additionally, we took a practical approach, which involved watching how things are done and closely looking at the forms and reports that are currently being used. We did this to make sure we could easily include these existing elements in the design of the proposed graphical user interface (GUI) forms.

The application modules were derived from the events and scenarios outlined in the existing BISARU procedures. To assess whether the developed system fulfills the requirements of end-users, a system usability survey was administered. The study's participants included 15 individuals, consisting of one (1) paramedic, one (1) EMT, one (1) staff member, and 13 clients. Table 1 presents the breakdown of the respondents participating in the system usability assessment.

**Table 1.Distribution of System Usability Respondents**

|  |  |
| --- | --- |
| **Respondents** | **Frequency** |
| Bilar Search and Rescue Unit EMT | 2 |
| Bilar Search and Rescue Unit Staff | 1 |
| IT Professional | 2 |
| Computer Science Student | 10 |
| **Total** | 15 |

Table 2 shows the interpretation of the results used for system usability. The rating was done based on the interpretation guide of device factors using ISO-25010.

**Table 2.Interpretation Guide of the System Usability**

|  |  |  |  |
| --- | --- | --- | --- |
| **Weight** | **Range** | **Description** | **Interpretation** |
| 5 | 4.6 – 5.0 | Strong Agree | The respondents strongly believe and confident that the  system is very usable. |
| 4 | 3.7 – 4.5 | Tend to Agree | The respondents tend to believe that the system is  usable. |
| 3 | 2.8 – 3.6 | Neither Agree nor  Disagree | The respondents are neutral in  trusting that the system usable. |
| 2 | 1.9 -2.7 | Tend to Disagree | The respondents tend not to  trust that the system is usable. |
| 1 | 1.0 – 1.8 | Strongly Disagree | The respondents strongly believe that the system is not  usable. |

To determine the acceptability of the system, the weighted mean score was computed to evaluate the system usability level the following formula:

**1 f 1 + 2 f 2 + 3 f 3 +**

**WMS 4 f 4 + 5 f 5**

**n**

Where: =

WMS = Weighted Mean Score

f1 = frequency of respondents who give a rate of 1

f2 = frequency of respondents who give a rate of 2

f3 = frequency of respondents who give a rate of 3

f4 = frequency of respondents who give a rate of 4

f5 = frequency of respondents who give a rate of 5 n = number of respondents

1,2…5= constant(rating)

### OPERATIONAL DEFINITION OF TERMS

The terminologies and acronyms used or mentioned in this study and were further defined operationally in this study:

1. BERA – is the name of the software application means Bilar Emergency Response Application
2. BISARU – is the name of the government agency in Bilar, Bohol, Philippines, specifically the Bilar Search and Rescue Unit.
3. SMS – short message service
4. GPS – stands for Global Positioning System. It is a satellite-based navigation system that allows users to determine their approximate location (latitude, longitude, and altitude) anywhere on Earth.
5. EN – Emergency Notification
6. DSR – Design Science Research is a research paradigm that focuses on the development and evaluation of artifacts, such as systems, models, designs, or methodologies, to address complex problems.
7. SLR - stands for "Systematic Literature Review." It is a research method used to systematically identify, evaluate, and interpret all available research relevant to a particular research question, topic area, or phenomenon of interest.
8. VOS viewer - is a software tool used for constructing and visualizing bibliometric networks.
9. ET - stands for "Emerging Technologies refer to technologies that are newly developed or are in the process of becoming more widely adopted.

**Chapter 2**

**PRESENTATION OF FINDINGS, ANALYSIS AND INTERPRETATION OF DATA**

### Existing Operation and Practices

The Bilar Search and Rescue Unit (BISARU) operates around the clock to provide assistance with emergency concerns throughout the area of Bohol. BISARU continues to employ manual processes for the identification and location of emergency concerns. BISARU acknowledges the need for advancements and is actively exploring opportunities to integrate modern technologies for more efficient and effective emergency response.

**A. Receiving Emergency Through Communication Channels**

The traditional operational workflow of emergency response agencies involves a manual sequence of steps, beginning with the reception of emergency calls through various communication channels such as mobile phones, telephones, or frequency radios.

**B. Emergency Verification**

Upon receiving a distress call, the specialized response team promptly initiates a thorough verification process to confirm the authenticity and urgency of the reported incident though call or onsite assessment.

**C. Deployment**

The response team promptly proceeds to the incident location, where they actively engage in the process of confirming essential details related to the accident, carefully assessing factors such as its nature and impact, and ensuring

accuracy in pinpointing the precise of the incident.

**D. Generation of Reports.**

The generation of reports involves systematically documenting and recording pertinent details in a dedicated logbook, highlighting the ongoing need to refine and enhance the process for maintaining a comprehensive and organized record of relevant information.

These comprehensive verification steps are essential for ensuring accurate and effective emergency response. However, inherent delays are incurred in the overall emergency response due to the manual nature of these processes and the constraints imposed by travel time to reach the incident site promptly.

### Needs of the Existing Operation

The present process will employ a cross-platform application-based approach, which can be customized to better suit user needs. The researcher’s observations led to the identification of the following needs:

1. To develop an application with communication and location tracking capabilities for users in need of assistance through the use of Internet, ensuring that responders have information about emergency situation. The goal is to enhance response through effective communication and provide location details by the use of google maps.

**BERA: BILAR EMERGENCY RESPONSE APPLICATION USING DYNAMIC CLUSTERING PROTOCOL**

After all the data and information that was gathered from the current system of BISARU Bohol and developed the “BILAR EMERGENCY RESPONSE APPLICATION”, a system that would handle emergency concerns more efficiently, optimize the verification process, and automate the generation of reports.

**A. Administration**

Authorized individuals are required to log in to access the system by entering their email and password in the login form. This security measure is implemented to prevent unauthorized access and potential breaches, safeguarding responder and user personal information from loss or compromise. Within the administration module, authorized personnel can adjust settings and oversee user roles.

**B. Emergency Notification and Dispatch**

The new system can generate alerts efficiently notifying designated responders. It provides details to inform response, and the system logs and tracks the entire alert and response process for comprehensive record-keeping and continuous improvement of emergency response procedures.

1. **Location Tracking**

The system can pinpoint the location of an incident through the use of geolocation technology. The location tracking feature allows for the quick identification of the exact site, facilitating timely assistance and intervention.

**D. Generation of Reports.**

The system will generate reports containing data on responders, including registered app users. This encompasses the utilization of graphical data visualization methods, such as daily, weekly, monthly, and annual incident reports. These reports may be formatted in tables, ready for convenient printing.

### Data Flow Diagram

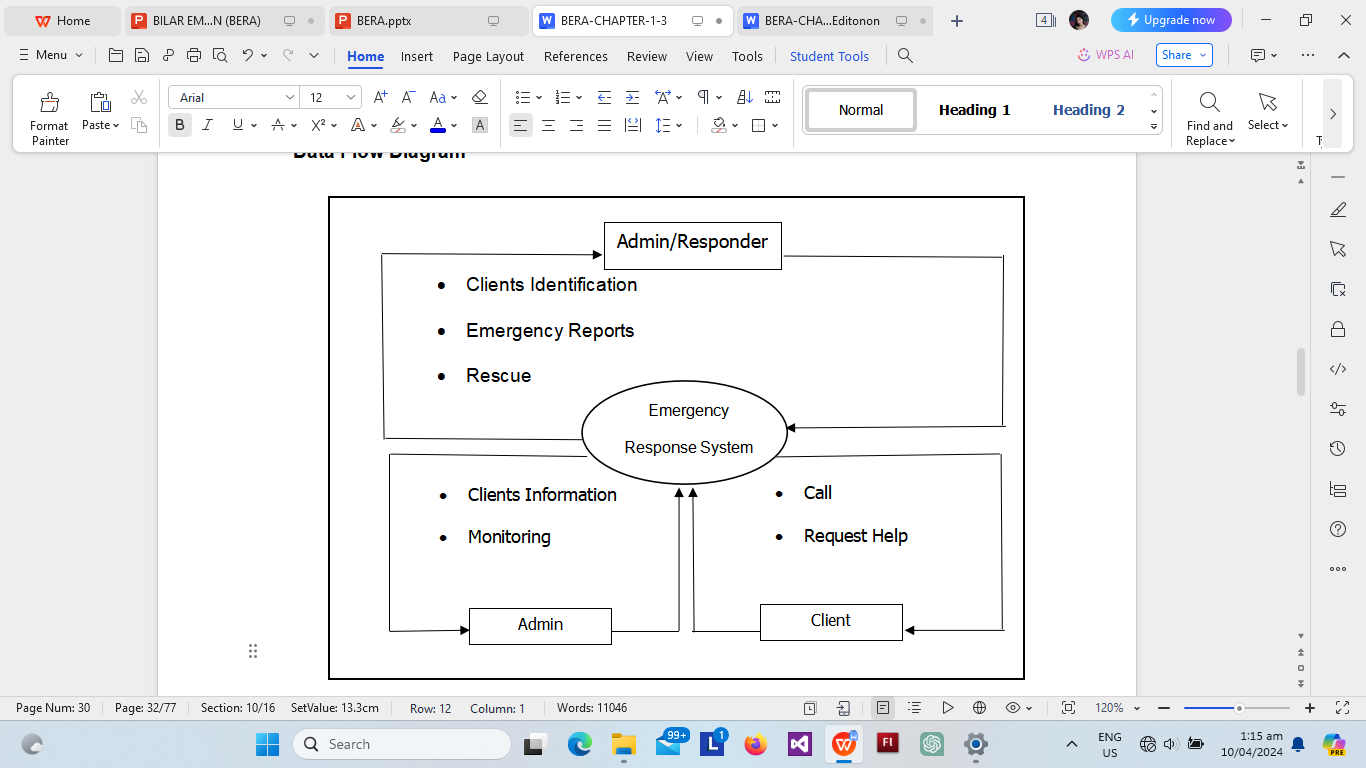
****

Figure 4.Contextual Diagram of the Present System

**Event Specifications:**

**Event list:**

1. Requesting Help Process
2. Recording of Requestor Data
3. Locating the Requestors
4. Generation of Reports

### Event List Diagram

Figure 5 show the event diagram of inquiry process of the present system.

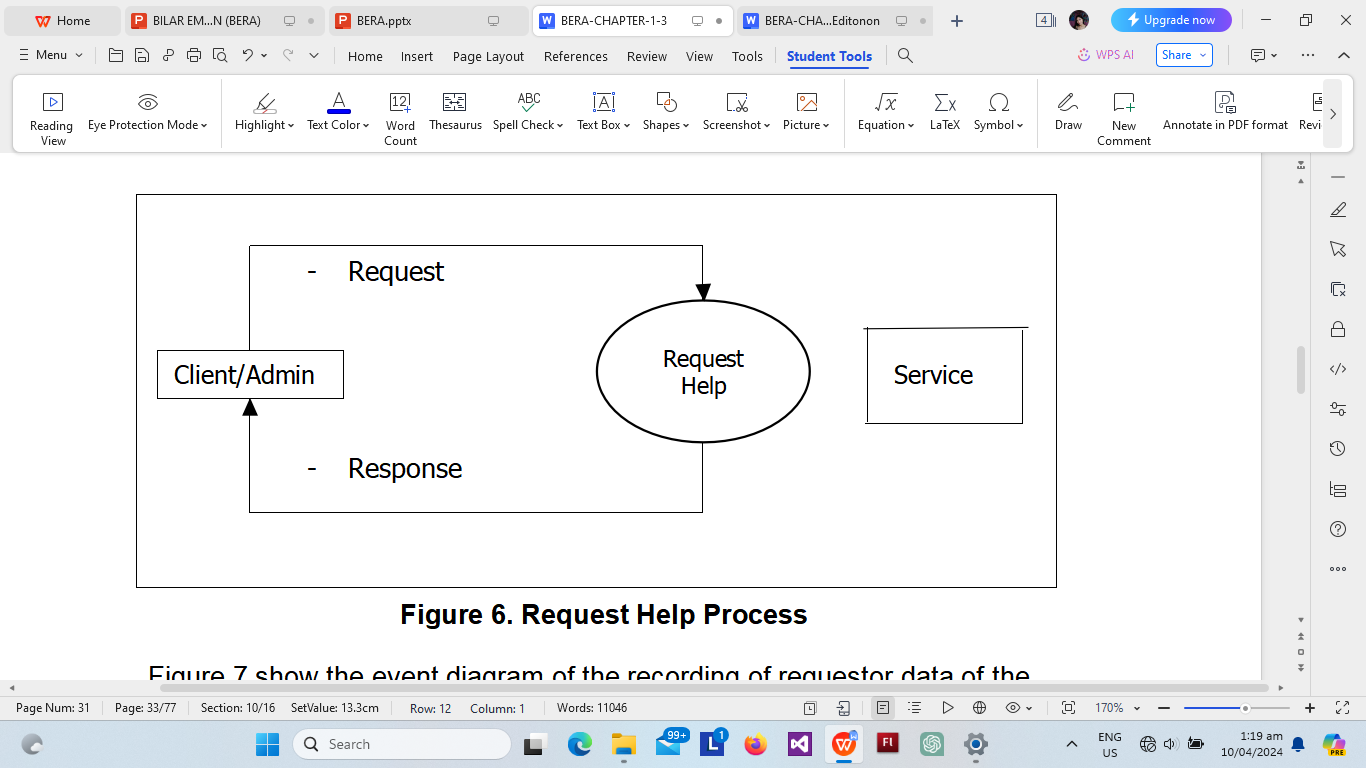


Figure 5. Request Help Process

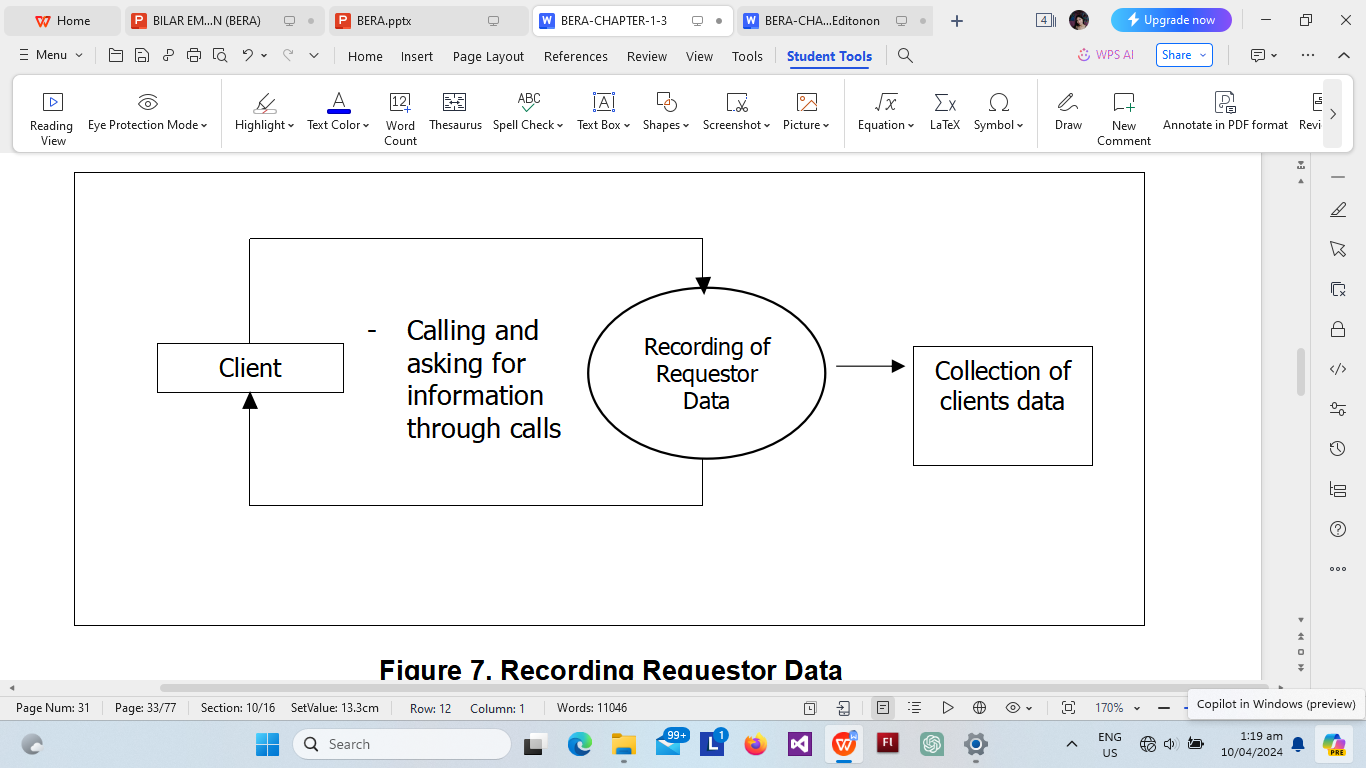
Figure 6 show the event diagram of the recording of requestor data of the present system. ****

