Mafqodaty

Submitted as Partial Fulfillment of the Requirements for the Bachelor's Degree in Information Technology

By

Iamail Ahmad Abdelhadi (20191521)

Software Engineering

Supervised by

Dr. Ahmad Al-Naboot

Jan, 2024

COMMITTEE REPORT

We certify that we have read this graduation project report as examining committee, examined the student in its content and that it is adequate as a project document for B.Sc. in Computer Science.

Chairman: Supervisor:

Name: …………………   Name: …………………

Signature: ……………….   Signature: …………….

Date: /   /   Date: /   /

Examiner:

Name: …………………

Signature: …………….

Date: /   /

TABLE OF CONTENT

TABLE OF FIGURES

TABLE OF TABLES

ABSTRACT

"Mafqodaty" is an innovative lost-found Android application designed to address the challenges users face in reporting lost items and finding possessions in Jordan. The application offers a user-friendly platform where individuals can seamlessly report details about their lost belongings and discover found items within their community. By incorporating advanced search algorithms, secure communication features, and robust matching mechanisms, "Mafqodaty" streamlines the lost-found process, fostering a collaborative environment for users. With a focus on data security, privacy, and user customization, the application aims to redefine the lost-found experience, providing a reliable and efficient solution tailored to the unique needs of users in Jordan. "Mafqodaty" envisions becoming a central hub for reuniting individuals with their lost possessions, enhancing community engagement, and contributing to a more connected and secure society.

DEDICATION

This work is dedicated to ourselves and our families, who have never failed to provide us with financial and emotional support, for meeting all our requirements while we were developing our system, and for showing us that even the most difficult endeavor can be completed if done one step at a time. We dedicate this project to all the people who have worked hard to assist us in completing it.

ACKNOWLEDGMENT

First, we would like to express our very great appreciation to Dr. Ahmad Al-Nabot for his valuable and constructive suggestions during the planning and development of this project, His willingness to give his time so generously has been very much appreciated.

We would also like to express our gratitude and great appreciation to the virtuous discussion committee. We also thank the Faculty of Information Technology, represented by all faculty members without exception, for the academic and moral support and guidance they provided us, which had a prominent impact on this effort. Likewise, do not forget about our dear university, which we cherish our belongings to during this decisive and important period of our lives, which we spent in its entirety.

Finally, our sincere thanks to our people, especially the source of tenderness, care, and guidance, the mother and our guide in our lives, the virtuous father, for their encouragement, patience, and help over the years, we owe them forever, and I ask God Almighty to extend our life so that we can fulfill them.

CHAPTER ONE: INTRODUCTION

1.1 Overview

Welcome to "Mafqodaty," your go-to Android application for seamlessly reporting lost items and discovering found belongings. Mafqodaty streamlines the process, allowing users to submit details effortlessly and search for matches within the community. With secure authentication, privacy-centric policies, and real-time notifications, Mafqodaty ensures a reliable and efficient experience. Join our collaborative network and download Mafqodaty today to be a part of a community dedicated to reuniting people with their lost possessions.

1.2 Problem Statement

In today's fast-paced world, the unfortunate scenario of losing personal belongings is not uncommon. The lack of a streamlined and efficient system for reporting lost items and discovering found possessions poses a significant challenge. Existing solutions often suffer from fragmented processes, security concerns, and limited community engagement. Recognizing this gap, "Mafqodaty" aims to address these issues by providing a user-friendly Android application that simplifies the reporting and retrieval of lost items, fostering a secure and collaborative environment for users to reconnect with their belongings.

1.3 Research Objectives

Search objectives are key aspects of our research strategy for "Mafqodaty." They guide our exploration into user experiences and existing platforms, aiming to enhance communication, data accuracy, security, and feature integration. These objectives ensure a focused approach to optimize the lost-found experience, providing a seamless and valuable service for users in Jordan.

1.4 Scope of the study

This study encompasses a comprehensive exploration of the "Mafqodaty" lost-found application within the context of users in Jordan. The research will delve into user experiences, focusing on reporting lost items and searching for found possessions. The scope extends to a comparative analysis of existing lost-found platforms, identifying features that could elevate "Mafqodaty's" functionality. Additionally, the study will address communication efficiency, data accuracy, security measures, and potential feature integrations. The aim is to provide valuable insights and recommendations for optimizing the application's performance and user satisfaction in the unique landscape of lost-found activities in Jordan.

1.5 Used Technologies and Tools

1.5.1 Software

• Firebase

Firebase is a Backend-as-a-Service (Baas). It provides developers with a variety of tools and services to help them develop quality apps, grow their user base, and earn profit. It is built on Google's infrastructure. Firebase is categorized as a NoSQL database program, which stores data in JSON-like documents. [1]

• Android SDK

The Android SDK provides you the API libraries and developer tools necessary to build, test, and debug apps for Android.

If you're a new Android developer, we recommend you download the ADT Bundle to quickly start developing apps. It includes the essential Android SDK components and a version of the Eclipse IDE with built-in ADT (Android Developer Tools) to streamline your Android app development.

1.6.2 Languages

• Java

Java is a programming language and computing platform first released by Sun Microsystems in 1995. It has evolved from humble beginnings to power a large share of today's digital world, by providing the reliable platform upon which many services and applications are built. New, innovative products and digital services designed for the future continue to rely on Java, as well.[2]

1.6.3 Editors and Other Tools

• Android Studio

Android Studio is the official Integrated Development Environment (IDE) for Android app development, based on IntelliJ IDEA. On top of IntelliJ's powerful code editor and developer tools, Android Studio offers even more features that enhance your productivity when building Android apps, such as:[3]

- A flexible Gradle-based build system
- A fast and feature-rich emulator
- A unified environment where you can develop for all Android devices
- Apply Changes to push code and resource changes to your running app without restarting your app
- Code templates and GitHub integration to help you build common app features and import sample code

2.7 Methodology

This application was developed using a form of Agile methodology, because it was the most suitable method that could match our project's nature that needs users while developing the software to get feedbacks with fast responding and related time constraints. the application development needed to work through iterations with specific amount of time.

The Agile is an iterative approach to project management and software development that helps teams deliver value to their customers faster and with fewer troubles. Instead of betting everything on a "big bang" launch, an agile team delivers work in small, but consumable, increments. Requirements, plans, and results are evaluated continuously so teams have a natural mechanism for responding to change quickly [5].
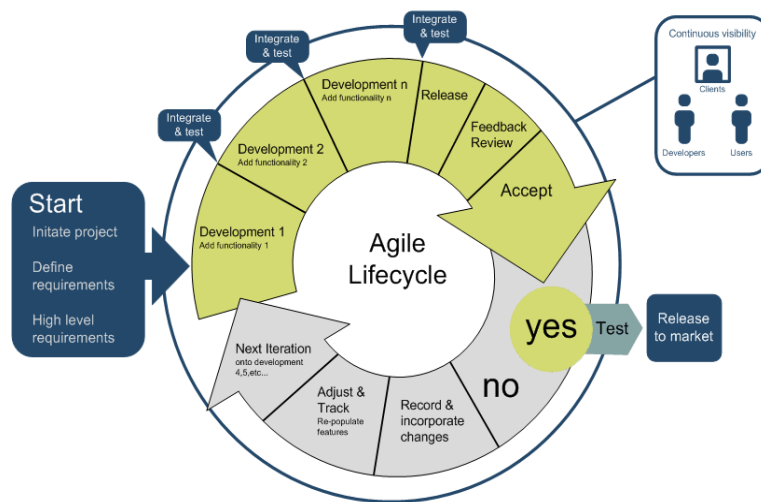

Figure 1- Agile methodology

2.8 Sprint in Gannt Chart

➢ Sprint 1: Planning

It was at this stage that I began planning every project and writing down all the steps I must take to reach the goal I wanted.

Estimated time: 3 days.

➢ Sprint 2: Learn the basics of the android development

3

At this point, we began studying android development.

Estimated time: two weeks

- ➤ Sprint 3: Test what I learned

  At this point, I did some Task to make sure I understood what I learned well.

  Estimated time: 1 week.

- ➤ Sprint 4: Design and Create Database

  In this Stage I draw the ER diagram and the relation between tables then create Database and tables.

  Estimated time: 4 days.

- ➤ Sprint 5: Store data in Database

  After I create Database and table connect Database with scraping code and start store fetched data in Database.

  Estimated time: 3 days.

- ➤ Sprint 6: UI Design process

  In this Stage I start the design process to all screens in project.

  Estimated time: 4days.

- ➤ Sprint 7: Planning the code

  In this stage we am planning all xml code in android studio to see the sequence of the application and how the data will display to users.

  Estimated time: 8 days.

- ➤ Sprint 8: Connect firebase with the application

  In this Stage we write java code to fetch all data from firebase to the website and take all design attribute.

  Estimated time: 15 days.

4

> ➢ Sprint 10: Acceptance testing

 After we finish all requirement of project, we do acceptance testing to every
 code and screen work as it should.

 Estimated time: 10 days

2.9 Gantt Chart

A Gantt chart is a project management tool that illustrates work completed over a period of time in relation to the time planned for the work. A Gantt chart can include the start and end dates of tasks, milestones, dependencies between tasks, assignees, and more.[4]

Days / Tasks	3days	14 days	7 days	8days	4 days	38days	5days
Planning	■						
Learn java languages		■					
Test what we learn			■				
UI design process				■			
Planning					■		
Implementation						■	
Testing							■

Table 1 Gantt Chart

CHAPTER TWO: REQUIREMENTS AND PLANNING

2.1 Introduction

Planning is one of the most important project management and time management techniques. Planning is preparing a sequence of action steps to achieve some specific goal. If you do it effectively, you can reduce time and effort to achieve the goal.

