

# **AUTONOMOUS FIRST AID DRONE (AFAD)**

**YUSRA TAHIR  
SHAHZEB NOOR**



**DEPARTMENT OF COMPUTER SCIENCES  
COMSATS UNIVERSITY ISLAMABAD, WAH CAMPUS  
WAH CANTT – PAKISTAN**

**SESSION 2019-2023**

# **AUTONOMOUS FIRST AID DRONE (AFAD)**

*Undertaken By:*

**YUSRA TAHIR**

REG. NO. CIIT/FA19-BSE-100/WAH

**SHAHZEB NOOR**

REG. NO. CIIT/FA19-BSE-138/WAH

*Supervised by:*

**Ms. MEHWISH MUKHTAR**



A DISSERTATION SUBMITTED AS A PARTIAL FULFILLMENT OF THE  
REQUIREMENTS FOR THE DEGREE OF BACHELORS IN COMPUTER SCIENCE /  
SOFTWARE ENGINEERING

**DEPARTMENT OF COMPUTER SCIENCES  
COMSATS UNIVERSITY ISLAMABAD, WAH CAMPUS  
WAH CANTT – PAKISTAN**

**SESSION 2019-2023**

## **FINAL APPROVAL**

The final approval page will be provided by the department after the final evaluation.

## **DEDICATION**

We devote this project to **Allah Almighty** who gave us strength, knowledge, skills, wisdom, and health to complete this project, also we devote to our **beloved family**, who have always been our pillar of strength, support and encouragement throughout our academic journey. We would like to dedicate this Final Year Project to our amazing supervisor, **Ms. Mehwish Mukhtar** who has been a constant source of guidance, motivation and inspiration throughout this journey. Your expertise, insights and constructive feedback have been instrumental in shaping this project, and I am grateful for your valuable contributions. Last but not least, we dedicate this report to the countless individuals who have contributed to the field of Software Engineering, and whose work has paved the way for future generations. It is because of your groundbreaking research, innovation and hard work that we have been able to make progress in this field. Thank you for your dedication and commitment to advancing technology, and for inspiring us to do the same.

# **ACKNOWLEDGEMENT**

In the name of Allah, the most Gracious, Beneficent and Merciful, who has given the strength and ability to complete this project.

We would like to express our deepest gratitude to all those who have contributed to the completion of this Final Year Project. First and foremost, we would like to thank our project supervisor **Ms. Mehwish Mukhtar**, for her guidance, support, and valuable insights throughout the entire project. Her feedback and suggestions were instrumental in shaping our approach and achieving our objectives. We would also like to extend our appreciation to the faculty and staff of the **Computer Science Department** for their unwavering support and resources. Their dedication and expertise have greatly enhanced our learning experience and helped us develop our skills as software engineers. Finally, we would like to thank our **families** for their unwavering support and encouragement throughout our academic journey.

***YUSRA TAHIR***

***SHAHZEB NOOR***

# **PROJECT BRIEF**

PROJECT NAME	AUTONOMOUS FIRST AID DRONE
(AFAD)	
ORGANIZATION NAME	COMSATS UNIVERSITY ISLAMABAD, WAH CAMPUS
OBJECTIVE	FASTER DELIVERY OF FIRST AID REDUCE HUMAN RISK IMPROVED SURVIVAL RATES.
UNDERTAKEN BY	YUSRA TAHIR SHAHZEB NOOR
SUPERVISED BY	Ms. MEHWISH MUKHTAR LECTURER COMPUTER SCIENCE COMSATS UNIVERSITY ISLAMABAD, WAH CAMPUS
STARTED ON	OCTOBER 7, 2022
COMPLETED ON	JUNE 13, 2023
COMPUTER USED	HP CORE I5, PROBOOK
SOURCE LANGUAGE	DART LANGUAGE
OPERATING SYSTEM	WINDOWS 10
TOOLS USED	ANDROID STUDIO 11.0.13 MISSION PLANNER 1.3.79 ANDROID EMULATOR DRAW.IO

# **ABSTRACT**

Autonomous First Aid Drone (AFAD) system is designed to provide a rapid response to emergency situations where access to medical aid is restricted, such as in natural disasters or remote areas. The AFAD can deliver medical supplies, such as first aid kits and defibrillators, to the site of an emergency faster than traditional methods of transport. The mobile application allows the user to provide location where aid is required. Mission planner is used to plan a mission through which AFAD can reach autonomously on provided location. The system is designed to reduce human risk, improve survival rates, and provide cost-effective and accessible emergency medical aid. The report outlines the technical specifications of the AFAD system, including its hardware and software components, and presents results from simulations and field tests. Overall, the AFAD system with mobile application has the potential to significantly improve the efficiency and effectiveness of emergency medical aid delivery.