Figure 6.Recording Requestor Data

Figure 7. show the event diagram of Locating the Requestors of the present system

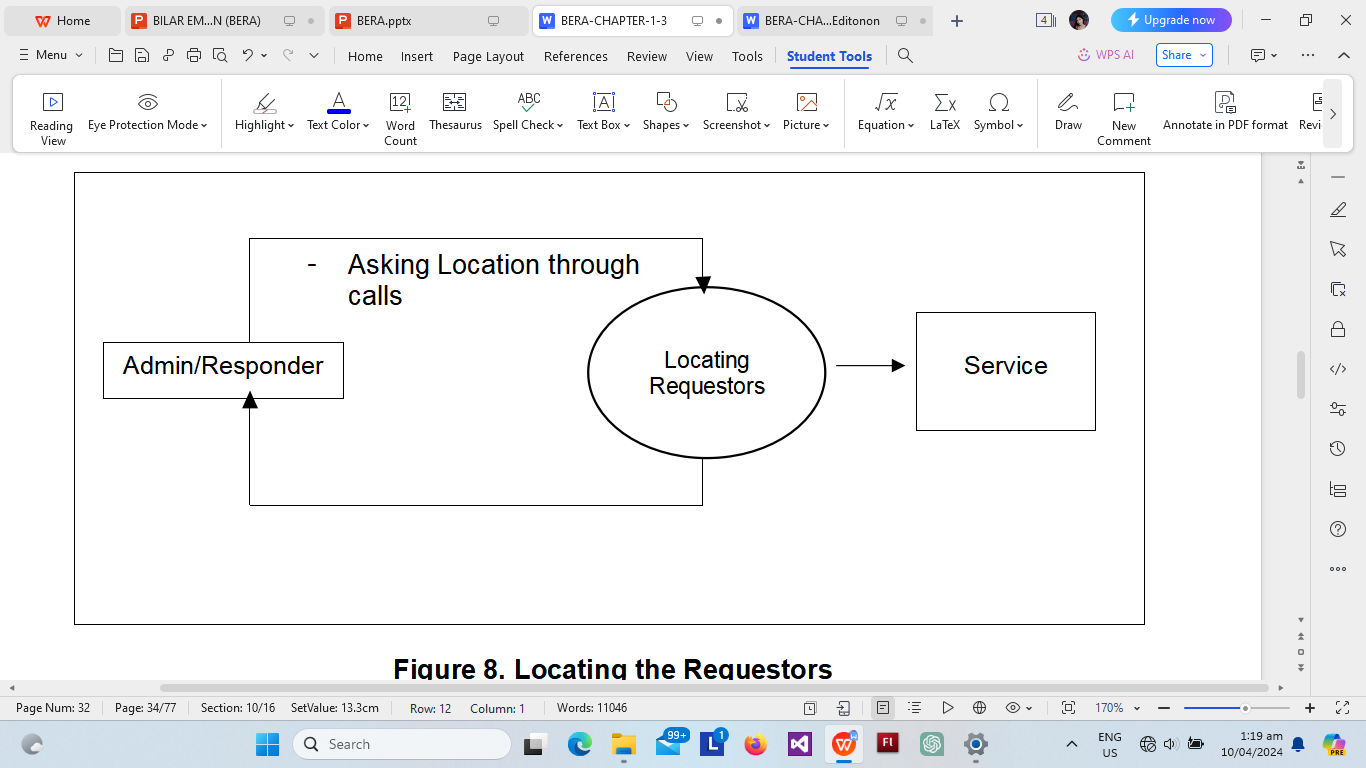
****

Figure 7. Locating the Requestors

Figure 8 show the event diagram of the Generation of Reports of the present system.

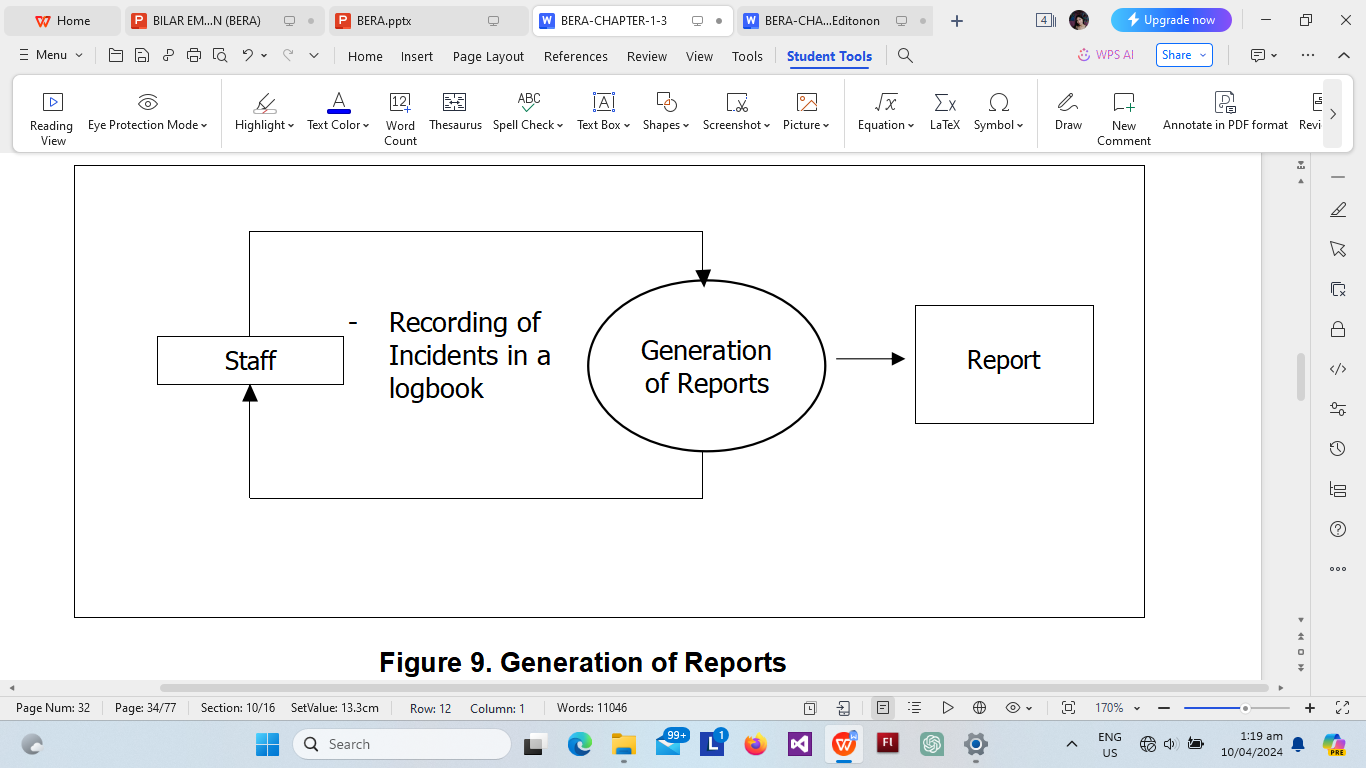


Figure 8. Generation of Reports

### Used Case Diagram

Figure 4 displays the use-case diagram of the BERA application, comprising the actors namely: emergency requestor, emergency message receiver, and emergency responder.

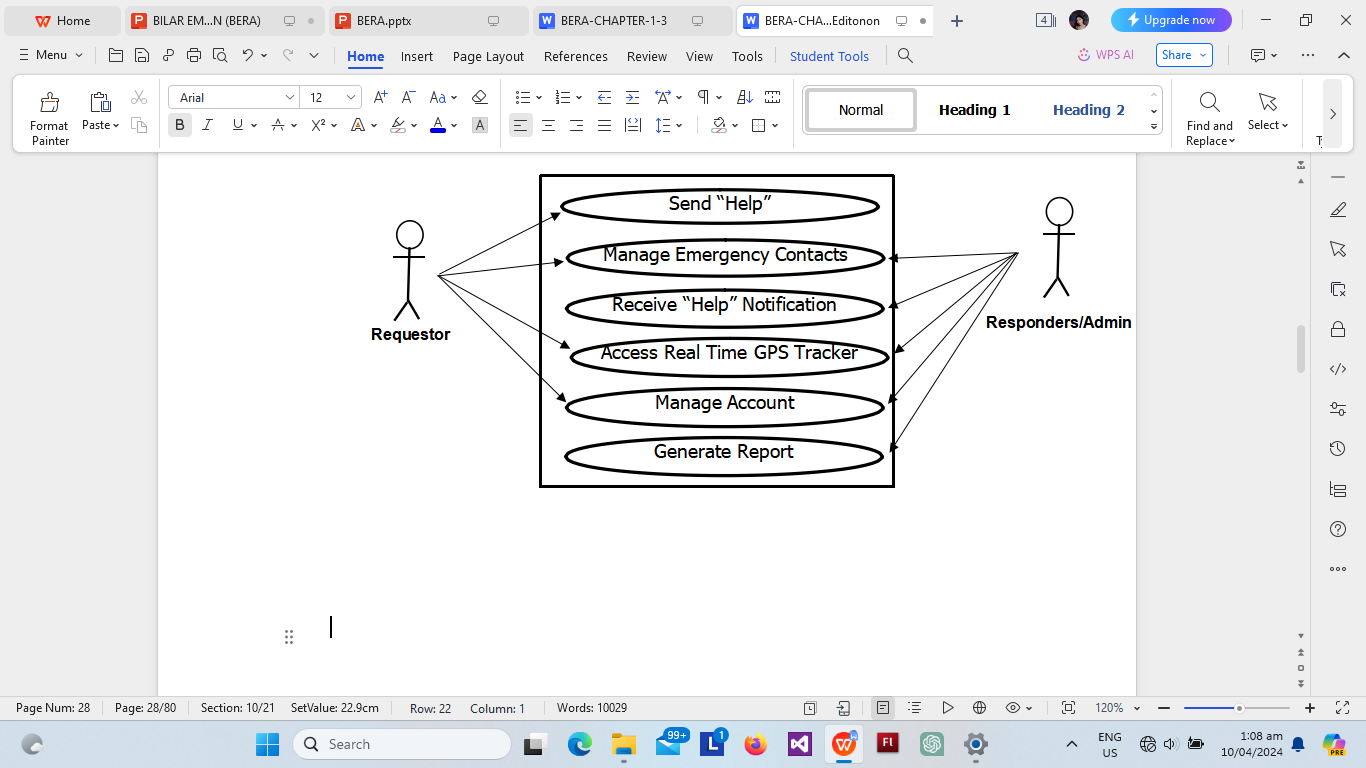
****

Figure 9. Use Case Diagram of Bilar Emergency Response Application

### Use Case Narrative

A use case narrative outlines a scenario that necessitates a comprehensive understanding of the use case framework, illustrating the interaction between the user (actor or use case) to attain a goal with observable value. It should encompass more than a basic sequence of user-to-system interactions, incorporating essential elements. Every use case narrative includes pre-conditions, processes, and post-conditions. Table 5 presents the use-case narrative for accessing the system by logging in.

**Table 3.Send “Help” (Requestor)**

|  |  |
| --- | --- |
| **Use case name** | Send “Help” |
| **Actor** | Requestor |
| **Precondition** | The app is installed on the user's device, and the requestor has a registered account. |
| **Description** | This use case describes the process of both users logging into the app to access emergency services. |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  The user may select multiple Emergency Responder (Medical, Fire and Police) and click “HELP!!!” button. | **Step 2**  Send “Help” Notification to the online and nearby responders and Display |
| **Alternate Paths**   * None | |

Table 4 shows use case narrative for submitting an incident, outlining the user's interactions with the system.

**Table 4.Manage Emergency Call (Requestor)**

|  |  |
| --- | --- |
| **Use case name** | Manage Emergency Call |
| **Actor** | Requestor |
| **Precondition** | The requestor has already logged into the app. |
| **Description** | This use case outlines the process of an emergency requestor call the responders numbers |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  The user clicks the Contact Icon | **Step 2**  Display the Emergency Responders Contact Information |
| **Step 3**  The user selects and call the Responders contact number | **Step 4**  Redirect to mobile phone call setting and tap “Call” |
| **Alternate Paths** | |

Table 5 outlines the use-case narrative for signing up for user roles, explaining the step-by-step process for users to register and take on specific roles within the system**.**

**Table 5.Signing Up - User Registration (Responder/Requestor)**

|  |  |
| --- | --- |
| **Use case name** | User Registration and Role Assignment |
| **Actor** | New User (Emergency Responder or Requestor) |
| **Precondition** | The user has downloaded and installed the application. |
| **Description** | This use case illustrates the process a new user goes through when signing up for the application and selecting their role, either as an Emergency Responder or an Emergency Requestor. |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  The user clicks on the "Sign Up" button and provides essential information such as name, email, phone number, photo, and password. | **Step 2**  After basic registration, the user is prompted to select their role:  Emergency Responder: Individuals interested in providing assistance during emergencies.  Emergency Requestor: Individuals who may need emergency assistance. |
| **Alternate Paths**   * None | |

Table 6 shows the use-case narrative of a responder receiving an alert explains the series of actions and interactions between the system and the responder during the alert reception process.

**Table 6.Receiving Incident Notification and Dispatch (Responder/Admin)**

|  |  |
| --- | --- |
| **Use case name** | Responder and Admin Receives Incident Notification |
| **Actor** | Responder/Admin |
| **Precondition** | The responder or administrator has logged into the application. |
| **Description** | This use case illustrates the process a new user goes through when signing up for the application and selecting their role, either as an Emergency Responder or an Emergency Requestor. |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  Both admin and responder receive an emergency notification. | **Step 2**  Clicks the notification and displays the profile of the patient and their location. |
| **Step 3**  The responder takes necessary actions, such as moving to the incident location. |  |
| **Alternate Paths**   * None | |

Table 7 shows the use-case narrative for managing user accounts in the admin details the process of overseeing and controlling user accounts for both regular users and administrators.

**Table 7.Account Management – (Administrator)**

|  |  |
| --- | --- |
| **Use case name** | Admin Manages User Accounts |
| **Actor** | Admin |
| **Precondition** | The admin is logged into the admin portal of the application. |
| **Description** | This use case outlines the process when an admin manages user accounts within the system. |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  The admin navigates to the user management section of the admin dashboard. | **Step 2**  The admin views a list of all registered users, including responders and requestors. |
| **Step 3**  The admin selects a specific user account for management. | **Step 5**  After making modifications, the admin saves the changes to the user account. |
| **Alternate Paths**   * None | |

Table 8 use-case narrative illustrates the editing of profile information for both requestors and responders.