A plan is like a map. When following a plan, you can always see how much you have progressed towards your project goal and how far you are from your destination. Knowing where you are is essential for making good decisions on where to go or what to do next.

One more reason why you need planning is again the 80/20 Rule. It is well established that for unstructured activities 80 percent of the effort give less than 20 percent of the valuable outcome. You either spend much time on deciding what to do next, or you are taking many unnecessary, unfocused, and inefficient steps.

2.2 Requirements

After gathering the requirements that we publish for a group of collages and analysis, we have the bellow requirements for our project

2.2.1 Functional Requirements

- As a new user, I want to create an account using my email or mobile number so that I can start using the "Mafqodaty" application.
- As a user, I want to securely log in with my credentials and have the option to recover my account if needed.
- As a user who lost an item, I want to easily report the details of my lost possession, including a description, location, and date.
- As a user, I want to upload images or documents related to my lost item to provide more information.
- As a user who found an item, I want to submit details about the found possession, including a description, location, and date.
- As a user, I want to upload images or documents to support my submission of a found item.
- As a user who reported a lost item, I want to be notified when there is a potential match for my lost possession.
- As a user involved in a potential match, I want to communicate securely with the other user to coordinate the retrieval of the lost item.
- As a user, I want to manage my profile information, including personal details and notification preferences.
- As a user, I want the application to support multiple languages so that I can use it comfortably in my preferred language.

7

2.2.2 Non-Functional Requirements
- Performance
  - The application has to load and respond within three seconds for all key functionalities.
  - It needs to handle a concurrent consumer load of as a minimum a thousand users without performance degradation.
- Reliability
  - The system should have an uptime of 99.9% to ensure users can access it reliably.
  - Regular data backups should be performed to prevent data loss in case of unforeseen events.
- Security
  - All user information must be encrypted and can only be accessed by the user himself.
- Compatibility
  - The system must have as few errors as possible, and the errors that may occur do not affect the function of the system
- Usability
  - The utility have to adhere to accessibility standards, ensuring it is usable by means of people with disabilities.
  - User interfaces and interactions must observe fine practices for a effective consumer experience.
- Maintainability
  - The codebase must be nicely-documented to facilitate ease of protection and destiny improvements.
  - Regular updates and bug fixes must be deployed seamlessly without disrupting consumer activities.

2.3 Communication Plan

A communication plan documents the proposed and agreed upon paths and mechanisms of communication across a project. It does involve:

1) Determining the objectives

2) Choosing the audience

3) Selecting appropriate channel to reach them.

In communication plan we have cover everything that result from this project Including setting object and choosing people and select appropriate channel.

2.4 Risk Management

A risk management plan identifies all possible problems which could lead to delays and/or failures of the project plan. These must be identified very early in the project so that they can be reacted to quickly and will hopefully not hinder the project in any way.

2.4.1 Project Risk

➢ Risk: Tasks not being completed on time/ Failure to meet a deadline.

Probability: High

Estimated Impact: High

Response: Re-schedule and re-prioritize. Understand why there a delay has been and ensure it does not happen again.

➢ Risk: Difficulty communicating with the supervisor due to online learning.

Probability: Medium

Estimated effect: Medium

Response: The supervisor makes a classroom on Microsoft Teams.

➢ Risks: Misunderstanding with the client

Probability: low

Estimated impact: High

Response: Listening to the customer's point of view and taking his opinion regularly on what is being done

2.4.2 Technical Risk

➢ Risk: failure in the code

Probability: high

Estimated Impact: high

Response: rewrite the code and make sure to write comments in the code, it's easy to edit on the code

➢ Risk: Files lost or deleted.

Probability: Moderate

Estimated Impact: high

Response: The advantage with using Dropbox is that files can be retrieved easily, however if a file is lost from somebodies' computer or a hard copy of a file is lost the only way to get it back is to re-create it.

2.5 Testing Plan

Test plans are a way to ensure that all aspects of a product are tested during each sprint of the process, and that nothing gets left out as the project progresses. As the Scrum methodology we follow we had to test every sprint separately to check verification and validation. Then we had to do integration test for the sprints which done. After that we tested the whole project, we developed by acceptance testing. This ensures that all aspects are tested every time.

2.6 Issue Log Problem

Surely any project management face a problem. We face that:

1) Internet connection disconnection process.

3) The project working time is less due to the other lectures.

4) Difficulty learning new languages from scratch.

5) The short period of time for the end of the project and the stress of the semester with exams.

7) Working alone was a great challenge and I tried to work with minimal losses

8) Science and the web world is developing tremendously and rapidly.

CHAPTER THREE: SYSTEM DESIGN AND ARCHITECTURE

3.1 Overview

The System Design and Architecture chapter is a critical component in the development of any software project. It serves as the blueprint for constructing a robust and scalable system that meets the functional and non-functional requirements outlined in the project specifications. This chapter delves into the high-level structure, components, and interactions that form the foundation of the entire software system.

3.2 Entity Relationship Diagram

An Entity Relationship Diagram (ERD) is a type of diagram that lets you see how different entities (e.g. people, customers, or other objects) relate to each other in an application or a database.

They are created when a new system is being designed so that the development team can understand how to structure the database. They can also be created on an existing system to help the team understand how the system works and to find and resolve any issues.

Entity Relationship Diagrams use a specific set of symbols, such as shapes and arrows, to depict the system and database.*[6]*
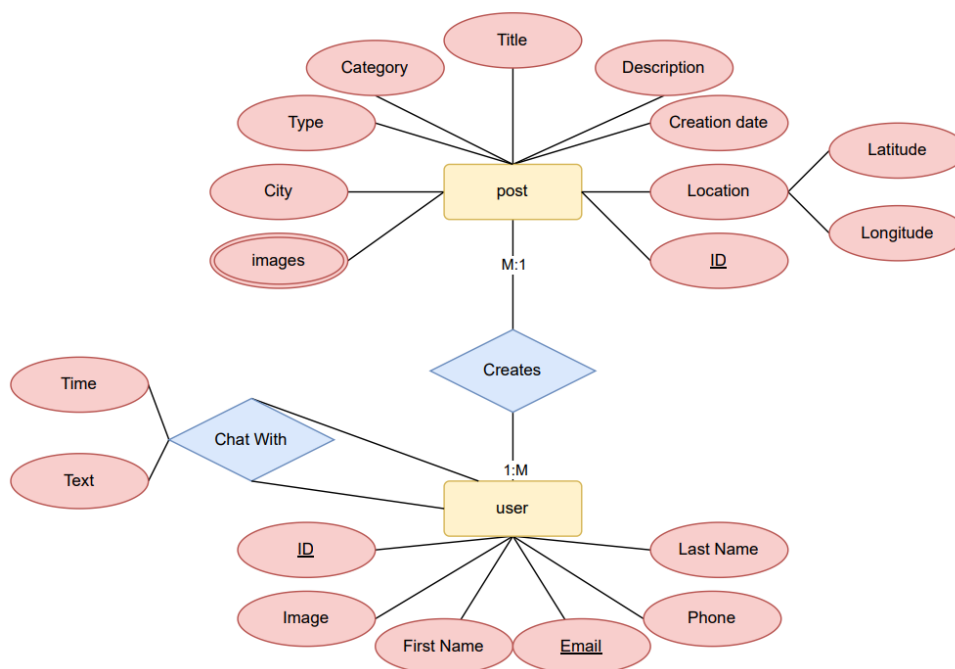
Figure 2 Entity Relationship Diagram

3.3 Flow chart

Flowcharts are nothing but the graphical representation of the data or the algorithm for a better understanding of the code visually. It displays step-by-step solutions to a problem, algorithm, or process. It is a pictorial way of representing steps that are preferred by most beginner-level programmers to understand algorithms of computer science, thus it contributes to troubleshooting the issues in the algorithm. A flowchart is a picture of boxes that indicates the process flow sequentially. Since a flowchart is a pictorial representation of a process or algorithm, it's easy to interpret and understand the process. To draw a flowchart, certain rules need to be followed which are followed by all professionals to draw a flowchart and are widely accepted all over the countries.*[7]*