# TABLE OF CONTENTS

<b>1</b>	<b>INTRODUCTION .....</b>	<b>1</b>
1.1	System Introduction .....	1
1.2	Background of the System .....	1
1.3	Objectives of the System.....	2
1.4	Significance of the System.....	2
<b>2</b>	<b>REQUIREMENT SPECIFICATIONS.....</b>	<b>3</b>
2.1	Product Scope.....	3
2.2	Product Description.....	3
2.2.1	Product Perspective.....	3
2.2.2	Product Functionality.....	4
2.2.3	Users and Characteristics.....	5
2.2.4	Operating Environment.....	5
2.3	Specific Requirements.....	6
2.3.1	Functional Requirements .....	6
2.3.2	Behavioral Requirements.....	9
2.3.3	External Interface Requirements.....	17
2.4	Non-functional Requirements .....	26
2.4.1	Performance Requirements .....	26
2.4.2	Safety and Security Requirements .....	26
2.4.3	Software Quality Attributes .....	27
<b>3</b>	<b>DESIGN SPECIFICATIONS.....</b>	<b>28</b>
3.1	Introduction .....	28
3.2	Composite Viewpoint.....	28
3.2.1	Composite Diagram .....	28
3.2.2	Package Diagram .....	29
3.3	Logical Viewpoint.....	30
3.3.1	Class Diagram.....	30
3.4	Information Viewpoint.....	31
3.4.1	Entity Relation Diagram .....	31
3.5	Interaction Viewpoint.....	32
3.5.1	Admin Sequence Diagram: -.....	32
3.5.2	User Sequence Diagram.....	33

3.6	State Dynamics Viewpoint.....	34
3.7	Algorithmic Viewpoint .....	35
3.7.1	Algorithmic Viewpoint of User .....	35
3.7.2	Algorithmic Viewpoint of Admin.....	37
<b>4</b>	<b>DEVELOPMENT AND TOOLS .....</b>	<b>40</b>
4.1	Introduction .....	41
4.2	Development Plan .....	41
4.2.1	Project Modules Distribution.....	41
4.3	Development Tools .....	43
4.4	Conclusion and Future Work/Extensions.....	44
<b>5</b>	<b>QUALITY ASSURANCE .....</b>	<b>44</b>
5.1	Introduction .....	45
	Documentation Items: - .....	45
	Mobile Application Items: - .....	45
	Hardware Items: -.....	45
5.2	Traceability Matrix.....	45
5.3	Test Plan.....	46
5.3.1	Test Plan Identifier.....	46
5.3.2	References.....	47
5.3.3	Introduction.....	47
5.3.4	Objectives: - .....	47
5.3.5	Software Risk Issues .....	47
5.3.6	Testing Approach (Strategy).....	48
5.3.7	Testing Levels: - .....	48
5.3.8	Features to Be Tested.....	49
5.3.9	Test Cases .....	50
<b>6</b>	<b>USER MANUAL.....</b>	<b>64</b>
6.1	Introduction .....	64
6.2	Hardware/Software Requirements for the System .....	64
6.2.1	Hardware Requirements.....	64
6.2.2	Software Requirements.....	64
6.2.3	Other requirements.....	64
6.3	Installation guide for Application .....	65
6.4	Operating Manual.....	65

6.4.1	Mobile Application User Manual .....	66
6.4.2	Drone User Manual (Autonomous Flight).....	71
<b>7</b>	<b>GLOSSARY .....</b>	<b>79</b>
7.1	Definition, Abbreviation, Acronyms.....	79

## LIST OF FIGURES

Figure 2.1 Product Perspective .....	4
Figure 2.2 AFAD Use Case Diagram .....	9
Figure 2.3 Registration Use Case.....	10
Figure 2.4 Login Use Case.....	10
Figure 2.5 5Provide Location Use Case .....	11
Figure 2.6 Request first Aid Use Case.....	11
Figure 2.7 Access Profile Use Case .....	12
Figure 2.8 View History Use case .....	12
Figure 2.9 Login Admin Use Case .....	13
Figure 2.10 Get Location Use Case .....	13
Figure 2.11 View First Aid Details Use Case.....	14
Figure 2.12 Manage Aid Requests Use Case .....	15
Figure 2.13 Configure Drone Use case.....	15
Figure 2.14 Arm Drone Use Case.....	16
Figure 2.15 Splash Screen    Figure 2.16 Get Started Screen Figure 2.17 Enter Number Screen	17
Figure 2.18 OPT verification      Figure 2.19 Enter OTP notification	17
Figure 2.20 OTP	
Figure 2.21 Home Screen .....	18
Figure 2.22 User Registration .....	18
Figure 2.23 User Registration Form .....	18
Figure 2.24 Request sent.....	18
Figure 2.25 Emergency details and Location .....	18
Figure 2.26 Emergency call and message .....	18
Figure 2.27 Feedback screen.....	19
Figure 2.28 Drawer .....	19
Figure 2.29 Aid Request Notification.....	19
Figure 2.30 Pending Request .....	19
Figure 2.31 User Update Screen .....	19
Figure 2.32 User Profile Screen.....	19
Figure 2.33 Admin Login .....	20
Figure 2.34 All requests .....	20
Figure 2.35 Admin Dashboard.....	20
Figure 2.36 New Request.....	20
Figure 2.37 Calculate Distance .....	20
Figure 2.38 Request accepted notification.....	20
Figure 2.39 Track Order .....	21
Figure 2.40 View All users .....	21
Figure 2.41 Request Rejected Notification .....	21
Figure 2.42 Mission Planner Screen .....	21
Figure 2.43 Start Screen.....	22