**Table 8.Editing of Profile Information (Requestor/Responder)**

|  |  |
| --- | --- |
| **Use case name** | Profile Editing |
| **Actor** | Requestor/Responder |
| **Precondition** | User is already logged in and in the account module |
| **Description** | Allow user to view, edit and update account |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  User selects “Profile” | **Step 2**  Displays user info and list of action menu. |
| **Step 3**  User can edit the his/her information | **Step 4**  Displays user info form and save button. |
| **Step 6**  If the user clicks the save button, data stored in the database will be updated and displays a confirmation message.  **Go back to Step 4.** |  |
| **Alternate Paths**  Step 5   * If the form is not filled completely, A form invalidation label will display in each field. | |

Table 9 shows the use case narrative of logging out from the system.

Table 9.Logging Out (All)

|  |  |
| --- | --- |
| **Use case name** | Logging in |
| **Actor** | All |
| **Precondition** | Users are logged in |
| **Description** | Allow users to leave the system |
| **Typical Course of Action** | |
| **Actor Action** | **System Response** |
| **Step 1**  User clicks Profile tab. | **Step 2**  Displays user information with a list of actions. |
| **Step 3**  User presses Sign Out button | **Step 4**  The system will clear cached history of the previously signed in account and redirects to sign in screen. |
| **Alternate Paths**   * None | |

### Database Design

Database design involves creating a data model for the database, encompassing the essential logical and physical design decisions as well as the storage parameters necessary to formulate a comprehensive data design. definition language, which can then be used to create a database.

System design is the procedure of specifying the components, modules, interfaces, and data for a system to meet the specified requirements of the Bilar Search and Rescue Unit. The researchers' objective is to develop a new system tailored for office use. In order to meet the client's needs, various improvements were made to BISARU's existing infrastructure and operational processes.

### Class Diagram

A class diagram in the Unified Modeling Language (UML) is a graphical representation that depicts the structure of a system. It provides an overview of the classes present in the system, their attributes, operations, and the relationships or interactions among them. It provides visual representation of the system’s structure and facilitate communication across stakeholders. Figure 8, refers to the instances depicting the most utmost processes of BERA.

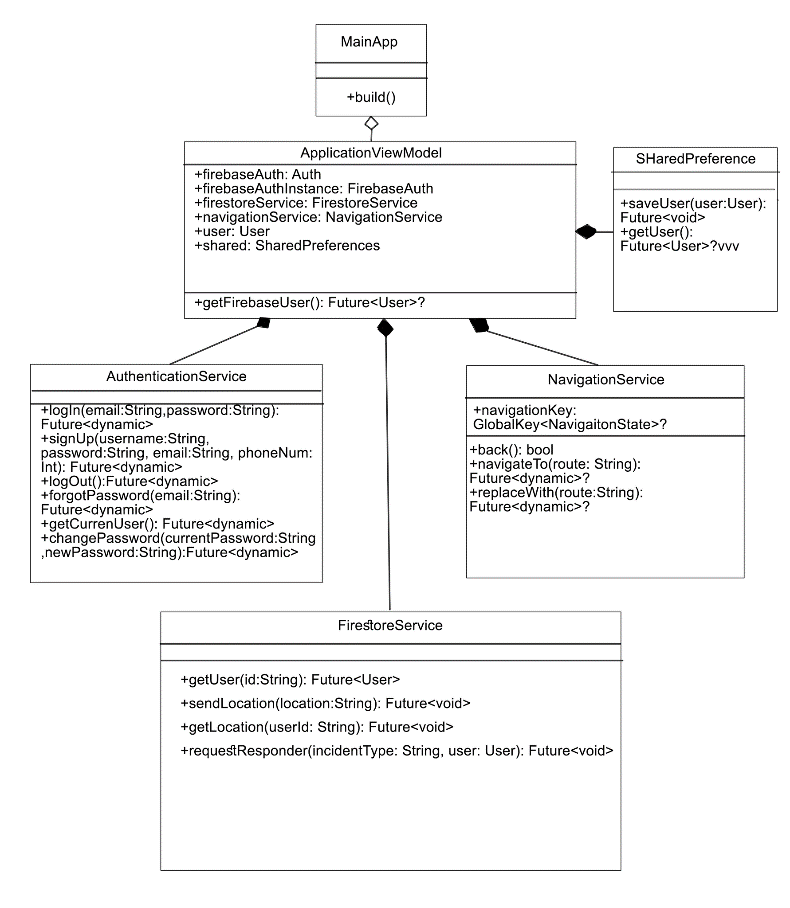


Figure 10.Class Diagram of the BERA

### Data Structure

The following tables below were the database tables that were used in storing the information that was inputted in the system together with a collection of requirements that facilitate searching, sorting, and similar activities. It is a particular way of organizing data on a computer so that it can be used effectively. Table 10 shows the data structure for user account credential.

User Account Credential

**Table 10.User Account Credential**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field No.** | **Field Name** | **Type** | **Description** |
| 1 | uid | String | User Id |
| 2 | email | String | Email |
| 3 | password | Hash | password |
| 4 | phone number | Int | phone |
|  | concerns | String | concern |
|  | fcmToken | String | Firebase token |
|  | image | Image | profile |
|  | latitude | Int | latitude |
|  | longitude | Int | longitude |
|  | situation | String | situation |
|  | situationPhoto | Image | Incident photo |
|  | timestamp | Timestamp | date and time |

Table 11 shows the data structure for system users. It is organized in a way that is efficient to index and to retrieve.

User Information

**Table 11. System User**

|  |  |  |  |
| --- | --- | --- | --- |
| **Field No.** | **Field Name** | **Type** | **Description** |
| 1 | uid | String | User Id |
| 2 | firstname | String | First Name Middle |
| 3 | middlename | String | Name |
| 4 | lastname | String | Last Name |
| 5 | email | String | Email |
| 6 | image | String | Image Uri |
| 7 | phoneNum | String | User Contact No. |

Table 12 shows the data structure for incident information.

Incident Information

|  |  |  |  |
| --- | --- | --- | --- |
| **Field No.** | **Field Name** | **Type** | **Description** |
| 1 | uid | String | User id |
| 2 | incident | String | Type of incident |
| 3 | location | String | location |
| 4 | username | String | Username |
| 5 | phoneNum | String | User Contact No. |

**Table 12. Incident Information**

### 

**Program Hierarchy**

A program hierarchy is a diagram that displays the system's breakdown into its most basic, controllable layers. Each Module is symbolized as a box that holds the Module within. The high-level design or architecture of a computer program is described using a program hierarchy. Figure 12 show the program hierarchy of the BERA.

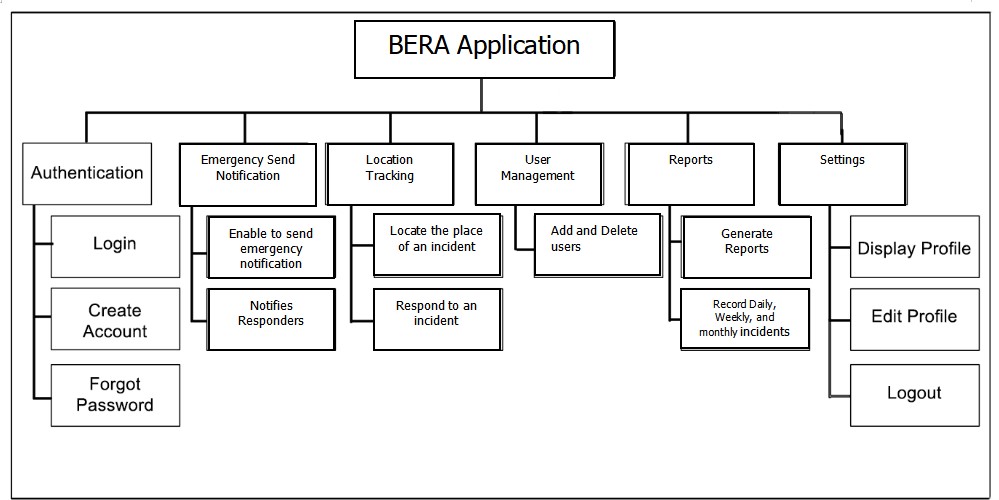


Figure 11.Program Hierarchy

### Functional Requirements

The function of a software system or component is specified by its functional requirements. Three components make up a function: inputs, behavior, and outputs. Functional requirements include processes like calculations, technical specifications, data processing and manipulation, and other particular functionality that outlines what a system is intended to do. A prototype was used to assist determine the functional requirements. A functioning prototype will be developed in close cooperation with the staff numbers of BISARU and a few of its clients in order to strategically identify functional requirements. Also, functionalities and modules will be based upon the existing standard operating procedures of Bohol Veterinary. The resulting functionalities are as follows:

**Process Log in**

FREQ 1: Access to the system should allow authorized Staff only to login

FREQ 2: The system should limit module accessibility corresponding to their access rules and privileges.

FREQ 3: All information gathered should be saved and secured

**Process Requestor Identification**

FREQ 4: The system should allow the recording of client’s information.

FREQ 5: All information gathered should be saved and secured

**Process Sending Incident Information**

FREQ 6: The system should send incident information.

FREQ 7: All information gathered should be validated by the admin

**Process of Location Tracking**

FREQ 8: The system should send location to the responders

FREQ 9: The system should identify the location of the incident.

**Generation of Reports**

FREQ 10: The system should allow the personnel in-charge to search, view and print reports.

FREQ 11: The system should allow the personnel in-charge to generate daily, weekly, and monthly incidents.

**User Account Management**

FREQ 12: The system should automatically display save user account.

FREQ 13: The system should allow its users to modify corresponding account. **Log out**

FREQ 14: The system should execute log-out 3

### Non-Functional Requirement

A non-functional requirement specifies criteria that can be used to evaluate a system’s functioning rather than specific actions. Functional requirements, on 54 the other hand, describe specific behaviors or functions. The system design will include a plan for implementing functional requirements.

1. The system should be implemented with internet connection.

2. Classified modules must run on mobile device.

### Test Cases

A test case is a comprehensive procedure that thoroughly tests a feature or aspect of a feature under which a tester will determine whether or not an application or software system is operating as intended. It also consists of a set of input values, evolution preconditions, results, and executions created for a specific objective or test condition. A variety of scenarios were tested during the acceptance testing. The test cases approach is used to assess the suggested system by letting users use it as long as they follow the guidelines. To be deemed successful, the system needs to deliver the intended outcome in every test scenario. The system's user manual is contained in this test case.

### Test Case 1:

Module: User Registration And Log-in Severity: 1

Instructions:

1. User account registration
2. Input email and password
3. Account set up

Expected Result:

* User can access the controlled modules of the system depending on user role and privileges.

### Test Case 2:

Module: Adding a User Profile Severity :1

Instructions:

1. On the main menu, select Profile Tab.
2. Input the user’s photo.
3. Click “save “button for confirmation

Expected Result

* It should be successful saved.
* The newly added profile photo should be displayed.

### Test Case 3:

Module: Selecting an Emergency Concern Severity 1:

Instructions:

1. On Selecting a Emergency Concern select at least 1 Fire Emergency, Medical Emergency and Police Emergency,

1. On the home screen, select either “Medical”, “Police”, and “Fire”.
2. Press the help button.

2. Describe what is the situation.

3. Add a photo of the incident.

4. Submit your inputs

Expected Result:

* It should be sent successfully.
* The responder will receive a notification

### Test Case 4:

Module: Showing the places of responders

Severity: 1

Instruction:

1. Click the map page
2. Requestor can see the places of responders.
3. Click the mark to show the name of responder.

Expected Result:

* Requestors will know the location of the responders

### Test Case 5:

Module: Call Emergency Numbers in Offline

Severity: 1

Instructions:

1. On the contacts page the hotline numbers are listed
2. Select a number to call.
3. Click the call button.

Expected Result:

* Dialing the hotline numbers should function.

**Test Case 6:**

Module: Editing the Personal Information

Severity: 1

Instructions:

1. On the profile page you can edit the details of the account.
2. Click edit button.
3. Input detailed information.
4. Click “Save” for confirmation.

Expected Result:

* It should be successfully saved.
* The newly added information should add to the database

**Test Case 7:**

Module: Account Recovery

Severity: 1

Instructions:

1. In the login screen, select “Forgot Password”.
2. Input the email
3. Click “Send Password reset link”.

Expected Results:

* The client should receive an email for a password reset link.

### Test Case 8:

Module: Receiving Emergency and Respond

Severity: 1

Instructions:

1. In the main screen of the responder wait for an emergency
2. Notification will be received if someone needed help.
3. Dialog box pops up and select call or navigate.
4. Click call to dial the number of requestors.
5. Click navigate to navigate the location of requestor in google maps.
6. Requestors will be notified

Expected Results:

* Requestor’s number should be dialed successfully saved.
* Responders can pinpoint the location of the requestors

### Test Case 9:

Module: Account Management

Severity: 1

Instructions:

1. In the main admin screen, select “Account” page.
2. Click “Manage Requestor Account” tab.
3. Delete requestor account.

Expected Results:

* It should be successfully saved.
* Requestor and responder account should be edited or delete

### Technical Requirements

Smartphones have evolved into essential tools for communication, productivity, and entertainment. To ensure optimal performance, both hardware and software components must be carefully considered. The System-on-Chip (SoC) serves as the smartphone's CPU, incorporating components like the processor and GPU, crucial for smooth multitasking and prolonged battery life. Random-Access Memory (RAM) provides quick data access, enhancing user experience, especially for resource-intensive tasks. The display quality, battery capacity, and camera system significantly influence user satisfaction, requiring considerations such as resolution, battery efficiency, and camera features like optical image stabilization and HDR. Connectivity options like Wi-Fi, Bluetooth, and GPS enable seamless internet browsing and location-based services. Storage capacity, including support for expandable storage, ensures users can store large amounts of data without compromise. Software optimization, including OS updates and collaboration with app developers, enhances system stability and performance. Overall, integrating these components effectively enables smartphone manufacturers to meet diverse user needs while fostering innovation in the mobile industry.

### Minimum Hardware Specifications

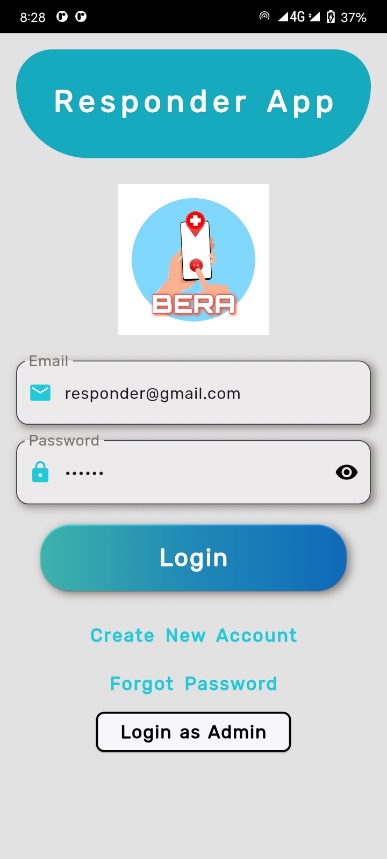
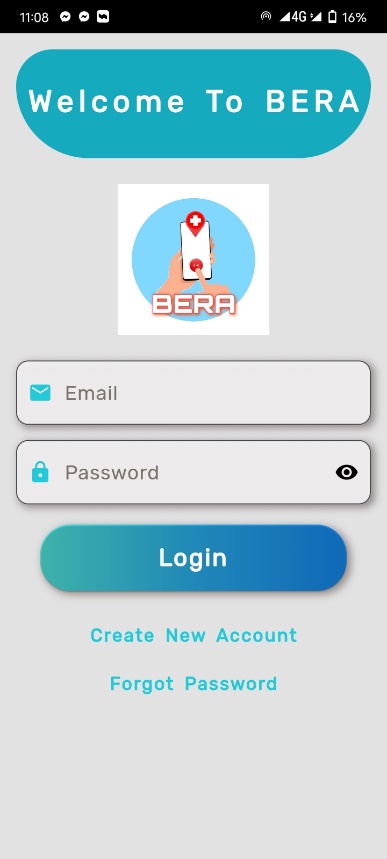
Table 13 shows the minimum android hardware specifications need by the system. This covers the minimum hardware requirements for the system to work as intended and expected. These specifications were chosen based on what is already available on the market and what most mobile phone systems provide.