3.3.1 Login Flow Chart
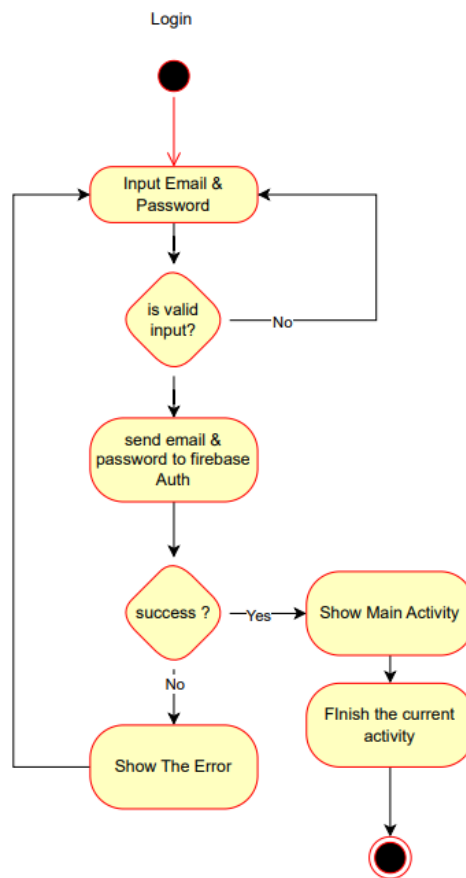
Figure 3-Login Flowchart

3.3.2 Register Flow Chart
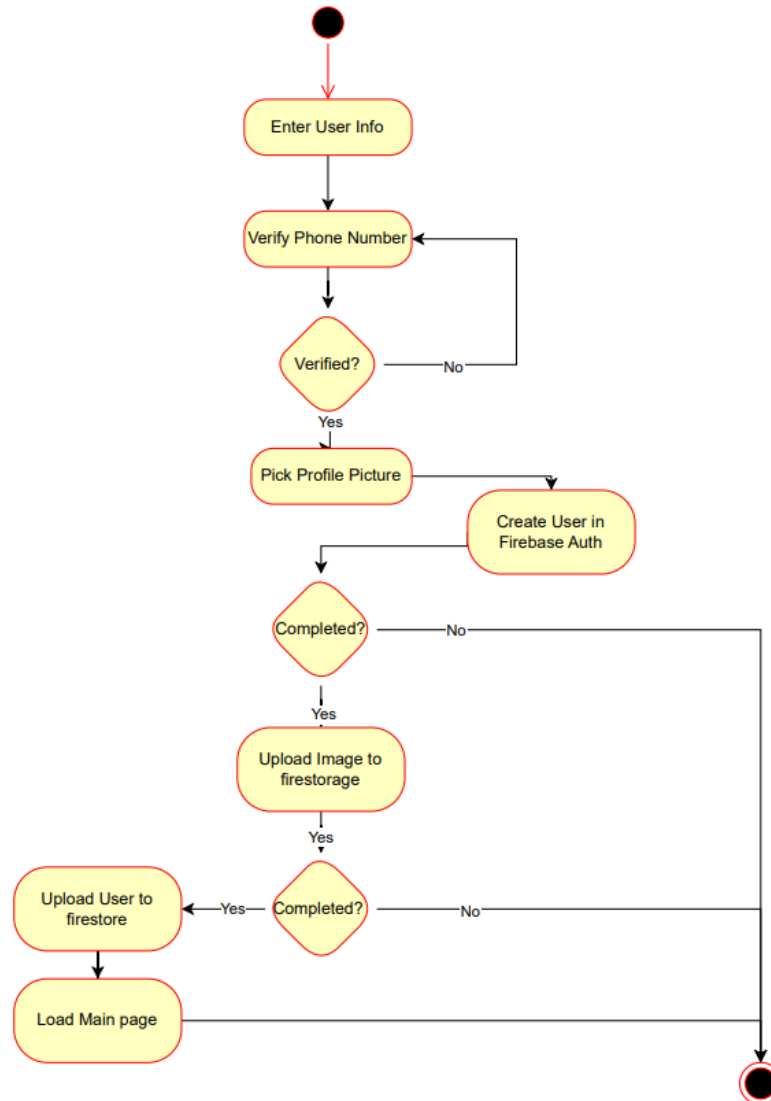
Figure 4-Register flow chart

3.3.4 Create Post Flow Chart
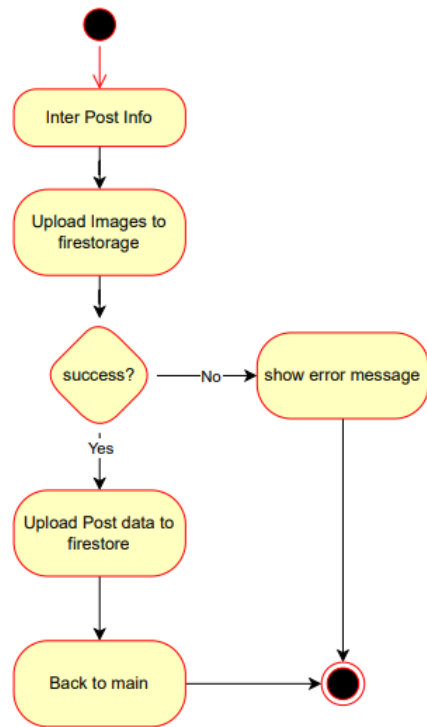
Figure 5-Create Post Flow Chart

3.4 Usecase diagram

A Use Case Diagram is a vital tool in system design, it provides a visual representation of how users interact with a system. It serves as a blueprint for understanding the functional requirements of a system from a user's perspective, aiding in the communication between stakeholders and guiding the development process.[8]
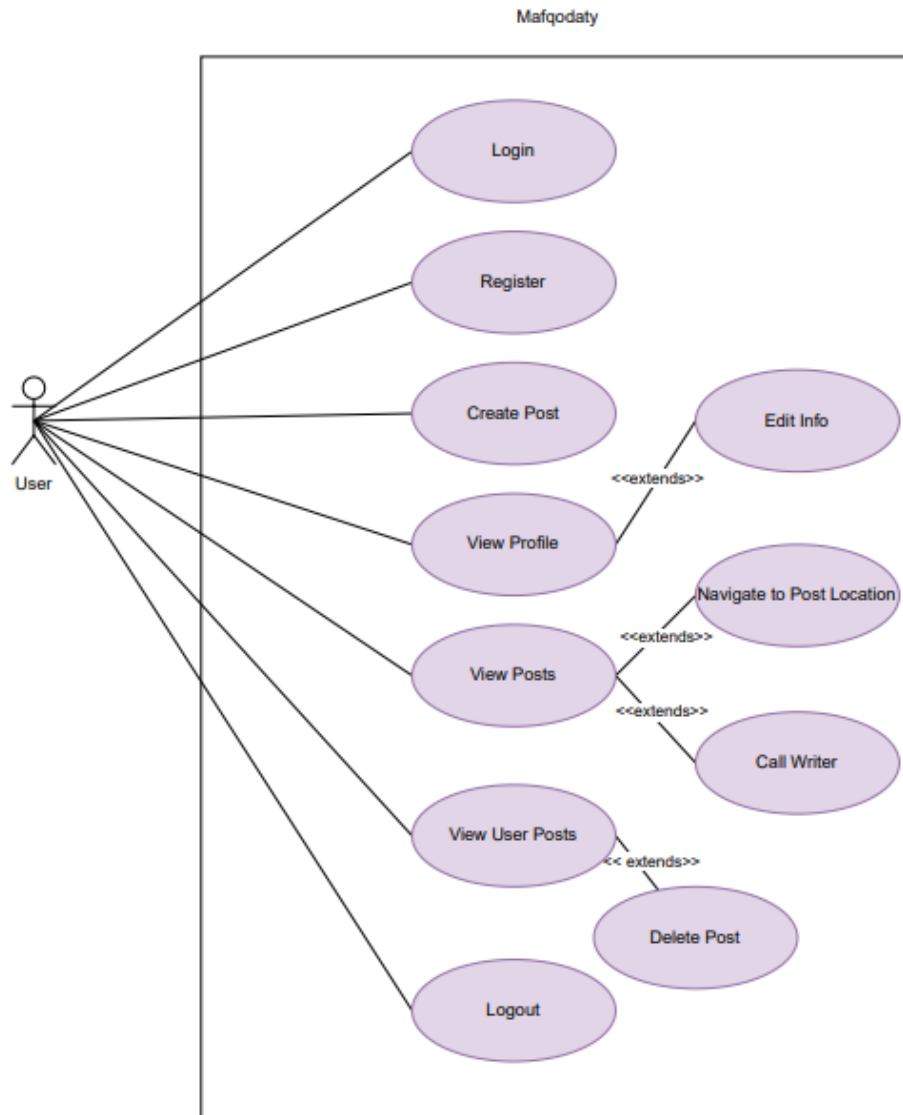
Mafqodaty

Login

Register

Create Post

Edit Info

User

<<extends>>

View Profile

Navigate to Post Location

<<extends>>

View Posts

<<extends>>

Call Writer

View User Posts

<< extends>>

Delete Post

Logout

Figure 6- Usecase Diagram

3.5 UML Class Diagram

It is the general-purpose modeling language used to visualize the system. It is a graphical language that is standard to the software industry for specifying, visualizing, constructing, and documenting the artifacts of software systems, as well as for business modeling.[9]
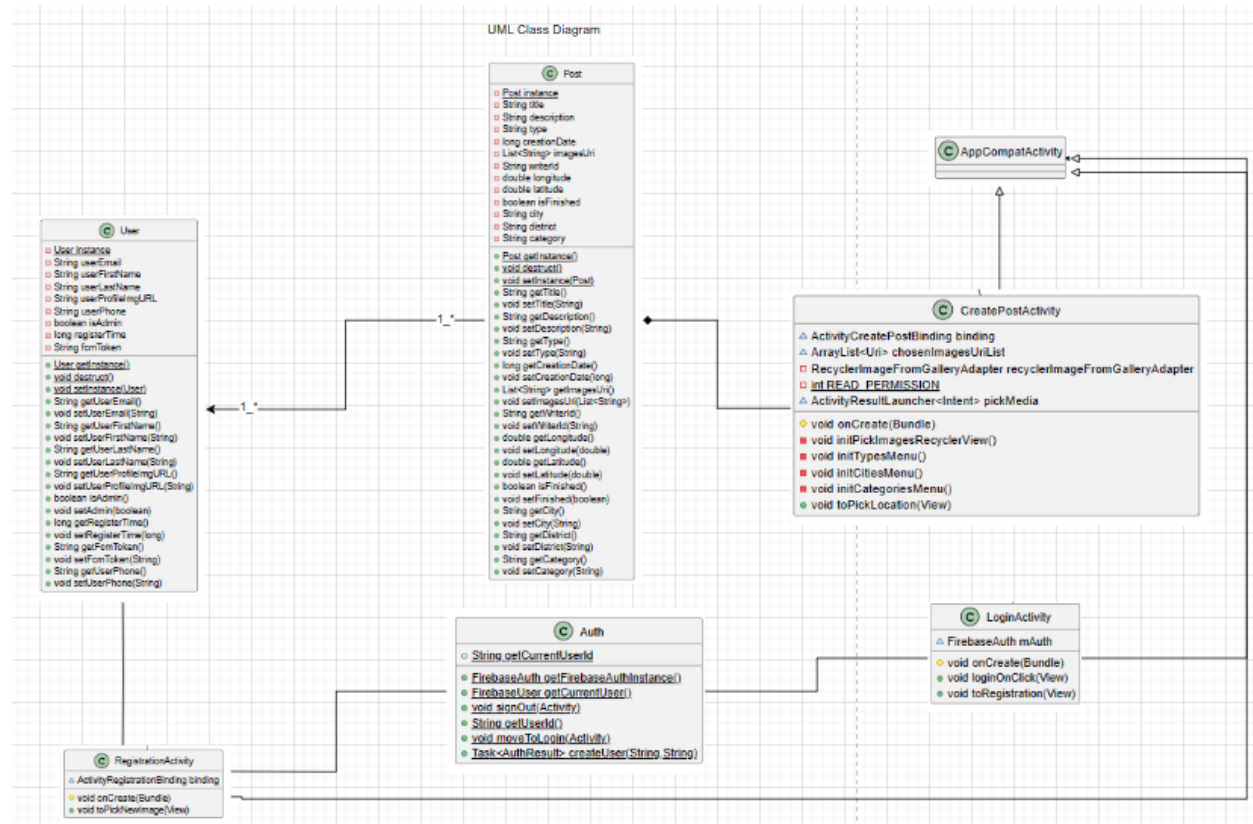
Figure 7- UML Class Diagram