**Table 13.Minimum Android Hardware Specifications**

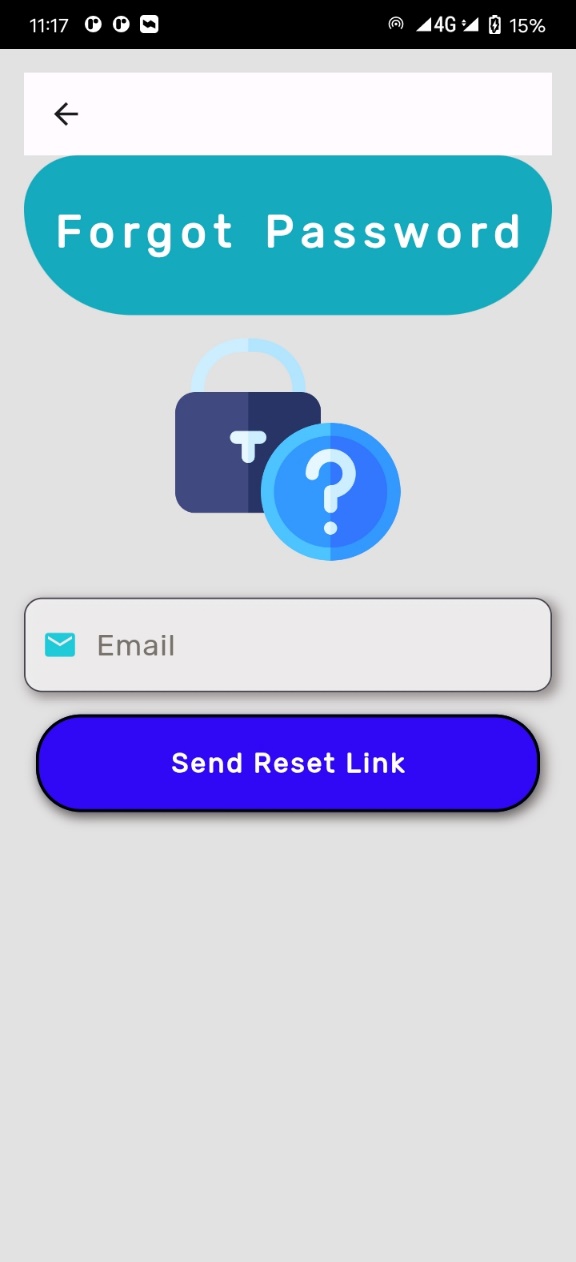
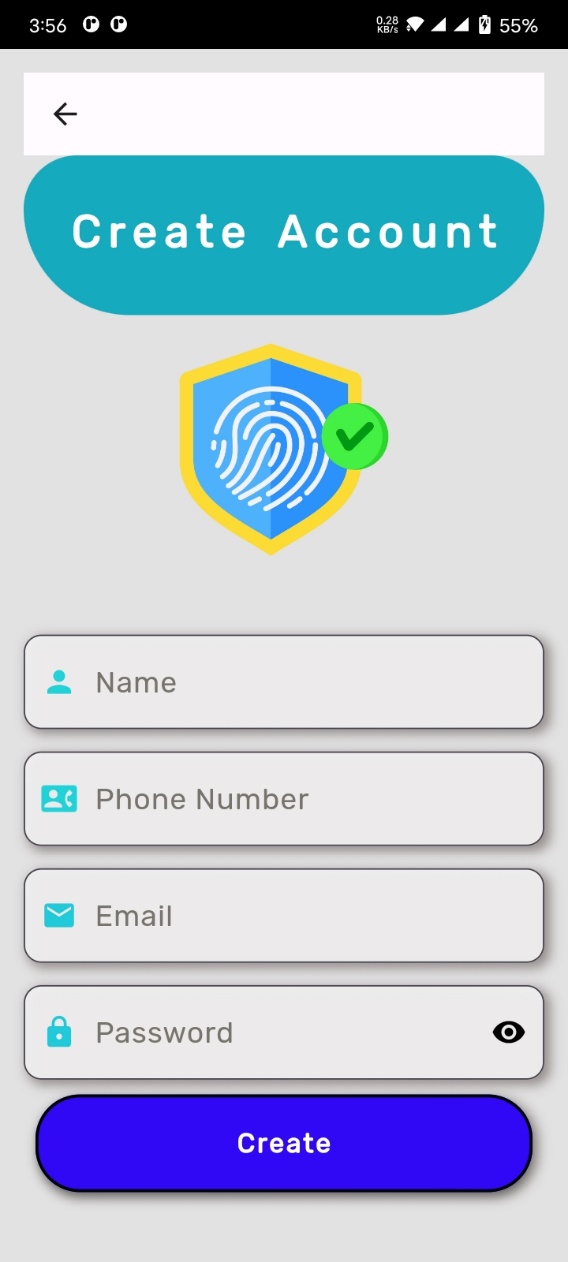
|  |  |
| --- | --- |
| **COMPONENT** | **SPECIFICATION** |
| CPU | DUAL CORE AND UP |
| RAM | 2 GB |
| ROM | 4GB |
| CONNECTIVITY | WIFI or LTE |

### Screen Layout

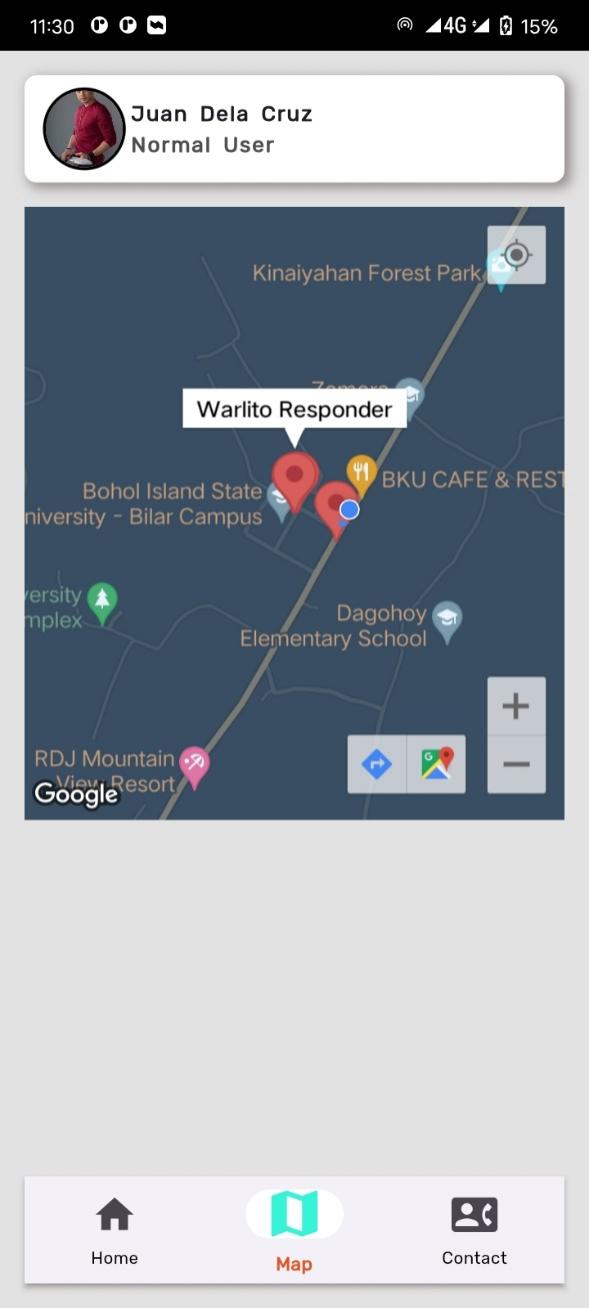
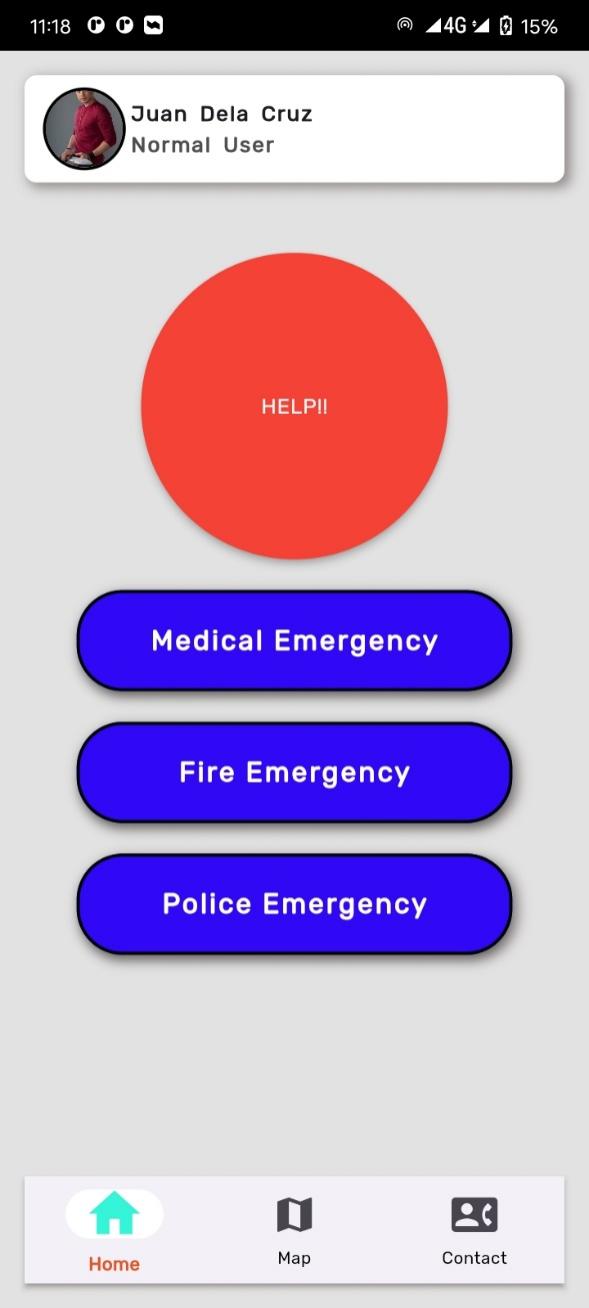
The graphical user interface's design is explained by the screen layout. It should be separated from the graphical user interface's features and encompasses a range of applications where screens can be utilized in human-machine interaction. One of the many aspects of the system's user-friendliness is the screen layout. It should be made so that users can quickly and easily navigate the system and that the tasks they need to complete are clearly identified.

****

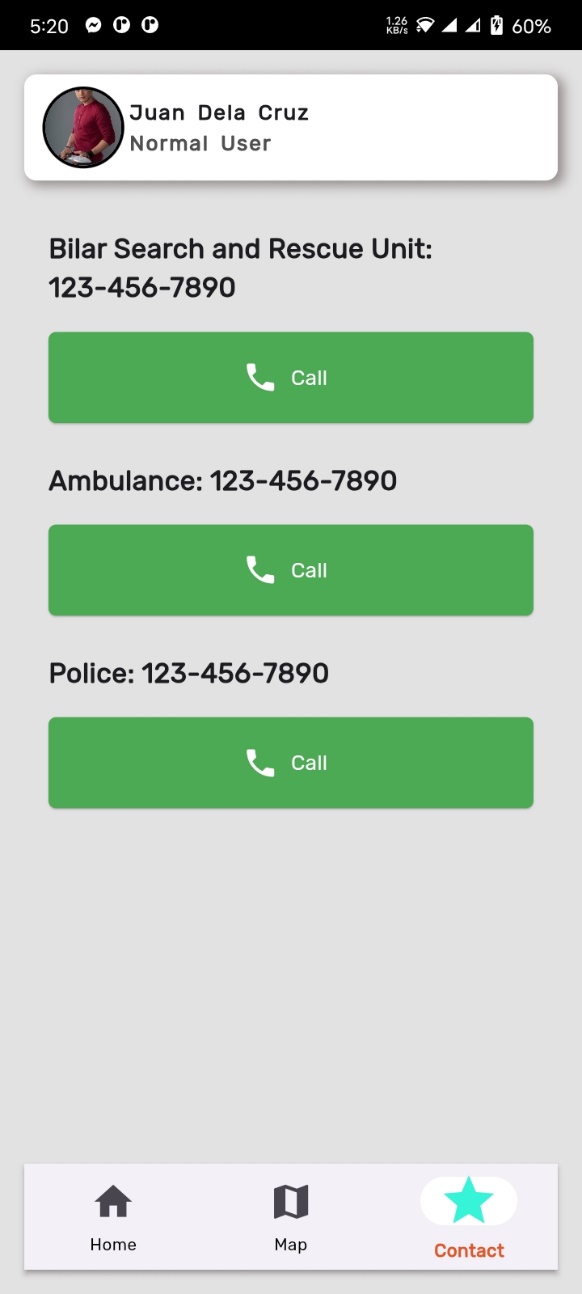
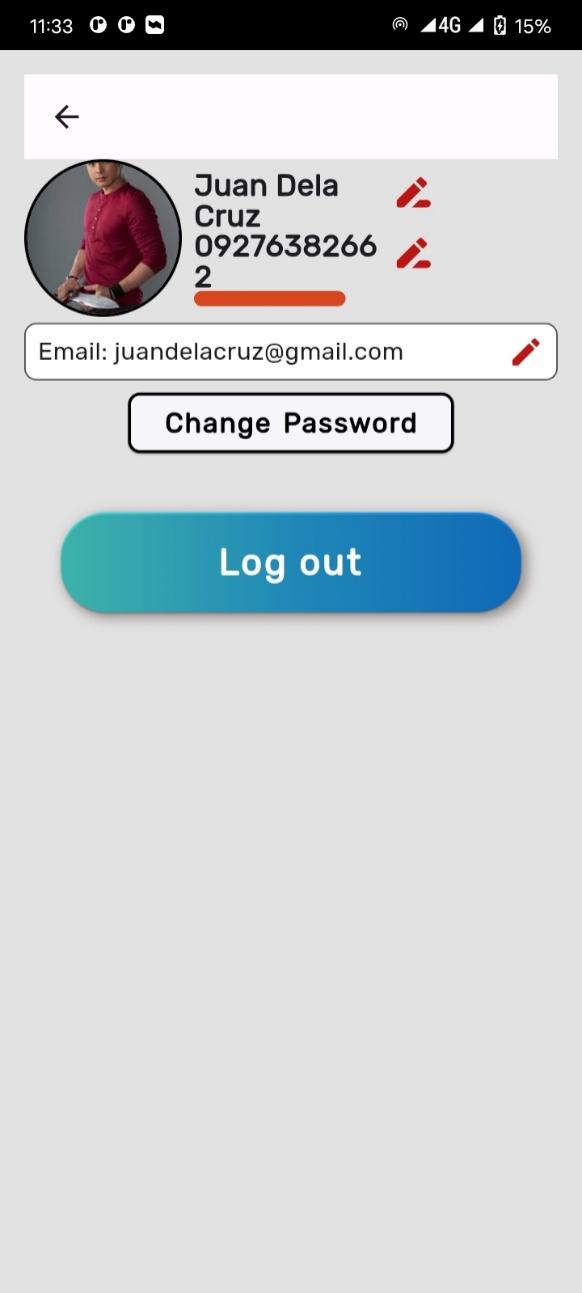
Preview 1: Splash Screen & Login Page