CHAPTER FOUR: IMPLEMENTATION

4.1 Introduction

This Chapter will discuss the implementation of the system (App). and it provides the technical information about the system, including the system and software design decisions taken. we should note that this application has been built from scratch or by composition. we did use java language on android studio platform to develop this app and this app has been developed using firebase tools.

4.2 Hardware Specification

This app will target phones that works with android OS, these phones should include these specifications as a minimum requirement:

1- Free Storage Space 50MB (Size of App is 15MB)
2- Internet Access
3- Minimum of memory 1 GBs

*Kit Kat Android version (4.4) or above is required for this application. any increase in the aspects mentioned above, user experience will be much better in performance.

4.3 System Interfaces

In this part of the doc, we are going to show the screens of the system and we will discuss each one of them.

Android Studio is the integrated development environment for Google's Android platform. Versions of Android Studio are compatible with some Apple, Windows and Linux operating systems. With support for Google Cloud Platform and Google app integration, Android Studio offers developers a well-stocked toolkit for creating Android apps or other projects**.**

4.3.1 Login Page

This is the first page of our application which allows user to access to our application using their email and password

7:54

Login

user123@gmail.com

••••••••

Login

don't hava account? **register now**
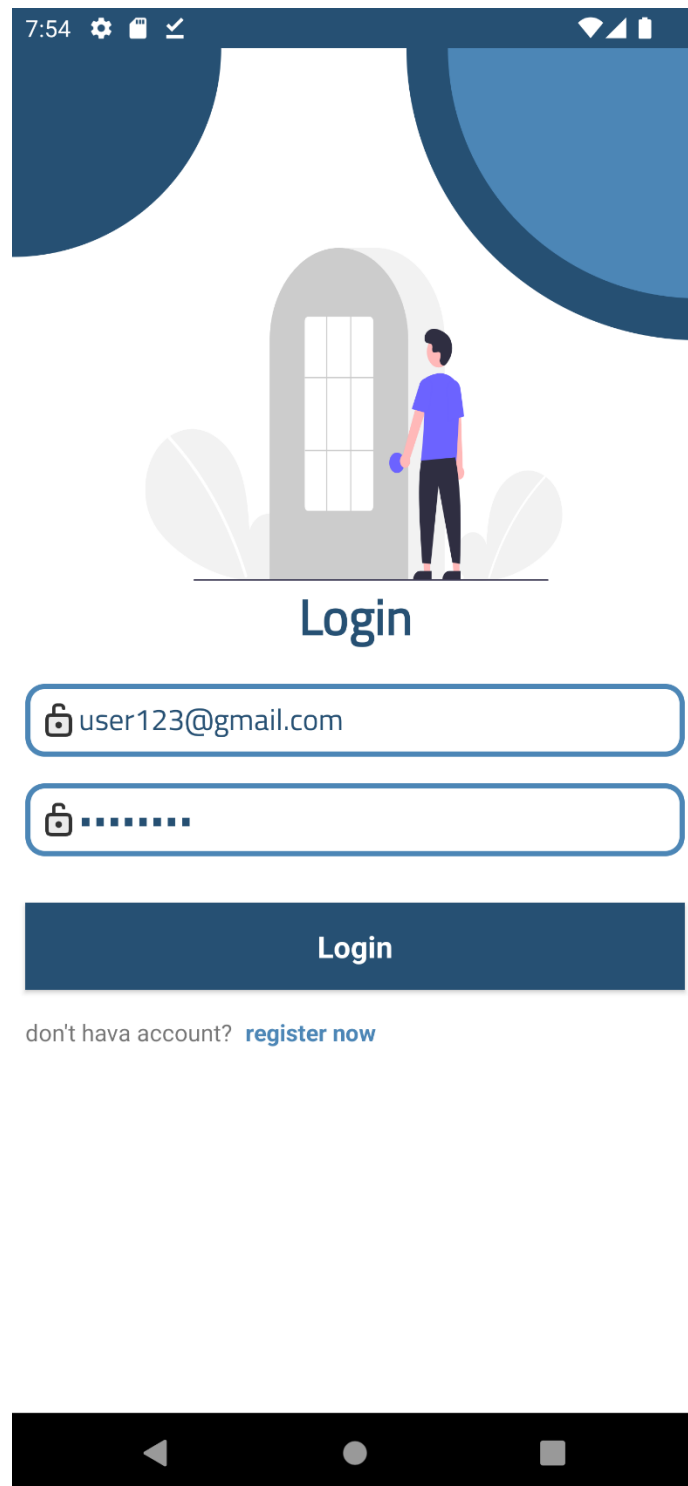
2 Signup page

This page allows user to create an account using their email, user have to insert his first name, last name, a valid email, and password.
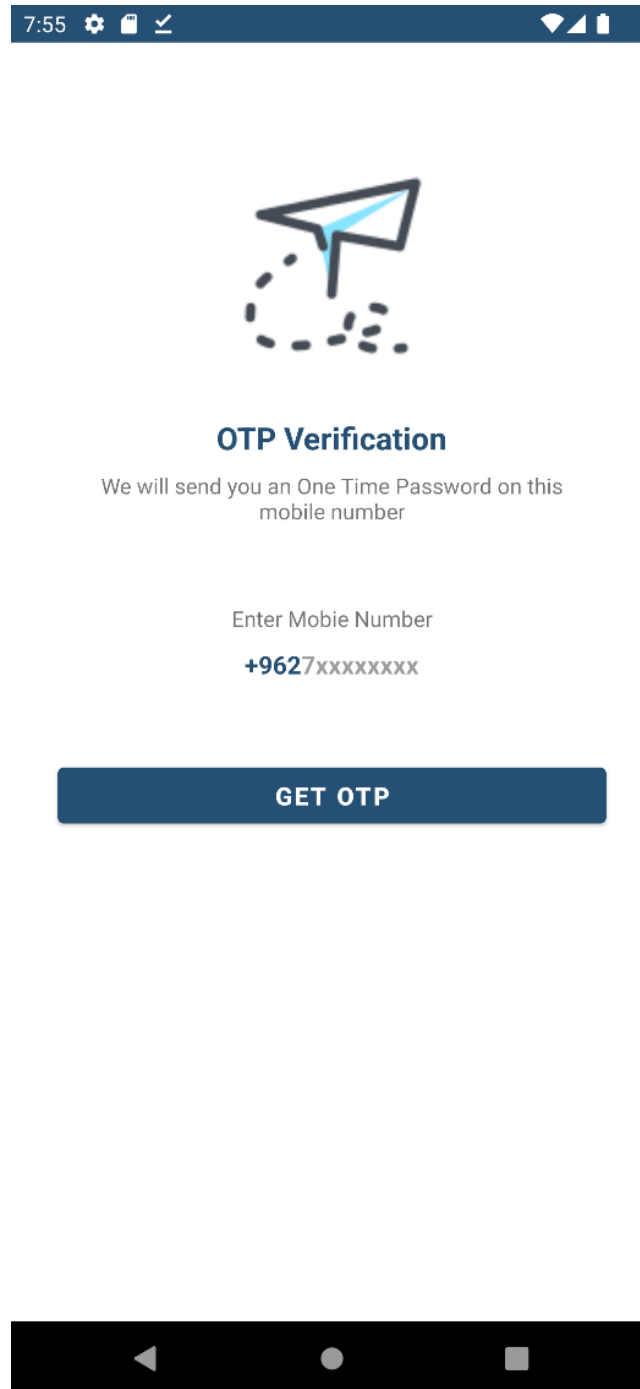
Figure 9- Signup Page

4.3.3 Enter phone Number

This page allows the user to his phone number before sending a One Time Password (OTP) for that number