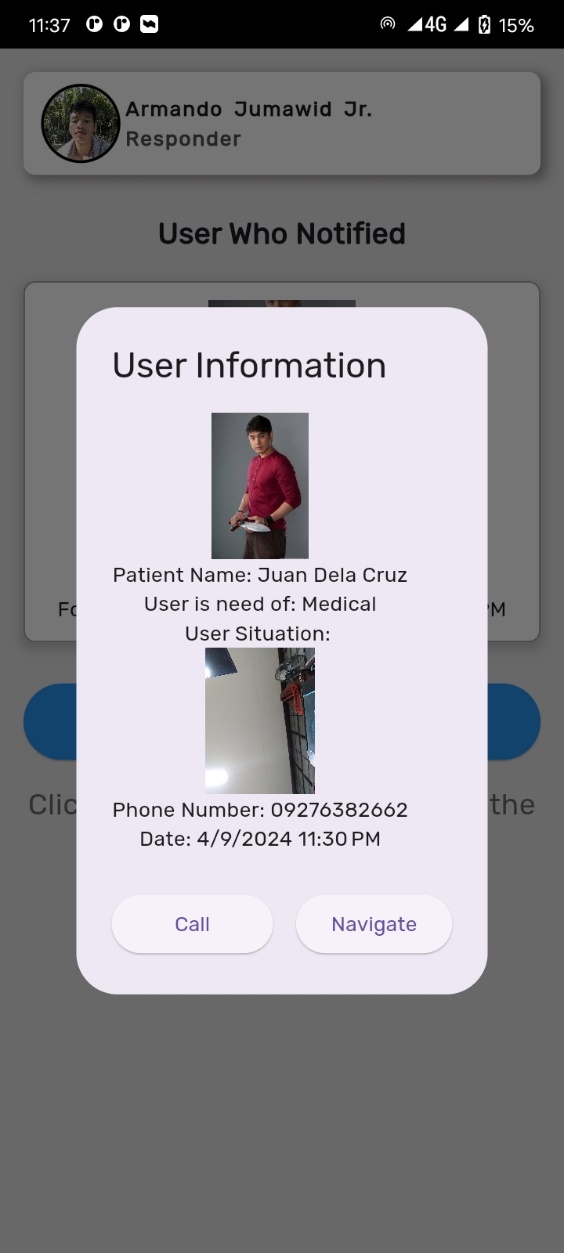
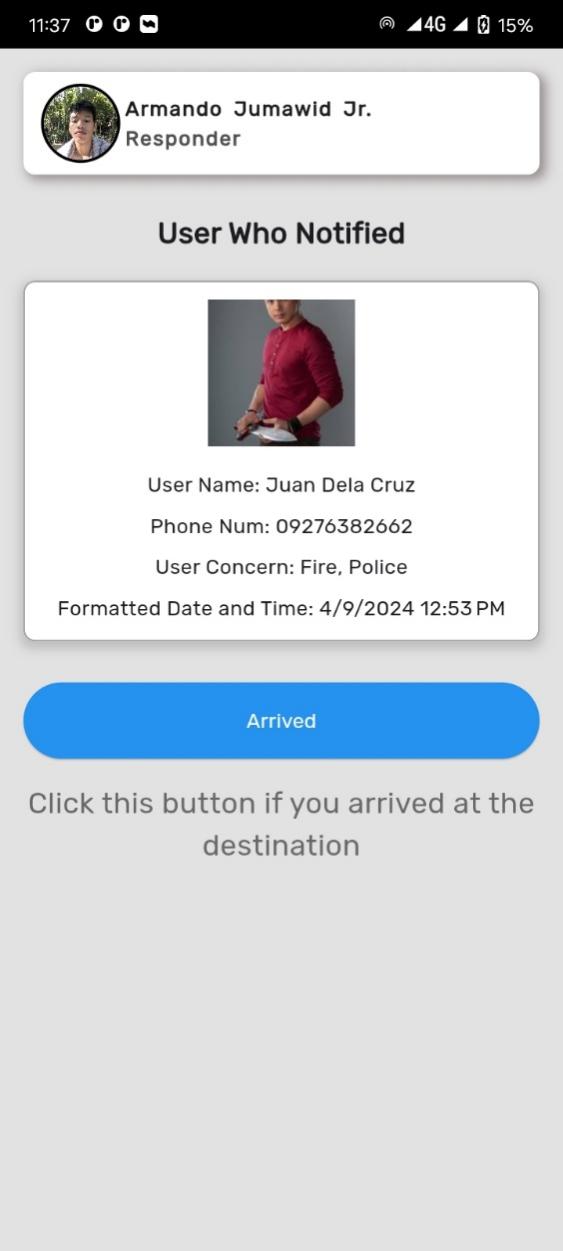
Preview 2: Sign Up Page Preview 3: Forgot Password Page

****

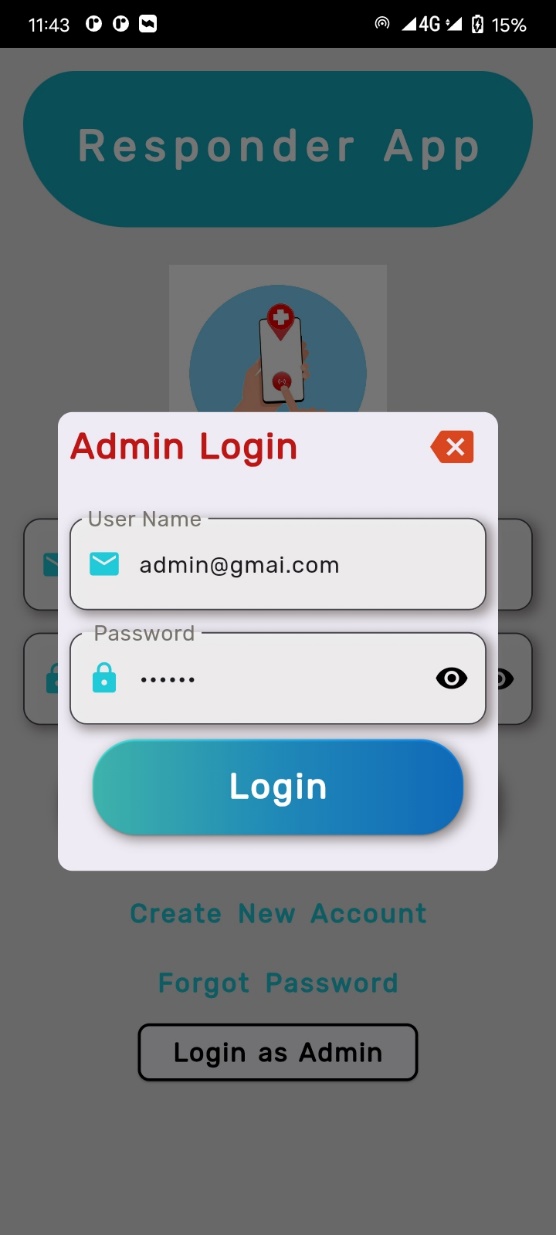
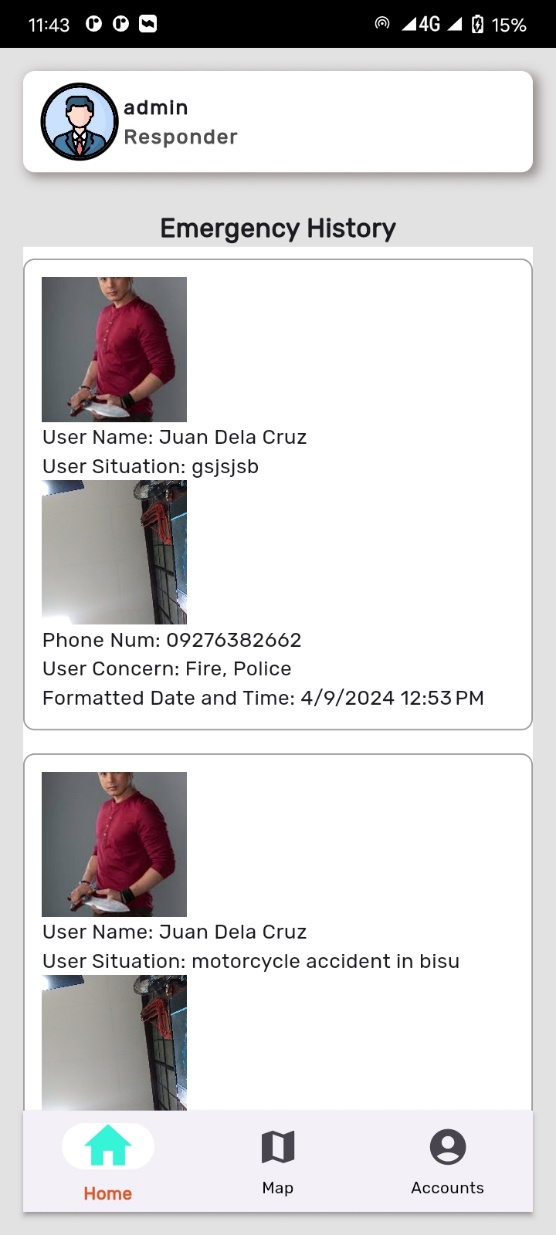
Preview 4: Requestor Home Screen Preview 5: Map Page



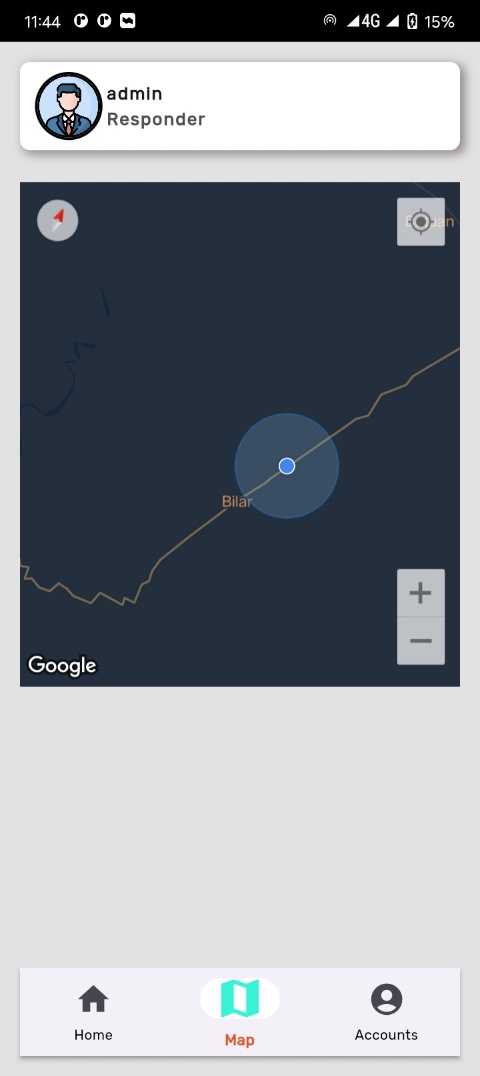
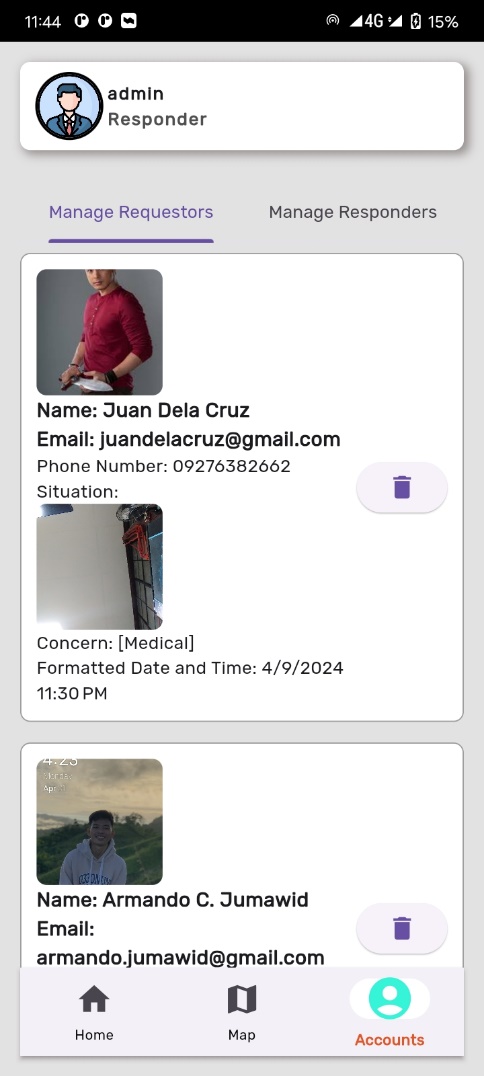
Preview 6: Contact Page Preview 7: Profile Page



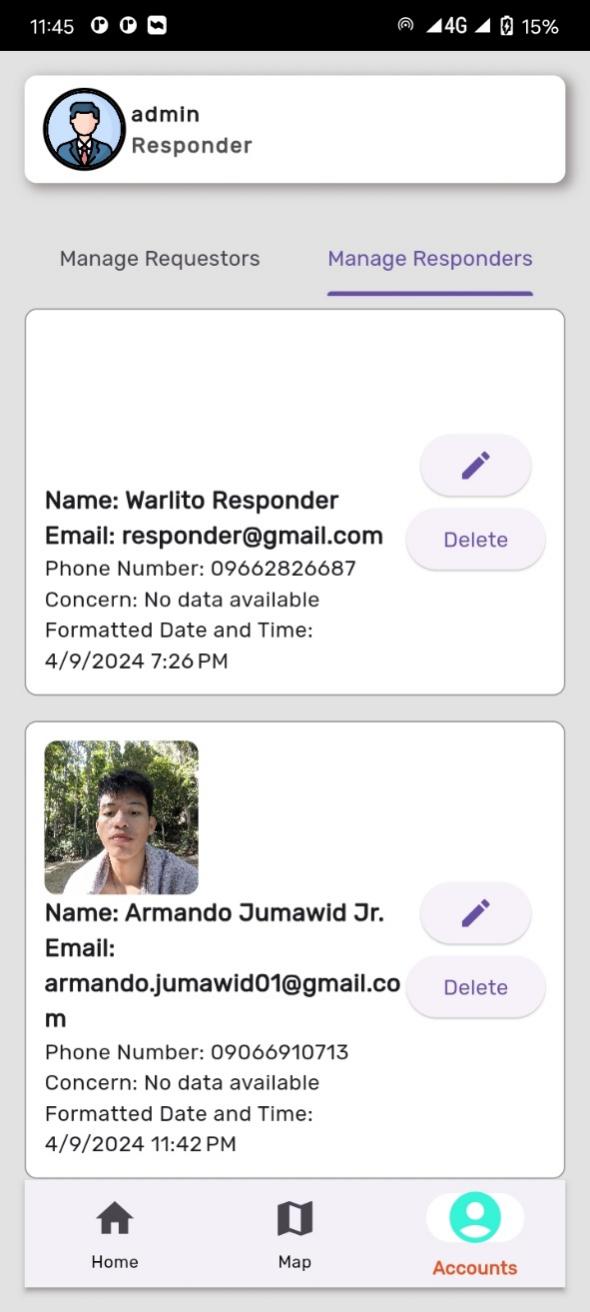
Preview 8: Incident Dialog Preview 9: Responder Homepage



Preview 10: Admin Login Preview 11: Emergency History



Preview 12: Admin Map Preview 13: Manage Requestors



Preview 14: Manage Responders

### Economic Performance Evaluation

Table 19 shows the initial investment and annual operating costs of BERA Application. BERA economic performance was assessed in terms of initial investment. The initial investment was the amount required by the client prior to the start of the system’s operation and deployment.