Figure 10-Enter Phone Number

4.3.4 Verify OTP Code
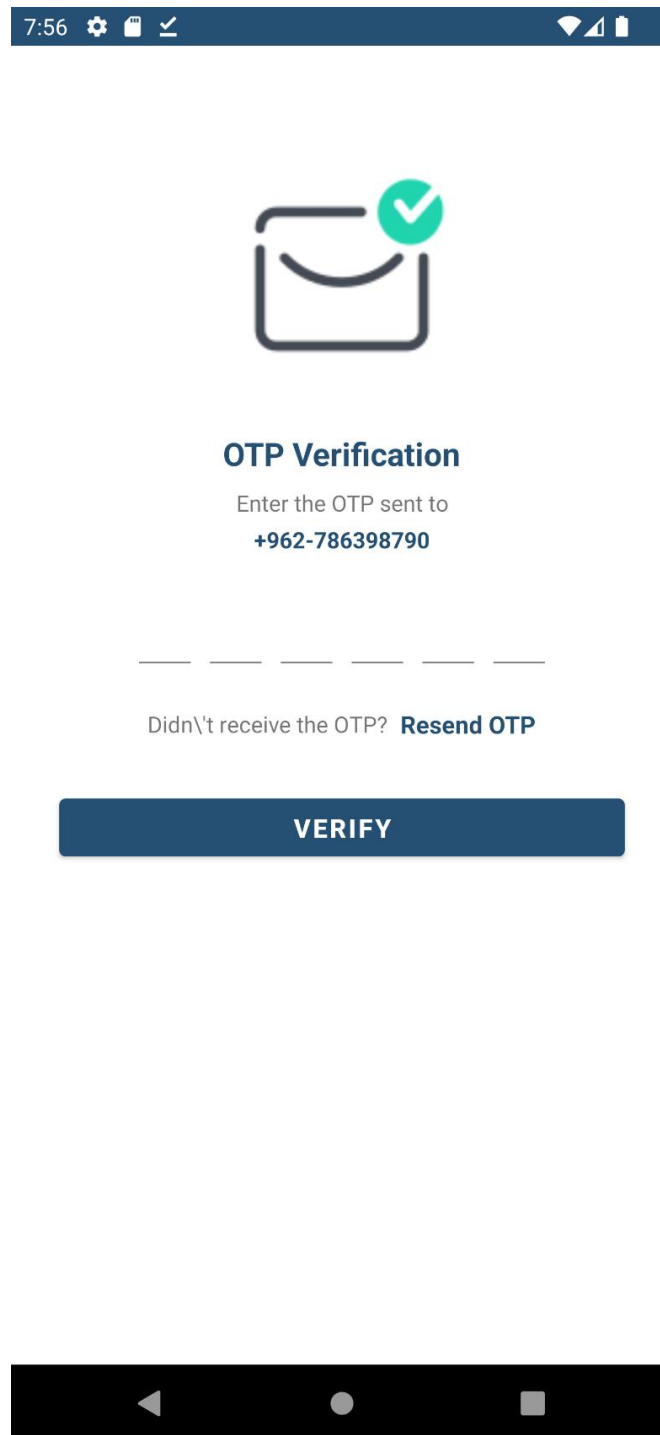After inserting a phone number, user will receive an OTP code on the number that he inserts,

Figure 11- Verify OTP Code

4.3.5 Pick Profile Picture

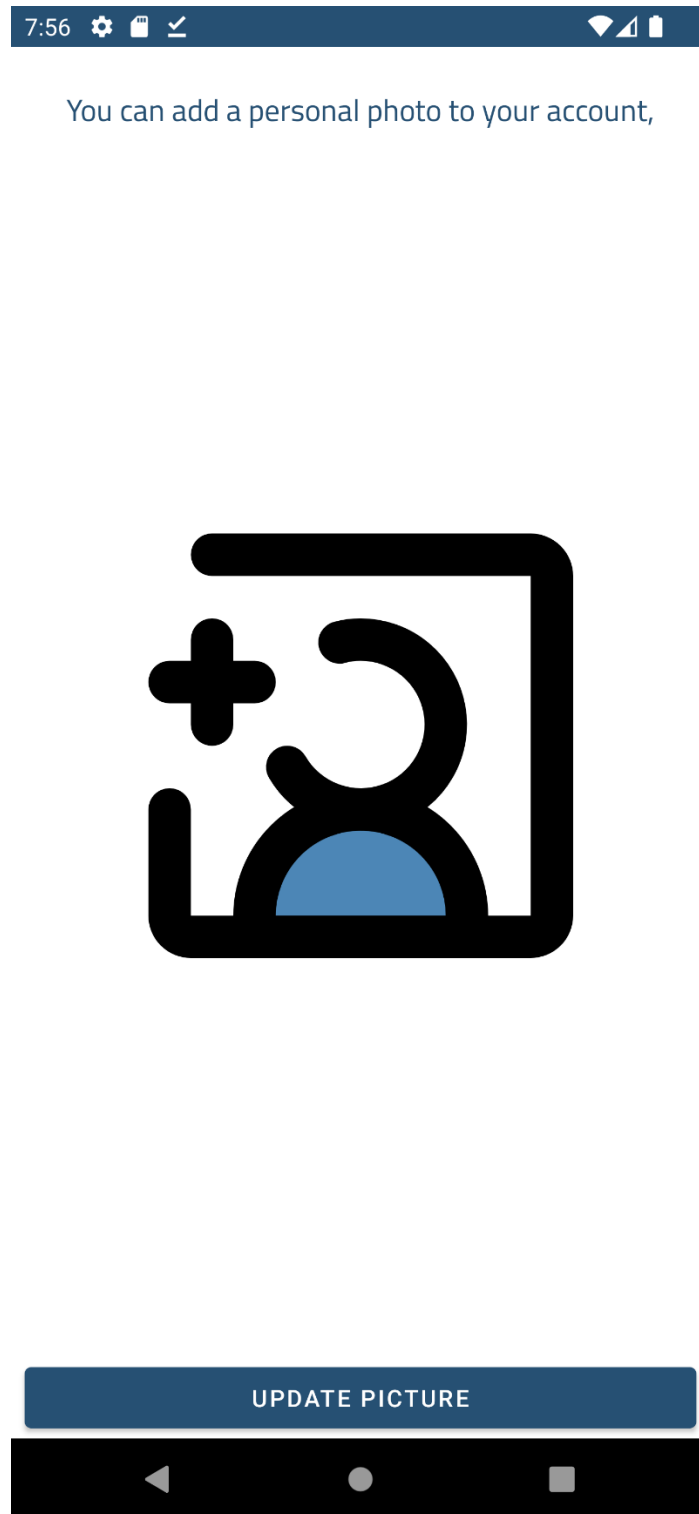This page allow user to pick a profile picture before creating his account

Figure 12- Add Profile Picture

4.3.6 Main Page

This the main page of our application, it's shown the list of post that ordered descending by creation date, user can explore posts, filter on them, can redirect directly to phone calls application to call the writer when click on button call
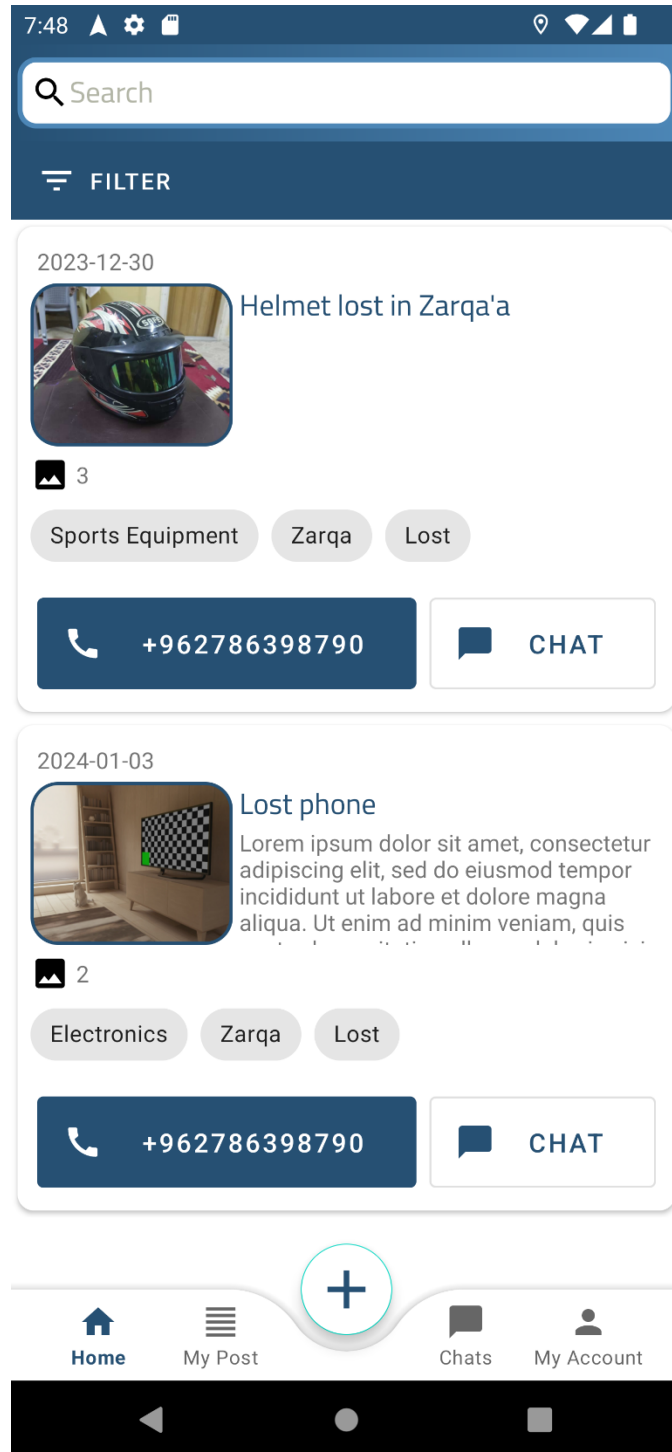

Figure 13-Main Page

4.3.7 Filtering page

When user click on "Filter" which in the top of main page, this page will show up to allow users for filtering the result that on main page

4.3.8 Post Page
This page shows the post detailed info and two button "CALL" which will redirect to phone call application, button "Navigate" which will redirect to maps application with post location .
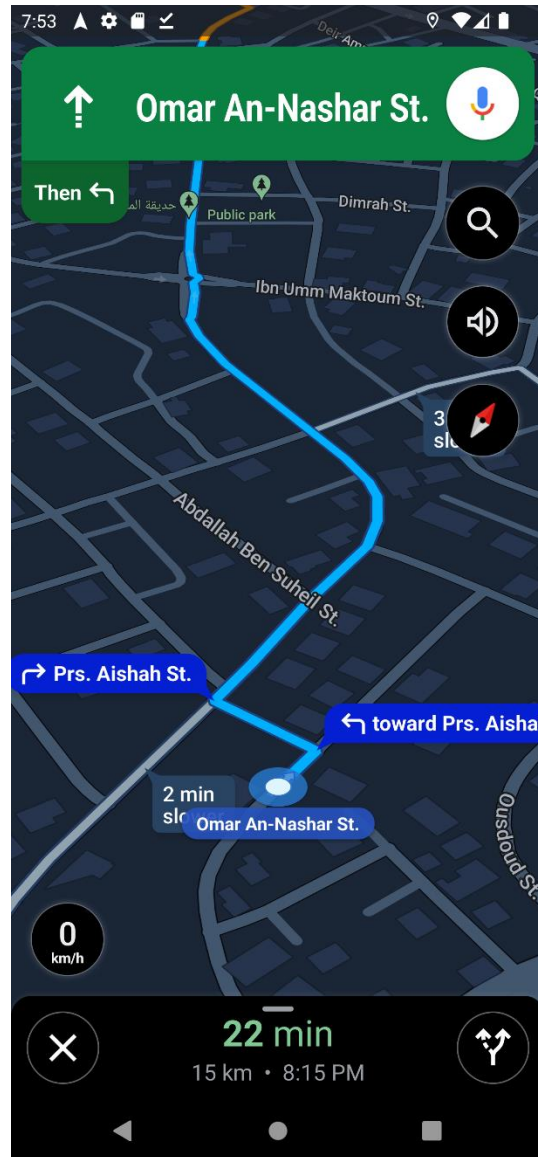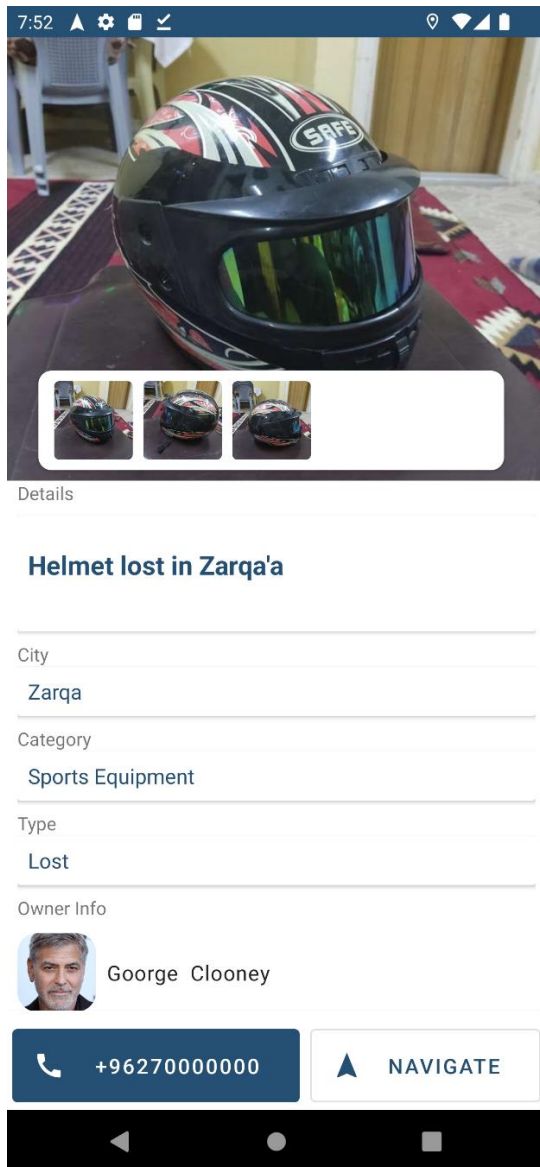

Figure 15- Post Page

4.3.9 My Post Page

This page will show the posts that the user posted ordered descending by creation date, users can explore his posts and delete them.
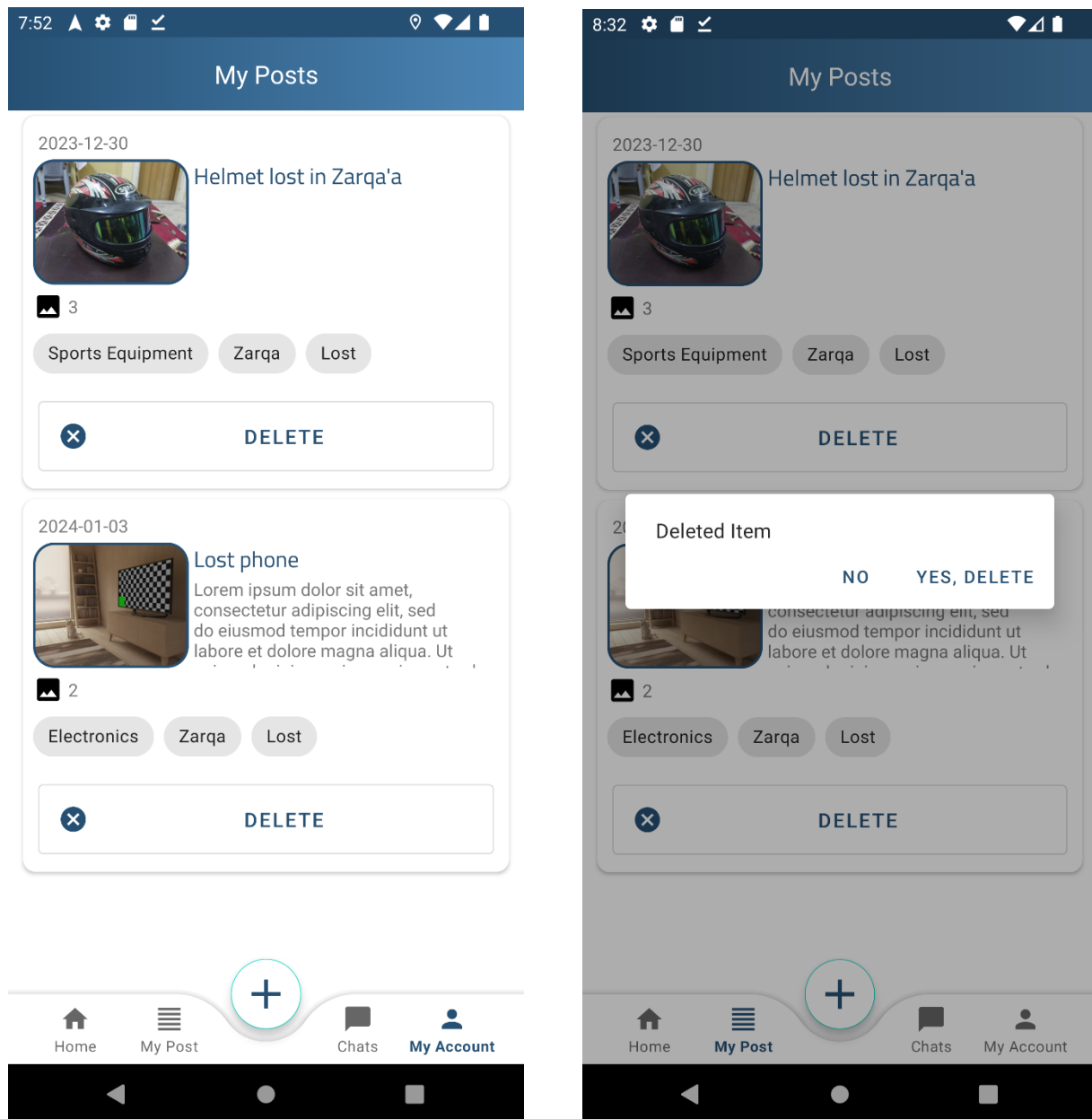