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Item** | **Qty** | **Unit** | **Unit Price** | **Total** |
| 1. **Initial Investment** | | | | |
| 1. **Hardware** | | | | |
| Android Mobile Phone | 5 | Pieces | ₱ 6,000.00 | (Existing) |
| Wifi Modem | 1 | Pieces | ₱ 5,000.00 | (Existing) |
| 1. **Software** | | | | |
| Internet | 12 | Month | ₱ 1,500.00 | ₱ 18,000.00 |
| **Total Initial Investment Cost** | | | | **₱ 18,000.00** |
| 1. **General Services** | | | | |
| System Maintenance | 4 | Quarter | ₱ 1,500.00 | ₱ 6,000.00 |
| **Sub-Total General Services Cost** | | | | **₱ 6,000.00** |
| **Total** | | | | ₱ 24,000.00 |

**Table 14.Initial Investment and Annual Operating Costs**

### Testing and Evaluation

Testing evaluation refers to the process of assessing the effectiveness, accuracy, and comprehension of software testing activities. Testing evaluation helps it determine if the software meets the specified requirements and performs as expected. It involves analyzing test results, identifying defects, tracking defects to resolution, and generating reports to provide insights into the overall quality of the software.

### System Usability

Table 20 shows below the based on the results of the system usability test using the questionnaire prepared by ISO 25010, the target users perceived the system to be acceptable with a general rating of "Strongly Agree". The respondents stated that the system was suitable for recording student data and met their expectations in terms of functions and capabilities. There may be room for improvement but overall, the system performed well in the test.

**Table 15.System Usability Assessment Result N=10**

|  |  |  |  |
| --- | --- | --- | --- |
| **Criteria for System Usability** | | **Weighted**  **Mean** | **Rating**  **Strongly Agree** |
| **Functionality Suitability** | |  |  |
|  | Functional completeness | 4.8 | Strongly Agree |
|  | Functional correctness | 5 | Strongly Agree |
|  | Functional appropriateness | 4.6 | Strongly Agree |
| **Performance Efficiency** | |  |  |
|  | Time behavior | 4.8 | Strongly Agree |
|  | Resource utilization | 4.6 | Strongly Agree |
|  | Resource utilization | 4.6 | Strongly Agree |
| **Compatibility** | |  |  |
|  | Co-existence | 4.7 | Strongly Agree |
|  | Interoperability | 4.5 | Agree |
| **Usability** | |  |  |
|  | Appropriateness Recognizability | 4.9 | Strongly Agree |
|  | Learnability | 4.6 | Strongly Agree |
|  | Operability | 4.5 | Agree |
|  | User error protection | 4.3 | Agree |
|  | User interface aesthetics | 4.7 | Strongly Agree |
|  | Accessibility | 4.5 | Agree |
| **Reliability** | |  |  |
|  | Maturity | 4.7 | Strongly Agree |
|  | Availability | 4.7 | Strongly Agree |
|  | Recoverability | 4.8 | Strongly Agree |
| **Security** | |  |  |
|  | Confidentiality | 4.7 | Strongly Agree |
|  | Integrity | 4.9 | Strongly Agree |
|  | Accountability | 4.7 | Strongly Agree |
|  | Authenticity | 4.8 | Strongly Agree |
|  | Non-repudiation | 4.6 | Strongly Agree |
| **Maintainability** | |  |  |
|  | Modularity | 4.8 | Strongly Agree |
|  | Reusability | 4.8 | Strongly Agree |
|  | Analyzability | 4.5 | Agree |
|  | Modifiability | 4.5 | Agree |
|  | Testability | 4.5 | Agree |
| **Portability** | | |  |
|  | Adaptability | 4.7 | Strongly Agree |
|  | Installability | 4.5 | Agree |
|  | Replaceability | 4.6 | Strongly Agree |
|  |  |  |  |
|  | **Overall Mean** | 4.66 | **Strongly Agree** |

**Chapter 3**

**SUMMARY OF FINDINGS, CONCLUSION, AND RECOMMENDATION**

### Summary of Findings

The Bilar Search and Rescue Unit (BISARU) was observed, and it was known that they still respond to emergencies by using a manual dispatch system. When a dynamic clustering protocol is incorporated into the application, it intelligently groups nearby requests to improve response times and dispatch emergency response teams more efficiently. To address the issues found in BISARU, the researchers planned to create an application called "Bilar Emergency Response Application using Dynamic Clustering Protocol of BISARU in Bilar, Bohol.". The user's information, location tracking, and notifications are all handled by this mobile application.

Based on the identified needs of BISARU, the “Bilar Emergency Response Application using Dynamic Clustering Protocol” was developed with the modules notification, location tracking, account management and reports. The developed system was pilot tested and evaluated in terms of system usability. Based from the results of the survey, the respondents gave a general rating of with an interpretation of “strongly agree”. The result generally indicated that the application was usable in handling user’s information. Specifically, the expectations of the respondents as to functions and capabilities had been achieved by the application. Moreover, the system has clear organization of information, the content was comprehensive, with good user interface and ease of use.

### Conclusion

The study's results led the developers to the conclusion that BISARU's manual emergency response and dispatch method is inefficient and lacks a system for managing user information.

The "BILAR EMERGENCY RESPONSE APPLICATION USING DYNAMIC CLUSTERING PROTOCOL" was presented as a remedy for these problems. The system was designed to increase BISARU's emergency response effectiveness. The study's findings showed that BERA accomplished the goals and fulfilled the client's expectations by successfully improving the emergency response.

Furthermore, BERA has low running costs and little investment requirements. This suggests that BISARU can afford to implement the system and that it is economically feasible. The system becomes a financially viable option for BISARU by reducing the costs related to implementation and maintenance.

The study concluded, in essence, that BISARU's emergency response and dispatch was inadequate and depended on ineffective manual procedures. However, the introduction of the BERA application met BISARU's expectations and requirements by improving user management efficiency and response time through location tracking. Furthermore, because of its small initial investment and low running costs, the application was regarded as economical and affordable.

### Recommendations

Based from the observations during implementation and on the aforementioned conclusions, the developers have recommended the following for efficient record management, smooth system adaptation and operation for the future development.

* It is recommended that Bilar Search and Rescue Unit (BISARU) should adopt the new application BERA, to improve their response to an emergency.
* It is advised to hold an orientation and training seminar for the target users to guarantee a seamless transition to the new system. The purpose of this seminar is to familiarize users with BERA's functionality. Staff members can become proficient users of the application and maximize its features with training and orientation, which will improve emergency response and dispatch.
* Considering the diverse needs and preferences of users, it is recommended to develop web and desktop versions of BERA alongside the mobile application. This multi-platform approach ensures accessibility for users who may prefer or require alternative devices for accessing emergency response services. Moreover, it expands the reach of BERA beyond mobile devices, accommodating users who may not have access to smartphones or prefer using computers for emergency communication.

**REFERENCES**

## REFERENCES

Agrawal, S.A.; and Chavan, S.B. (2014). "EMS: An Android Application for Emergency Patients." \*International Journal of Computer Science and Information Technologies (IJCSIT)\*, 5(4), 5536-5538.

Aher, R.; Baviskar, P.; Chaudhari, P.; and Mahale, M. (2016). "Emergency Notification Services Application Design for Mobile Devices." \*International Journal of Engineering Science and Computing\*, 6(11), 3501-3502.

Bolle, S.; Hasould, P.; and Heuriksen, E. (2011). "Video Calls from Lay Bystanders to Dispatch Centres - Risk Assessment of Information Security." \*BMC Health Services Research\*, 11: 244, 1-7.

De Guzman, J.B.; De Guzman, R.C.C.; and Ado, R.G. (2014). "Mobile Emergency Response Application Using Geolocation for Command Centres." \*International Journal of Computer and Communication Engineering\*, 3(4), 235-238.

Drury, J., & Tekin Guven, S. (2020). "Emergencies and Disasters." In Together Apart:

The Psychology of COVID-19, 80-83.

Erkhembayar, R., Dickinson, E., Badarch, D., Narula, I., Warburton, D., Thomas, G. N., & Manaseki-Holland, S. (2020). "Early Policy Actions and Emergency Response to the COVID-19 Pandemic in Mongolia: Experiences and Challenges." The Lancet Global Health, 8(9), e1234-e1241.

Errikson E., Pemker M. (2000). Business Modeling with UML: Business Patterns at Work.

Fabito B.S.; Balahadia, F.F.; and Cabatlao, J.D.; (2016). "AppLERT: A Mobile Application for Incident and Disaster Notification for Metro Manila." \*2016 IEEE Region 10 Symposium (TENSYMP)\*. Bali, Indonesia, 288-292.

Hong, N.K.; Hafit, H.; Wahid, N.; Kasim, S.; and Yulof, M.M. (2017). "Help Me Please! Designing and Developing Application for Emergencies." \*IOP Conference Series: Materials Science and Engineering\*, 226 012116.

Ineya, S.S.; Ramya, R.; and Dinesh, P.S. (2017). "Emergency Reporting Using Smartphone." \*International Journal of Computer Science Engineering\*, 6(6), 140-144.

Jadhav, R.; Patel, J.; Jain, D.; and Phad-htare, S. (2014). "Emergency Management System Using Android Application." \*International Journal of Advanced Research in Computer Engineering & Technology\*, 5(3), 2803-2805.

Khan, Z. (2014). "An Emergency Alert SMS and GPS Tracking Application for Android Smart Phones." \*International Journal of Advanced Research in Computer Science\*, 5(3), 89-94.

Manoj, B. S., & Baker, A. H. (2007). "Communication Challenges in Emergency Response." Communications of the ACM, 50(3), 51-53.

Na, S., Xumin, L., & Yong, G. (2010, April). Research on k-means clustering algorithm: An improved k-means clustering algorithm. In 2010 Third International Symposium on intelligent information technology and security informatics (pp. 63-67). Ieee.

Pinheiro, A. S., Gouveia, R., Jesus, Â., Baptista, J. S., & Baptista, J. S. (2019). The importance of emergency response training: a case study. In *Studies in systems, decision and control*. https://doi.org/10.1007/978-3-030-14730-3\_79

Ramirez, L.P.; Pondoyo, J.M; Laya, T.P.; and Bahio, N. (2013). "Sagip Pinoy: A Web and Mobile-Based Philippine Emergency Quick Response Portal." Retrieved October 1, 2021, from [link](https://docplayer.net/26227204-Sagippinoy-a- web-and-mobile-based-philippine-emergency-quick-response-portal.html).

Raut, D.B.; and Patil, P. (2015). "Research on Emergency Call and Location Tracking System with Enhanced Functionality for Android." \*Internal Journal of Advance Research in Computer Science and Management Studies\*, 3(5), 2321-7782.

Romano, M.; Onorati, T.; Aedo, I.; and Diaz, P. (2016). "Designing Mobile Applications for Emergency Response: Citizens Acting as Human Sensors." \*Sensors\*, 16(3), 406.

Steurer, K. M. (2018). Crime Around the World: Using Mathematical Modeling

Techniques to Model Aggregated Worldwide Crime Rates.

Whitzman, C., et al. (2013). Building Inclusive Cities Women's safety and the right to

the city, Routledge.

Xu, R., & Wunsch, D. (2005). Survey of clustering algorithms. IEEE Transactions on neural networks, 16(3), 645-678.

Xu, D., & Tian, Y. (2015). A comprehensive survey of clustering algorithms. Annals of

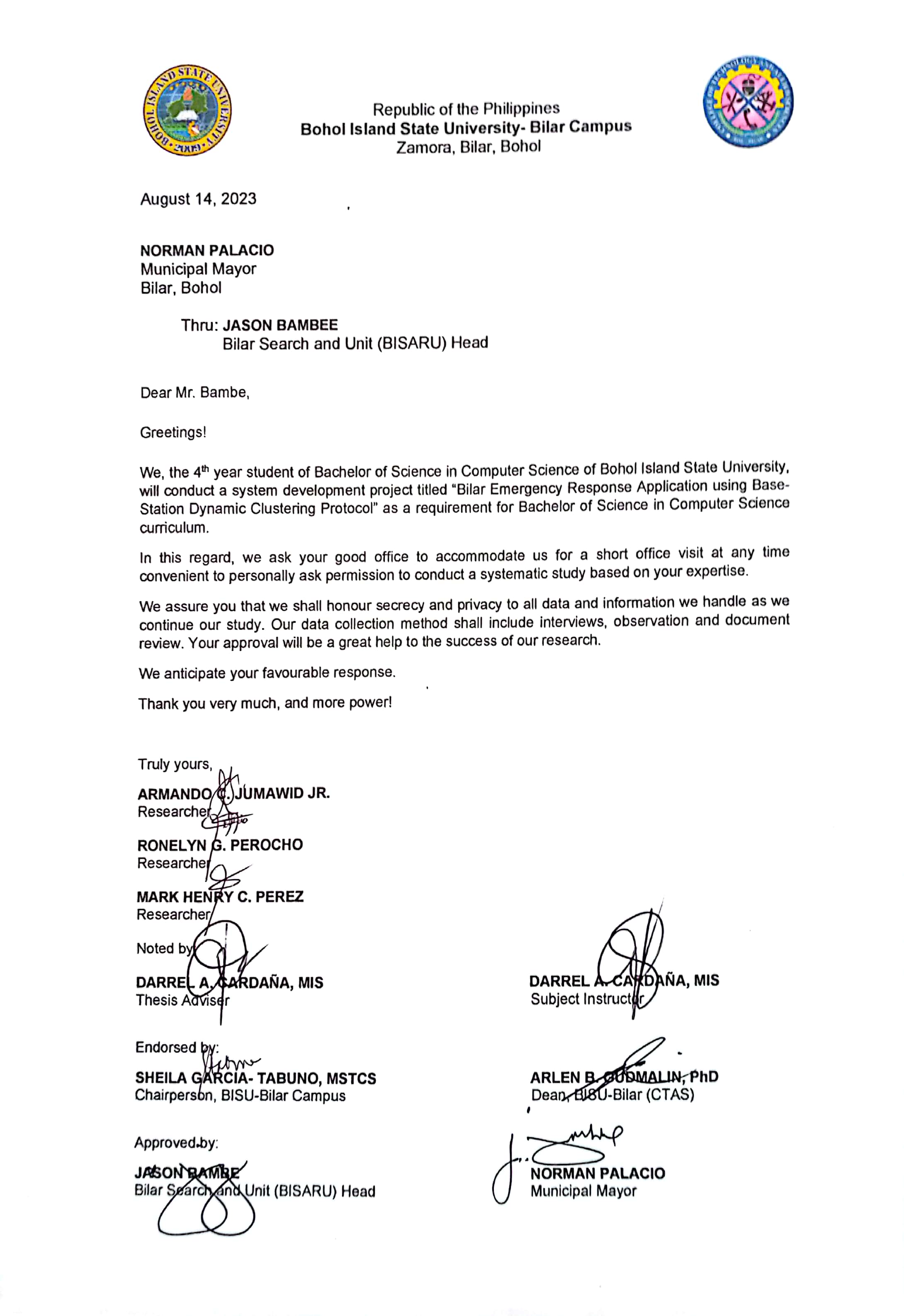
Data Science, 2, 165-193.

Zlatanova, S. and J. Li (2008). Geospatial Information Technology for Emergency Response

**APPENDICES**

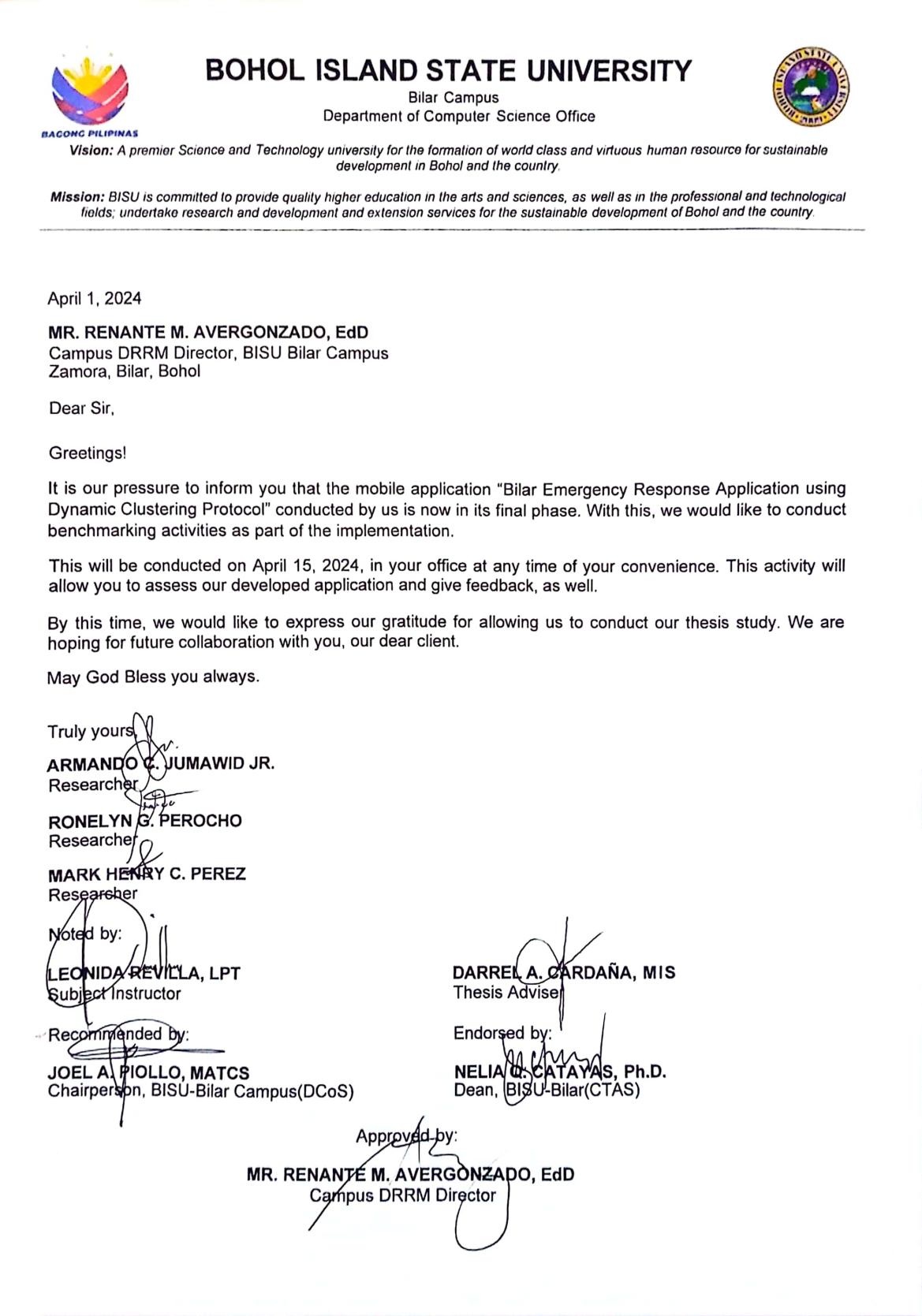
# APPENDIX A

## LETTER OF INTENT

****

# APPENDIX B

## LETTER OF IMPLEMENTATION

****

# APPENDIX C

## System Usability Questionnaire

**Instructions:**

* Please rate the usability of the system
* Try to respond to all the items
* For items that are not applicable, use N/A
* Make sure these fields are filled in

**Rating Scale:**

5- Strongly Agree

4- Agree

3- Neutral

2- Disagree

1. Strongly Disagree

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| **FUNCTIONALITY SUITABILITY** | **5** | | **4** | | | **3** | | | **2** | | **1** | |
| Functional completeness. The set of functions covers all the specified tasks and user objectives. |  | |  | | |  | | |  | |  | |
| Functional correctness. The project provides the correct results with the needed degree of precision. |  | |  | | |  | | |  | |  | |
| Functional appropriateness. The functions facilitate the accomplishment of specified tasks and objectives. |  | |  | | |  | | |  | |  | |
|  | | | | | | | | | | | | |
| **PERFORMANCE EFFICIENCY** | **5** | | **4** | | | **3** | | | **2** | | **1** | |
| Time behavior. The response and processing times and throughout rates of a product or system, when performing its functions, meet requirements. |  | |  | | |  | | |  | |  | |
| Resource utilization. The amounts and types of resources used by a product or system, when performing its functions, meet requirements. |  | |  | | |  | | |  | |  | |
|  | | | | | | | | | | | | |
| **COMPATIBILITY** | **5** | **4** | | | **3** | | | **2** | | | **1** | |
| Co-existence. A product can perform its required functions efficiently while sharing a common environment and resources with other products, without detrimental impact on any other product. |  |  | | |  | | |  | | |  | |
| Interoperability. Two or more systems, products or components can exchange information and use the information that has been exchanged. |  |  | | |  | | |  | | |  | |
|  | | | | | | | | | | | | |
| **USABILITY** | **5** | **4** | | | **3** | | | **2** | | | **1** | |
| Appropriateness Recognizability. Users can recognize whether a product or system is appropriate for their needs. |  |  | | |  | | |  | | |  | |
| Learnability. The project can be used by specified users to achieve specified goals of learning to use the product or system with effectiveness, efficiency, freedom from risk and satisfaction in a specified context of use. |  |  | | |  | | |  | | |  | |
| Operability. The project has attributes that make it easy to operate and control. |  |  | | |  | | |  | | |  | |
| User error protection. The project protects user against making errors. |  |  | | |  | | |  | | |  | |
| User interface aesthetics. User interface enables pleasing and satisfying interaction for the user. |  |  | | |  | | |  | | |  | |
| Accessibility. The project can be used by people with the widest range of characteristics and capabilities to achieve a specified goal in a specified context of use. |  |  | | |  | | |  | | |  | |
|  | | | | | | | | | | | | |
| **RELIABILITY** | **5** | **4** | | | **3** | | | **2** | | | **1** | |
| Maturity. The project or component meets needs for reliability under normal operation. |  |  | | |  | | |  | | |  | |
| Availability. The project or component is operational and accessible when required for use. |  |  | | |  | | |  | | |  | |
| Recoverability. In the event of an interruption or a failure, the project can recover the data directly affected and re-establish the desired state of the system. |  |  | | |  | | |  | | |  | |
|  | | | | | | | | | | | | |
| **SECURITY** | **5** | | | **4** | | | **3** | | | **2** | | **1** |
| Confidentiality. The project ensures that data are accessible only to those authorized to have access. |  | | |  | | |  | | |  | |  |
| Integrity. The project or component prevents unauthorized access to, or modification of, computer programs or data. |  | | |  | | |  | | |  | |  |
| Accountability. The actions of an entity can be traced uniquely to the entity. |  | | |  | | |  | | |  | |  |
| Authenticity. The identity of a subject or resource can be provided to be the one claimed. |  | | |  | | |  | | |  | |  |
| Non-repudiation. Actions or events can be proven to have taken place, so that the events or actions cannot be repudiated later. |  | | |  | | |  | | |  | |  |
|  | | | | | | | | | | | | |
| **MAINTAINABILITY** | **5** | | | **4** | | | **3** | | | **2** | | **1** |
| Modularity. The project is composed of discrete components such that a change to one component has minimal impact on other components. |  | | |  | | |  | | |  | |  |
| Reusability. An asset can be used in more than one system, or in building other assets. |  | | |  | | |  | | |  | |  |
| Analyzability. The effectiveness and efficiency with which it is possible to assess the impact on the project of an intended change to one or more of its parts, or to diagnose a project for deficiencies or causes of failures, or to identify parts to be modified. |  | | |  | | |  | | |  | |  |
| Modifiability. The project can be effectively and efficiently modified without introducing defects or degrading existing product quality. |  | | |  | | |  | | |  | |  |
| Testability. The effectiveness and efficiency with which test criteria can be established for the project and test can be performed to determine whether those criteria have been met. |  | | |  | | |  | | |  | |  |
|  |  | | |  | | |  | | |  | |  |
| **PORTABILITY** | **5** | | | **4** | | | **3** | | | **2** | | **1** |
| Adaptability. The degree to which a product or system can effectively and efficiently be adapted for different or evolving hardware, software, or other operational or usage environments. |  | | |  | | |  | | |  | |  |
| Installability. The effectiveness and efficiency with which a product or system can be successfully installed and/or uninstalled in a specified environment. |  | | |  | | |  | | |  | |  |
| Replaceability. A product can replace another specified software product for the same purpose in the same environment. diagnose a project for deficiencies or causes of failures, or to identify parts to be modified. |  | | |  | | |  | | |  | |  |

# APPENDIX D

# INTERVIEW GUIDE QUESTIONNAIRE

## BISARU HEAD

1. What are the current processes in responding an emergency in Bilar?

2. What are the challenges or issues encountered by Bilar Search and Rescue Unit (BISARU) in Bilar?

3. How does the BISARU address emergency response situations?

4. What security measures have been implemented to ensure the privacy and integrity of data in the application?

5. What are the limitations of this system in terms of knowledge and capabilities?

6. What are the services offered by BISARU to the people living in Bilar?

7. The proposed system will focus in terms of:

a. Emergency Response

b. Clustering Protocol

8. Emergency Responders:

* Who are the responders when an emergency happens? What means of transportation will they use in responding?
* How can the emergency responders know if an emergency situation happens?
* Do the emergency responders arrive on time? What is the next step after responding to an incident?

9. Dispatchers and Emergency Management Centers:

* What are the processes during emergency response?
* What are the precautionary measures will the management centers provide?

10. Affected Individuals and Communities:

* In emergency situations, what means of communication will the individuals use?
* Does the system provide the needed information of an incident?
* How can individuals use the application in areas with no network coverage? Does the application can be use offline?

11. Local Government and Authorities:

* Who are the authorized responders that the government approved?
* How will the local government unit address emergency related situations?

12. Procedure in responding emergency situations?

* If emergency happens, the base-station emergency response will send a signal together with the information of the incident and it's location, and it will directly be sent to the emergency responders at BISARU?
* If an emergency happened at a place where there is no network signal, how can they use the application? How will the responders know about their location?
* What are the steps to be follow in using the application?

13. Monitoring:

* Are there automated alerts or notifications integrated into the monitoring system to promptly notify relevant authorities or responders during critical events?
* Does the BERA application implement monitoring mechanisms to track and analyze emergency situations?
* How often do you evaluate emergency situations, is it daily, weekly, quarterly or yearly?
* Who is those people involved in doing the monitoring in emergency response situations?

15. Reports:

* Will the reports of emergencies can be generated by the system itself?
* Who are those people involved in generating reports?
* What are the processes being done in generating reports?

# APPENDICE E

## User’s Manual

1. **User Account Login**

**User Login**

Steps:

1. On the login page of the mobile app input your registered email address and password in the textbox.
2. Click the “Login” Button
3. System will automatically identify your role (Requestor/Witness and Responder) then load the necessary modules for the access role.

Expected Results:

1. Users can access the modules depending on the user roles of their account.

Clean Up:

1. Click the Profile Avatar above then click the “Logout” Button below.
2. **Account Registration**

**Responder Registration**

Steps:

1. On the login page click the “Create New Account” text below.
2. Input needed field data.
3. Click “Create” button.

Expected Results:

1. New Responder user was registered on the system.

Clean Up:

1. Click the home button to close the app.

**Requestor/Witness Registration**

Steps:

1. On the login page click the “Create New Account” text below.
2. Input needed field data.
3. Click “Create” button.

Expected Results:

1. New Requestor/Witness user was registered on the system.

Clean Up:

1. Click the home button to close the app.
2. **Update Profile**

**Update Name**

Steps:

1. On the main menu, click profile avatar in the top part or the system.
2. Then click the “Edit Icon” in the right side of the user name.
3. Input your new name.
4. Click “Save” button.

Expected Results:

1. The updated name is saved in the system.

Clean Up:

1. Click the exit button.

**Update Email**

Steps:

1. On the main menu, click profile avatar in the top part or the system.
2. Click the edit icon in the email box.
3. Input new email to be saved and the user password.
4. Click “Save” button.

Expected Results:

1. The new updated email is saved in the system.

Clean Up:

1. Click the exit button.

**Update Password**

Steps:

1. On the main menu, click profile avatar in the top part or the system.
2. Click “Change password” button.
3. Input the email, current password and the new password.
4. Click “Save” button.

Expected Results:

1. The new updated password is saved in the system.

Clean Up:

1. Click the exit button.

**Update Profile Picture**

Steps:

1. On the main menu, click profile avatar in the top part or the system.
2. Click the profile image.
3. Choose image either from camera or gallery.
4. Click “Save” button.

Expected Results:

1. The new updated profile image is saved in the system.

Clean Up:

1. Click the exit button.

**Update Phone Number**

Steps:

1. On the main menu, click profile avatar in the top part or the system.
2. Click the phone number.
3. Input your new phone number.
4. Click “Save” button.

Expected Results:

1. The new updated profile image is saved in the system.

Clean Up:

8. Click the exit button

1. **Requestor/Witness**

**Send Help**

Steps:

1. On the Home Page, Select a concern Police, Fire, Medical
2. A dialog box will show in the screen and describe the situation
3. Attach a photo of the situation
4. Click the submit button

Expected Results:

1. Send a notification to responder and admin

Clean Up:

9. Click the back button

**Show Nearby Responders**

Steps:

1. On the main menu, click the “Map” in the bottom navigation bar
2. Click the marked Responder to show the name

Expected Results:

1. Requestors/Witness will know the nearby responders.

Clean Up:

10.Click the back button

**Call Emergency Numbers When offline**

Steps:

1. On the main menu, click the “Contacts” in the bottom navigation bar
2. Select a number to call.
3. Click the call button.

Expected Results:

1. Successfully call the emergency numbers.

Clean Up:

11.Click the back button

1. **Responder**

**Receiving Notification and Navigating**

Steps:

1. On the home page wait for someone who send notification.
2. A dialog box will show with the requestor’s details.
3. Click the “Call” button to call the requestor/witness.
4. Click “Navigate” button to navigate.
5. It automatically opens the google maps with the requestor’s coordinates.
6. Go back to the responder application and Click the “Arrived” button.

Expected Results:

1. Successfully call and navigate the requestors/witness.

Clean Up:

12.Click the back button

1. **Admin**

**Show History**

Steps:

1. On the main menu, click the “Home” button.
2. Admin will see the history

Expected Results:

1. The Admin will monitor the recent notifications.

Clean Up:

1. Click the back button.

**Admin Map**

Steps:

1. On the main menu, click the “Map” button in the bottom navigation bar.
2. Click the mark to show the registered users.

Expected Results:

1. The Admin will monitor the users.

Clean Up:

1. Click the back button.

**Manage Accounts**

Steps:

1. On the main menu, click the “Manage Users” button in the bottom navigation bar.
2. Click the “Manage Requestors” tab to delete account.
3. Click the “Manage Responders” tab to edit and delete account.