Figure 16- My Posts Page

4.3.10 Profile Page

This page shows the information of the current user, a button "EDIT" that will redirect to update profile page, button my posts which will redirect the user to "My Posts Page" and button "Logout"
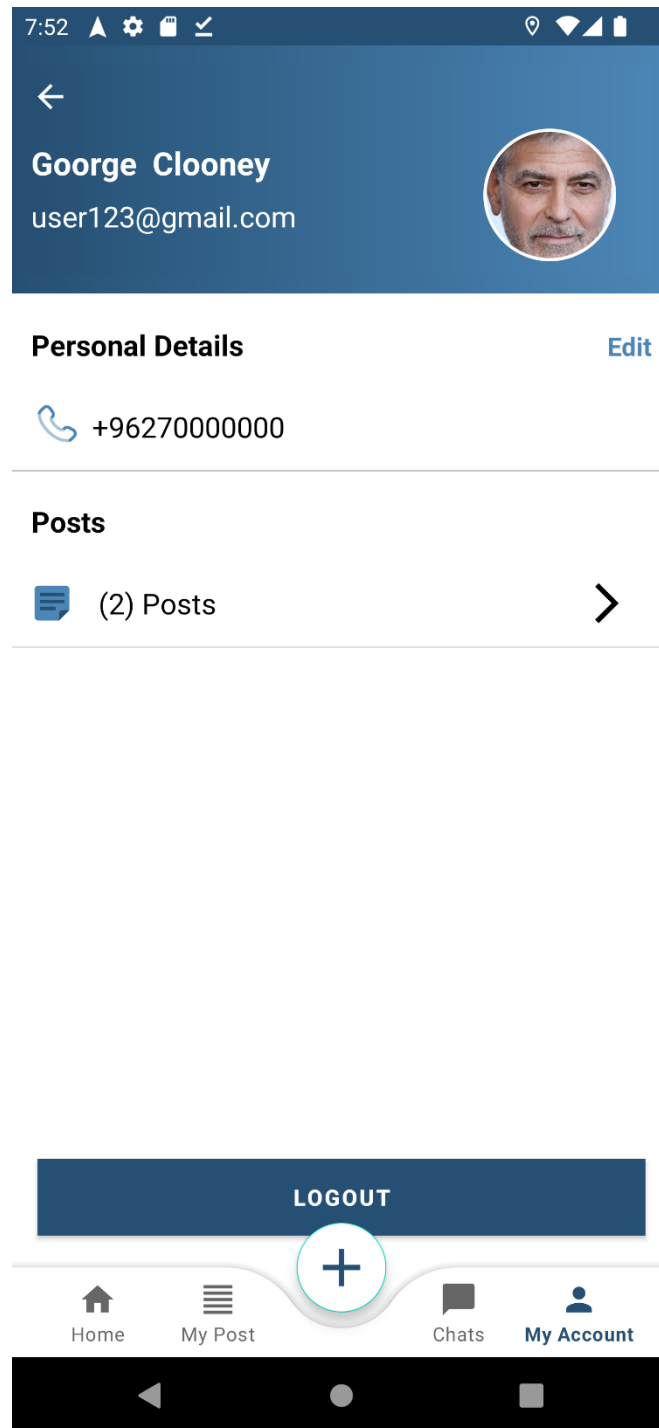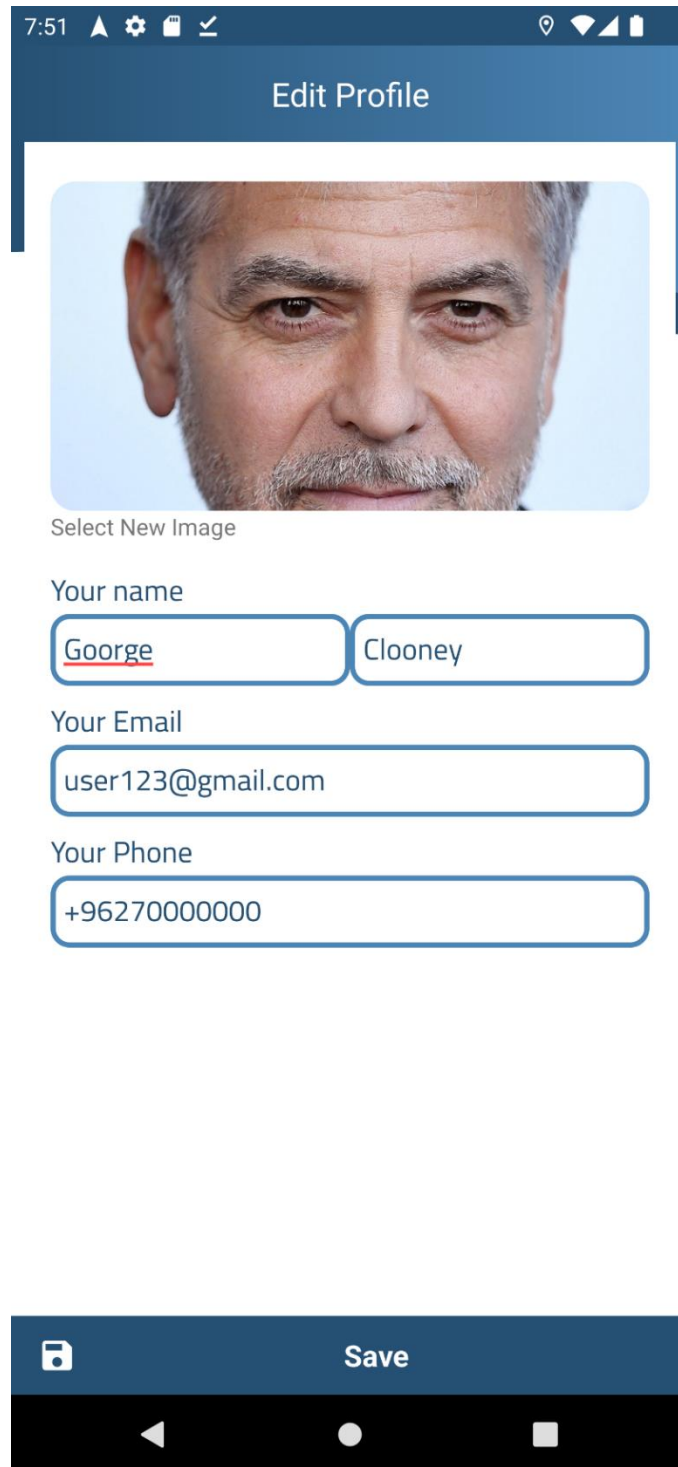
4.**3.11 Update Profile Page**
To allow user to update his account info (First name, last name, and profile image)

Figure 18-Update Profile

4.3.12 Add Post Page

This page allows user to create a new post, user can select multiple images

Figure 19-Add Post Page

4.3.13 Pick Location Page

When create a new post, user should select a location
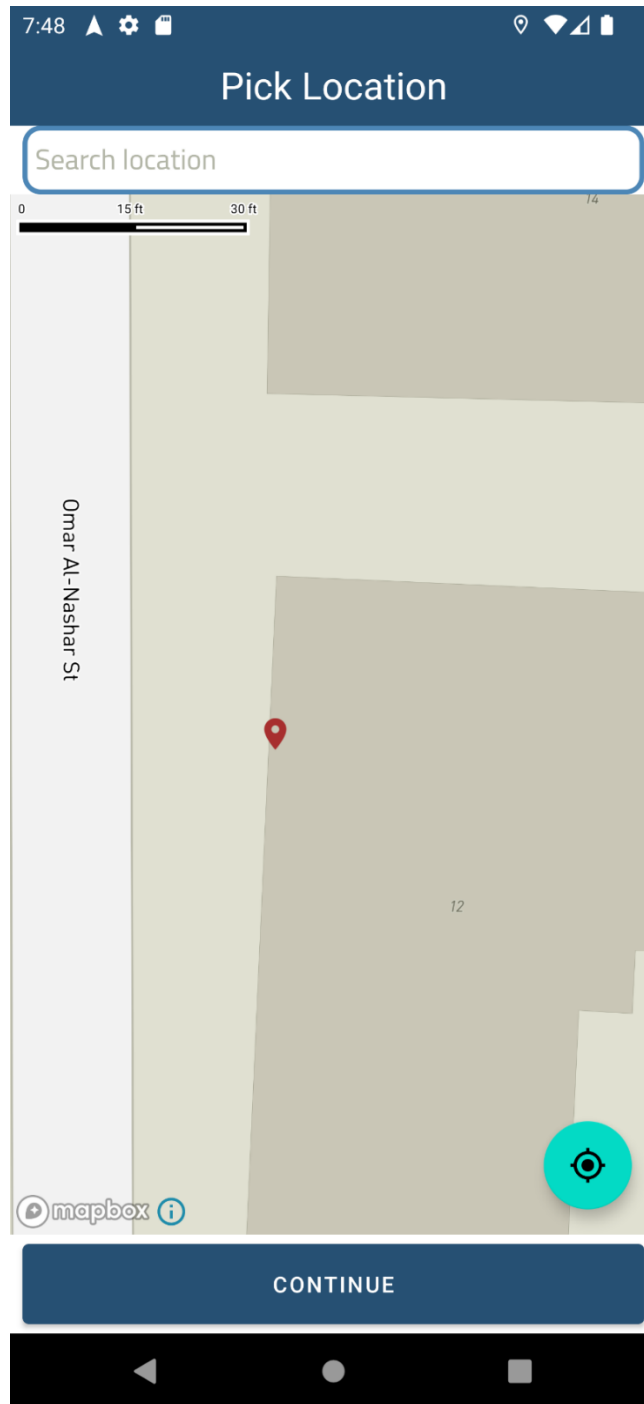
Figure 20- Pick Location Page

CHAPTER FIVE: TESTING

5.1 Result and Discussion:

"The V-model is an SDLC model where execution of processes happens in a sequential manner in a V-shape. It is also known as the Verification and Validation model. *[10]*

The V-Model is an extension of the waterfall model and is based on the association of a testing phase for each corresponding development stage. This means that for every single phase in the development cycle, there is a directly associated testing phase. This is a highly disciplined model, and the next phase starts only after the completion of the previous phase"
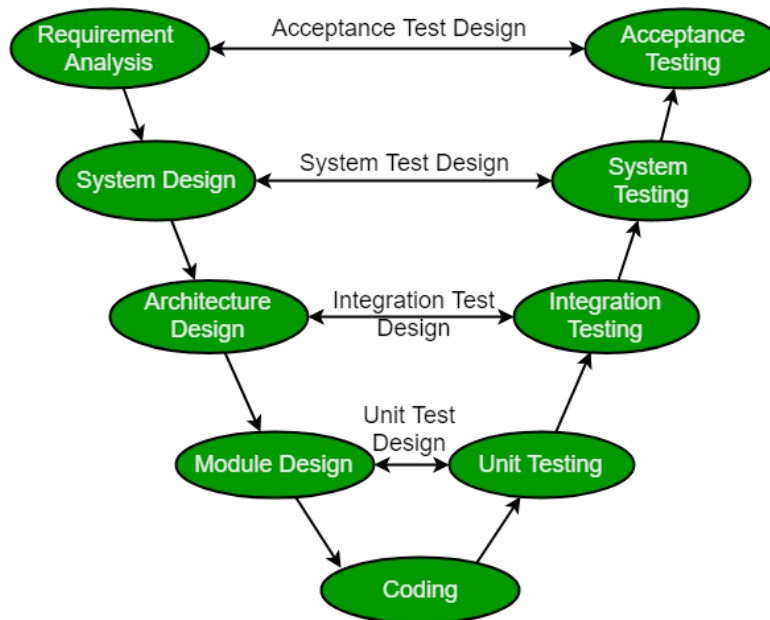
Figure 21-VModel

5.2 Testing the Functional Requirements

5.2.1 Using black box Decision table methodology (verification)

"The decision table or Cause-Effect Table is a technique, generally applied when the requirements/business rules are in the form of tables OR flow charts. The main and most important objective of Decision table testing is to ensure the overall test coverage without missing any possible relation. *[7]*

This technique has been used for all functional requirements that involve the user Entering an input because we want to make this test simulate the user experience.

Note:

☐   C1, C2...Cn are conditions (Causes)

☐   A1, A2…An are actions (Effects)

☐   R1, R2…Rm are Rules (test cases)

☐   1: means the condition is true.

☐   0: means the condition is false.

☐   x: don't care.

5.**2.1.1 Create an Account**

C1: Enter a valid name

C2: Enter a new and valid phone number

C3: Enter a new and valid email

C4: Enter a valid password

C5: Enter a valid and matched confirmation password

A1: Error message

A2: Go to the page home screen

	R1	R2	R3	R4	R5	R6
C1	1	1	1	1	1	0
C2	1	1	1	1	0	1
C3	1	1	1	0	1	1
C4	1	1	0	1	1	1
C5	1	0	1	1	1	1
A1	0	1	1	1	1	1
A2	1	0	0	0	0	0

Table 2 - Create account table

☐   A1 is to be performed when at least one of C1, C2, C3, C4, and C5 is false or all of them.

☐   A2 is to be performed when C1, C2, C3, C4, and C5 is all true.

Result:

R1: Go to the page home screen

R2: Error message (password not matches)

R3: Error message (password must be more that 8 characters)

R4: Error message (invalid email)

R5: Error message (invalid phone number)

R6: Error message (invalid email)

5.2.1.2 Login

C1: Enter an Existing email

C2: Enter a valid and authenticated password

A1: Error message

A2: Go to the page home screen

	R1	R2	R3	R4
C1	1	0	0	1
C2	0	1	x	1
A1	1	1	1	0
A2	0	0	0	1

Table 3-Login

☐   A1 is to be performed when at least one of C1, and C2 is false or all of them.

☐   A2 is to be performed when C1, and C2 is all true.

Result:

R1: Show Error Message (Password is not correct)

R2: Show Error Message (Email is not existing)

R3: Show Error Message (Email is not exiting)

R4: Go to the page home screen

5.2.1.3 Add Post

C1: Enter a valid title

C2: Enter a valid description

C3: Enter a valid city

C4: Enter a valid type

C5: Enter a valid category

A1: Show error message

A2: Add Post and back to main

	R1	R2	R3	R4	R5	R6
C1	1	x	x	x	x	0
C2	1	x	x	x	0	x
C3	1	x	x	0	x	x
C4	1	x	0	x	x	x
C5	1	0	x	x	x	x
A1	0	1	1	1	1	1
A2	1	0	0	0	0	0

Table 4-Add Post Test

☐   A1 is to be performed when at least one of C1, C2, C3, C4, and C5 is false or all of them.

☐   A2 is to be performed when C1, C2, C3, C4, and C5 is all true.

Result:

R1: Add Post and back to main

R2: Show error message (Enter a valid category)

R3: Show error message (Enter a valid type)

R4: Show error message (Enter a valid city)

R5: Show error message (Enter a valid description)

R6: Show error message (Enter a valid title)

5.2.2 User Acceptance Testing

"User acceptance testing (UAT), also called application testing or end-user testing, is a phase of software development in which the software is tested in the real world by its intended audience." [11]

A sample of End users have experienced the functionality of the application and reviewed the satisfaction of their required functionality existence.

Functional Requirements	passed	failed
As a new user, I want to create an account using my email or mobile number so that I can start using the "Mafqodaty" application.	☒	
As a user, I want to securely log in with my credentials and have the option to recover my account if needed.	☒	
As a user who lost an item, I want to easily report the details of my lost possession, including a description, location, and date.	☒	
As a user, I want to upload images or documents related to my lost item to provide more information.	☒	
As a user who found an item, I want to submit details about the found possession, including a description, location, and date.	☒	
As a user, I want to upload images or documents to support my submission of a found item.	☒	
As a user who reported a lost item, I want to be notified when there is a potential match for my lost possession.	☒	
As a user involved in a potential match, I want to communicate securely with the other user to coordinate the retrieval of the lost item.	☒	

Table 5- user acceptance testing

CHAPTER SIX: CONCLUSION AND FUTURE WORK

6.1 Conclusion

In conclusion, "Mafqodaty" presents a cutting-edge solution to the prevalent challenges of lost and found items. This Android application not only simplifies the reporting process for lost possessions but also establishes a secure and collaborative environment within the community. Addressing the shortcomings of existing systems, Mafqodaty ensures a user-friendly experience with robust security measures and privacy-centric policies.

The development of Mafqodaty was inspired by a recognition of fragmented processes, security concerns, and limited community engagement in current solutions. By prioritizing efficiency, reliability, and user satisfaction, Mafqodaty has successfully filled the void in the lost and found domain.

As users engage with Mafqodaty, they become part of a dynamic network committed to reuniting individuals with their belongings. Real-time notifications enhance the system's responsiveness, contributing to a seamless and efficient user experience. We invite everyone to join this collaborative initiative, download Mafqodaty, and contribute to a community dedicated to reconnecting people with their lost possessions. Together, we can foster a more connected and supportive world through Mafqodaty.

6.2 Future Work

1. Implement advanced matching algorithms for more precise lost and found item identification.
2. Integrate geolocation-based features for real-time information about items in the vicinity.
3. Add community engagement features like forums and discussion boards.
4. Explore integration with popular social media platforms to broaden the application's reach.
5. Continuously enhance security measures to safeguard user data and communications.
6. Explore the integration of augmented reality (AR) for interactive item location and identification.
7. Collaborate with local authorities and law enforcement agencies to create a more extensive network.
8. Establish channels for gathering user feedback and conduct regular updates based on user preferences.
9. Prioritize implementation of accessibility features for users with diverse needs.
10. Consider expanding the application to additional platforms, such as iOS or web.
11. Create in app chat for contact between users.

References

[1] https://www.resmo.com/blog/what-is-firebase

[2] https://www.java.com/en/download/help/whatis_java.html

[3] https://developer.android.com/studio/intro

[4] https://www.atlassian.com/agile/project-management/gantt-chart

[5] https://mode.com/blog/what-is-agile-analytics

[6] https://www.smartdraw.com/entity-relationship-diagram/#:~:text=An%20ERD%20visualizes%20the%20relationships,the%20logical%20structure%20of%20databases.

[7] https://www.geeksforgeeks.org/what-is-a-flowchart-and-its-types/

[8] https://megainterview.com/use-case-diagram-interview-questions/

[9] https://www.visual-paradigm.com/guide/uml-unified-modeling-language/what-is-uml/

[10] https://www.geeksforgeeks.org/software-engineering-sdlc-v-model/

[11] https://www.panaya.com/blog/testing/what-is-uat-testing/

الخلاصة

وفي الختام، تقدم "مفقوداتي" حلاً متطورًا للتحديات السائدة المتمثلة في الأشياء المفقودة والموجودة. لا يعمل تطبيق هذا على تبسيط عملية الإبلاغ عن الممتلكات المفقودة فحسب، بل ينشئ أيضًا بيئة آمنة وتعاونية داخل المجتمع. من خلال معالجة أوجه القصور في الأنظمة الحالية، يضمن برنامج "مفقوداتي" تجربة سهلة الاستخدام مع تدابير أمنية قوية وسياسات تركز على الخصوصية..

عندما يتفاعل المستخدمون مع "مفقوداتي"، يصبحون جزءًا من شبكة ديناميكية ملتزمة بجمع شمل الأفراد بممتلكاتهم. تعمل الإشعارات في الوقت الفعلي على تحسين استجابة النظام، مما يساهم في توفير تجربة مستخدم سلسة وفعالة. نحن ندعو الجميع للانضمام إلى هذه المبادرة التعاونية، وتنزيل "مفقوداتي"، والمساهمة في مجتمع مخصص لإعادة ربط الأشخاص بممتلكاتهم المفقودة. معًا، يمكننا تعزيز عالم أكثر تواصلاً ودعمًا من خلال تطبيقنا.