Expected Results:

1. The Admin will successfully edit and delete accounts

Clean Up:

1. Click the exit button.

## APPENDICE F

### Source Code

**authentication\_service.dart**

import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:dartz/dartz.dart';

import 'package:firebase\_auth/firebase\_auth.dart' hide User;

import 'package:responder/app/app.locator.dart';

import 'package:responder/model/admin.dart';

import 'package:responder/services/internet\_service.dart';

import 'package:responder/services/shared\_pref\_service.dart';

import 'package:responder/ui/common/app\_exception\_constants.dart';

import 'package:responder/ui/common/firebase\_constants.dart';

import '../exception/app\_exception.dart';

import '../model/user.dart';

class AuthenticationService {

final auth = FirebaseAuth.instance;

final db = FirebaseFirestore.instance;

final \_sharedPref = locator<SharedPreferenceService>();

final \_internetService = locator<InternetService>();

bool get isLoggedIn => auth.currentUser != null;

Future<Either<AppException, None>> logout() async {

try {

await auth.signOut();

await \_sharedPref.deleteCurrentUser();

return const Right(None());

} catch (e) {

return Left(AppException(e.toString()));

}

}

Future<Either<AppException, None>> forgotPassword(

{required String email}) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

try {

await auth.sendPasswordResetEmail(email: email);

return const Right(None());

} on FirebaseAuthException catch (e) {

return Left(AppException(e.message.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

Future<Either<AppException, None>> logoutAdmin() async {

try {

await auth.signOut();

await \_sharedPref.deleteCurrentAdmin();

return const Right(None());

} catch (e) {

return Left(AppException(e.toString()));

}

}

Future<Either<AppException, User>> login(

{required String email, required String password}) async {

try {

final credential = await auth.signInWithEmailAndPassword(

email: email, password: password);

if (credential.user == null) {

return Left(AppException("User not found"));

}

final userId = credential.user!.uid;

final snap = await db.collection("responder").doc(userId).get();

final user = User.fromJson(snap.data()!);

return Right(user);

} on FirebaseAuthException catch (e) {

String errorMessage = e.message.toString();

if (errorMessage == "wrong-password") {

errorMessage = "The password you entered is wrong!";

} else if (errorMessage == "user-not-found") {

errorMessage = "No user found!";

}

return Left(AppException(errorMessage));

} catch (e) {

return Left(AppException(e.toString()));

}

}

Future<Either<AppException, Admin>> adminlogin(

{required String email, required String password}) async {

try {

final credential = await auth.signInWithEmailAndPassword(

email: email, password: password);

if (credential.user == null) {

return Left(AppException("User not found"));

}

final userId = credential.user!.uid;

final snap = await db.collection("admin").doc(userId).get();

final admin = Admin.fromJson(snap.data()!);

return Right(admin);

} on FirebaseAuthException catch (e) {

String errorMessage = e.message.toString();

if (errorMessage == "wrong-password") {

errorMessage = "The password you entered is wrong!";

} else if (errorMessage == "user-not-found") {

errorMessage = "No user found!";

}

return Left(AppException(errorMessage));

} catch (e) {

return Left(AppException(e.toString()));

}

}

Future<Either<AppException, None>> updateEmail(

{required String currentEmail,

required String newEmail,

required String password}) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

try {

var response = await login(email: currentEmail, password: password);

return response.fold((l) => Left(AppException(l.message)), (r) async {

try {

await auth.currentUser!.updateEmail(newEmail);

await db.collection("responder").doc(r.uid).update(

{"email": newEmail},

);

await getCurrentUser();

return const Right(None());

} catch (e) {

print(e.toString());

return Left(AppException(e.toString()));

}

});

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

Future<Either<AppException, None>> updatePassword(

{required String currentPassword, required String newPassword}) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

try {

if (auth.currentUser == null) {

return Left((AppException("No User Found")));

}

var response = await login(

email: auth.currentUser!.email!, password: currentPassword);

return response.fold((l) => Left(AppException(l.message)), (r) async {

try {

await auth.currentUser!.updatePassword(newPassword);

return const Right(None());

} catch (e) {

return Left(AppException(e.toString()));

}

});

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

Future<Either<AppException, User>> getCurrentUser() async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

try {

final currentUser = await \_sharedPref.getCurrentUser();

if (currentUser == null) {

return Left(AppException("No Current User"));

} else {

final updateUserDoc =

await db.collection('responder').doc(currentUser.uid).get();

User user = User.fromJson(updateUserDoc.data()!);

await \_sharedPref.saveUser(user);

return Right(User.fromJson(updateUserDoc.data()!));

}

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

Future<Either<AppException, None>> signup(

String name, String email, String password, String phoneNum) async {

try {

UserCredential credential = await auth.createUserWithEmailAndPassword(

email: email,

password: password,

);

if (credential.user == null) {

return left(AppException(AppExceptionConstants.userNotFound));

}

User responder = User(

name: name,

email: email,

phonenumber: phoneNum,

uid: credential.user!.uid,

);

await db

.collection(FirebaseConstants.userCollection)

.doc(credential.user?.uid)

.set(responder.toJson());

return right(const None());

} on FirebaseAuthException catch (e) {

String errorMessage = e.message.toString();

print(errorMessage);

if (e.code == 'weak-password') {

errorMessage = AppExceptionConstants.passwordIsWeak;

} else if (e.code == 'email-already-in-use') {

errorMessage = AppExceptionConstants.accountAlreadyExists;

}

return left(AppException(errorMessage));

} catch (error) {

return left(AppException(error.toString()));

}

}

Future<Either<AppException, None>> adminSignup(

String name, String email, String password, String phoneNum) async {

try {

UserCredential credential = await auth.createUserWithEmailAndPassword(

email: email,

password: password,

);

if (credential.user == null) {

return left(AppException(AppExceptionConstants.userNotFound));

}

Admin admin = Admin(

name: name,

email: email,

phonenumber: phoneNum,

uid: credential.user!.uid,

);

await db

.collection(FirebaseConstants.adminCollection)

.doc(credential.user?.uid)

.set(admin.toJson());

return right(const None());

} on FirebaseAuthException catch (e) {

String errorMessage = e.message.toString();

print(errorMessage);

if (e.code == 'weak-password') {

errorMessage = AppExceptionConstants.passwordIsWeak;

} else if (e.code == 'email-already-in-use') {

errorMessage = AppExceptionConstants.accountAlreadyExists;

}

return left(AppException(errorMessage));

} catch (error) {

return left(AppException(error.toString()));

}

}

}

**image service.dart**

import 'dart:io';

import 'package:dartz/dartz.dart';

import 'package:firebase\_storage/firebase\_storage.dart';

import 'package:responder/app/app.locator.dart';

import 'package:responder/exception/app\_exception.dart';

import 'package:responder/services/internet\_service.dart';

class ImageService {

final \_storageRef = FirebaseStorage.instance.ref();

final \_internetService = locator<InternetService>();

Future<Either<AppException, String>> uploadImage(

File image, String path) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

try {

final imageProfile = await \_storageRef.child(path).putFile(image);

final imageUrl = await imageProfile.ref.getDownloadURL();

return Right(imageUrl);

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

}

**internet\_service.dart**

import 'package:connectivity\_plus/connectivity\_plus.dart';

class InternetService {

Future<bool> hasInternetConnection() async {

final ConnectivityResult connectivityResult =

await Connectivity().checkConnectivity();

if (connectivityResult == ConnectivityResult.mobile ||

connectivityResult == ConnectivityResult.wifi ||

connectivityResult == ConnectivityResult.ethernet ||

connectivityResult == ConnectivityResult.vpn) {

return true;

} else {

return false;

}

}

}

**location\_service.dart**

import 'dart:async';

import 'package:geolocator/geolocator.dart';

class LocationService {

Future<Position> getCurrentLocation() async {

return await Geolocator.getCurrentPosition(

desiredAccuracy: LocationAccuracy.high);

}

}

**notificatiom\_service.dart**

import 'package:flutter/material.dart';

import 'package:firebase\_messaging/firebase\_messaging.dart';

import 'package:flutter\_local\_notifications/flutter\_local\_notifications.dart';

class NotificationService {

final FirebaseMessaging messaging = FirebaseMessaging.instance;

final FlutterLocalNotificationsPlugin flutterLocalNotificationsPlugin =

FlutterLocalNotificationsPlugin();

Future<void> requestNotificationPermission() async {

NotificationSettings settings = await messaging.requestPermission(

alert: true,

announcement: true,

badge: true,

carPlay: true,

criticalAlert: true,

provisional: true,

sound: true,

);

if (settings.authorizationStatus == AuthorizationStatus.authorized) {

print('User Granted Permission');

} else if (settings.authorizationStatus ==

AuthorizationStatus.provisional) {

print('User Granted Provisional Permission');

} else {

print('User Denied Permission');

}

}

Future<void> showFlutterNotification(

BuildContext context, RemoteMessage message) async {

showDialog(

context: context,

builder: (BuildContext dialogContext) {

return AlertDialog(

title: Text(message.notification?.title ?? ''),

content: Text(message.notification?.body ?? ''),

actions: [

TextButton(

onPressed: () {

Navigator.of(dialogContext).pop();

},

child: Text('Close'),

),

],

);

},

);

}

}

NotificationAppLaunchDetails? selectNotificationSubject;

BuildContext? navigationContext;

void initState(BuildContext context) {

navigationContext = context;

NotificationService().requestNotificationPermission();

}

**shared\_pref.dart**

import 'dart:async';

import 'dart:convert';

import 'package:responder/model/admin.dart';

import 'package:shared\_preferences/shared\_preferences.dart';

import '../model/user.dart';

class SharedPreferenceService {

late StreamController<User?> \_userStreamController;

late StreamController<Admin?> \_adminStreamController;

SharedPreferenceService() {

\_userStreamController = StreamController<User?>.broadcast();

\_adminStreamController = StreamController<Admin?>.broadcast();

}

Stream<User?> get userStream => \_userStreamController.stream;

Stream<Admin?> get adminStream => \_adminStreamController.stream;

Future<void> deleteCurrentUser() async {

final sharedPref = await SharedPreferences.getInstance();

sharedPref.remove("USER\_KEY");

} Future<void> deleteCurrentAdmin() async {

final sharedPref = await SharedPreferences.getInstance();

sharedPref.remove("ADMIN\_KEY");

}

Future<String?> getUserId() async {

final user = await getCurrentUser();

if (user != null) {

return user.uid;

} else {

return null;

}

}

Future<String?> getAdminId() async {

final admin = await getCurrentAdmin();

if (admin != null) {

return admin.uid;

} else {

return null;

}

}

Future<User?> getCurrentUser() async {

final sharedPref = await SharedPreferences.getInstance();

final user = sharedPref.getString("USER\_KEY");

if (user == null) return null;

return User.fromJson(json.decode(user));

}

Future<Admin?> getCurrentAdmin() async {

final sharedPref = await SharedPreferences.getInstance();

final admin = sharedPref.getString("ADMIN\_KEY");

if (admin == null) return null;

return Admin.fromJson(json.decode(admin));

}

Future<void> saveUser(User user) async {

if (\_userStreamController.isClosed) {

\_userStreamController = StreamController<User?>.broadcast();

}

\_userStreamController.add(user);

final sharedPref = await SharedPreferences.getInstance();

await sharedPref.setString("USER\_KEY", jsonEncode(user.toJson()));

}

Future<void> saveAdmin(Admin admin) async {

if (\_adminStreamController.isClosed) {

\_adminStreamController = StreamController<Admin?>.broadcast();

}

\_adminStreamController.add(admin);

final sharedPref = await SharedPreferences.getInstance();

await sharedPref.setString("ADMIN\_KEY", jsonEncode(admin.toJson()));

}

void dispose() {

\_userStreamController.close();

\_adminStreamController.close();

}

}

**user\_service.dart**

import 'dart:io';

import 'package:cloud\_firestore/cloud\_firestore.dart';

import 'package:dartz/dartz.dart';

import 'package:responder/app/app.locator.dart';

import 'package:responder/exception/app\_exception.dart';

import 'package:responder/model/user.dart';

import 'package:responder/services/authentication\_service.dart';

import 'package:responder/services/image\_service.dart';

import 'package:responder/services/internet\_service.dart';

import 'package:responder/services/shared\_pref\_service.dart';

class UserService {

final \_internetService = locator<InternetService>();

final \_db = FirebaseFirestore.instance;

final \_sharedPref = locator<SharedPreferenceService>();

final \_authServ = locator<AuthenticationService>();

final \_imageService = locator<ImageService>();

Future<Either<AppException, None>> updateName(String name) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

User? user = await \_sharedPref.getCurrentUser();

try {

await \_db

.collection("responder")

.doc(user!.uid)

.set({"name": name}, SetOptions(merge: true));

\_authServ.getCurrentUser();

return const Right(None());

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

Future<Either<AppException, None>> uploadProfilePicture(

File imageFile) async {

final bool hasInternet = await \_internetService.hasInternetConnection();

if (hasInternet) {

User? user = await \_sharedPref.getCurrentUser();

String path = "Images/${user!.uid}.png";

try {

final response = await \_imageService.uploadImage(imageFile, path);

return response.fold(

(l) => Left(AppException(l.message)),

(imageUrl) async {

try {

await \_db

.collection("responder")

.doc(user.uid)

.set({"image": imageUrl}, SetOptions(merge: true));

await \_authServ.getCurrentUser();

return const Right(None());

} catch (e) {

return Left(AppException(e.toString()));

}

},

);

} catch (e) {

return Left(AppException(e.toString()));

}

} else {

return Left(AppException("Please check your internet connection!"));

}

}

## APPENDICE G

### RESEARCHER’S BIODATA

**Name** : Armando Caja Jumawid Jr.

**Place of Birth** : Poblacion Sur, Carmen, Bohol

**Birthdate** : May 11, 2002

**Age** : 21

**Home Address** : Katipunan, Carmen, Bohol

**Email Address** : armando.jumawid@gmail.com

**Religion** : Roman Catholic

**Citizenship** : Filipino

**Father’s Name** : Armando Jumawid Sr.

**Mother’s Name** : Luzviminda Jumawid

**EDUCATIONAL BACKGROUND**

**Elementary** : Katipunan, Elementary, School

Katipunan, Carmen, Bohol

2013-2014

**Secondary**

**Junior High School :** Saint Anthony Academy

Poblacion, Carmen, Bohol

2017-2018

**Senior High School :** Katipunan National High School

Katipunan, Carmen, Bohol

2019-2020

**Tertiary** : Bohol Island State University

Zamora, Bilar, Bohol

2023-2024

**Work Experience** : On-the-Job Training

BLEND IT

Tagbilaran City, Bohol

**RESEARCHER’S BIODATA**

**Name** : Ronelyn Garcia Perocho

**Place of Birth** : San Vicente, Pilar, Bohol

**Birthdate** : September 1, 2001

**Age** : 22

**Home Address** : San Vicente, Pilar, Bohol

**Email Address** : lyngarciaperocho@gmail.com

**Religion** : Iglesia Filipina Independiente (IFI)

**Citizenship** : Filipino

**Father’s Name** : Roman Galido Perocho

**Mother’s Name** : Liza Garcia Perocho

**EDUCATIONAL BACKGROUND**

**Elementary** : San Vicente, Elementary School

San Vicente, Pilar, Bohol

2013-2014

**Secondary**

**Junior High School :** Pilar Technical Vocational High School

Poblacion, Pilar, Bohol

2017-2018

**Senior High School :** Pilar Technical Vocational High School

Poblacion, Pilar, Bohol

2019-2020

**Tertiary** : Bohol Island State University

Zamora, Bilar, Bohol

2023-2024

**Work Experience** : On-the-Job Training

Alturas Group of Companies

Poblacion, Talibon, Bohol

**RESEARCHER’S BIODATA**

**Name** : Mark Henry Cesar Perez

**Place of Birth** : Tagbilaran City

**Birthdate** : October 15, 1999

**Age** : 24

**Home Address** : Calvario, Loay, Bohol

**Email Address** : markhenryperez123@gmail.com

**Religion** : Roman Catholic

**Citizenship** : Filipino

**Father’s Name** : Edwin Perez

**Mother’s Name** : Camila Jane Perez

**EDUCATIONAL BACKGROUND**

**Elementary** : Senate President Protempore Jose Clarin Memorial School

Poblacion Ibabao, Loay, Bohol

2011-2012

**Secondary**

**Junior High School :** Camayaan, Loboc, National High School

Camayaan, Loboc, Bohol

2015-2016

**Senior High School :** Bohol Island State University

Zamora, Bilar, Bohol

2017-2018

**Tertiary** : Bohol Island State University

Zamora, Bilar, Bohol

2023-2024

**Work Experience** : On-the-Job Training

Municipality of Batuan

Poblacion, Batuan, Bohol