Figure 2.44 Plan Mission Interface .....	22
Figure 2.45 Drone Frame .....	23
Figure 2.46 Brushless DC motor .....	23
Figure 2.47 Electric Speed Controller .....	23
Figure 2.48 APM flight Controller .....	24
Figure 2.49 m8n GPS.....	24
Figure 2.50 Lipo battery .....	24
Figure 2.51 Propellers .....	24
Figure 2.52 AFAD circuit Diagram .....	25
Figure 2.53 Quad Directions.....	25
Figure 3.1 Composite Diagram.....	28
Figure 3.2 Package Diagram.....	29
Figure 3.3 AFAD Class Diagram.....	30
Figure 3.4 Entity Relation Diagram.....	31
Figure 3.5 Sequence Diagram (Admin) .....	32
Figure 3.6 Sequence Diagram (User).....	33
Figure 3.7 AFAD State Dynamic Diagram.....	34
Figure 5.1 Traceability Matrix (User FR) .....	46
Figure 5.2 Traceability Matrix (Admin FR) .....	46
Figure 6.1 User Manual (Enter number).....	66
Figure 6.2 User Manual (Splash Screen) .....	66
Figure 6.3 User Manual (Home Screen) .....	66
Figure 6.4 User Manual (OTP Verification).....	66
Figure 6.5 User Manual (Enter OTP) .....	66
Figure 6.6 (Cloud OTP notification).....	66
Figure 6.7 User Manual(User Registration).....	67
Figure 6.8User Manual (Drawer Screen).....	67
Figure 6.9 User Manual (Registration Form) .....	67
Figure 6.10 User Manual (Emergency Button).....	67
Figure 6.11 User Manual (Details Entry) .....	67
Figure 6.12User Manual (Get location and emergency details) .....	67
Figure 6.13User Manual (Request sent) .....	68
Figure 6.14 User Manual (Feed Back).....	68
Figure 6.15User Manual (Pending requests) .....	68
Figure 6.16 User Manual (Update Profile) .....	68
Figure 6.17 User Manual (View Profile) .....	68
Figure 6.18 User Manual(Admin Dashboard) .....	69
Figure 6.19 User Manual (Admin Login) .....	69
Figure 6.20 User Manual (Aid Request Notification) .....	69
Figure 6.21 ( Request Rejected Notification) .....	69
Figure 6.22 User Manual(Requests Screen) .....	69
Figure 6.23 Request Accepted Notification.....	69
Figure 6.24 Google Map .....	70

Figure 6.25 Track Order .....	70
Figure 6.26 Requests history.....	70
Figure 6.27 View all users .....	70
Figure 6.28 Distance Calculation.....	70
Figure 6.29 Flight Controller connection.....	71
Figure 6.30 USB port in Flight Controller.....	71
Figure 6.31 Startup Mission Planner .....	72
Figure 6.32 Mission Planner Interface.....	72
Figure 6.33 Build Connection.....	73
Figure 6.34 Connected .....	73
Figure 6.35 GPS Locked.....	74
Figure 6.36 PLAN mission .....	74
Figure 6.37 Selecting WP1 .....	75
Figure 6.38 Selecting WP2 .....	75
Figure 6.39 Parameters .....	76
Figure 6.40 Selecting Land point.....	76
Figure 6.41 Writing WP's to Flight Controller .....	77
Figure 6.42 Plug Battery Terminals.....	77
Figure 6.43 Battery Plugs .....	77
Figure 6.44 Transmitter FLYSKY FS i6s.....	78
Figure 6.45 AFAD .....	78
Figure 6.46 Autonomous Flight.....	78

## LIST OF TABLES

Table 2.1 Registration Use Case Description .....	10
Table 2.2 Login Use Case Description .....	10
Table 2.3 Provide Location Use Case Description .....	11
Table 2.4 Request first Aid Use Case Description .....	11
Table 2.5 Access Profile Use Case Description.....	12
Table 2.6 View History Use case Description .....	13
Table 2.7 Login Admin Use Case Description .....	13
Table 2.8 Get Location Use Case Description.....	14
Table 2.9 View First Aid Details Use Case Description .....	14
Table 2.10 Manage Aid Requests Use Case Description .....	15
Table 2.11 Configure Drone Use case Description .....	15
Table 2.12 Arm Drone Use Case Description .....	16
Table 3.1 User class , Attributes , Login method.....	35
Table 3.2 ProvideLocation Pseudo Code.....	36
Table 3.3 AddCase Pseudo Code.....	36
Table 3.4 ProvideFeedback Pseudo code.....	36
Table 3.5 ViewRequest Pseudo Code .....	36
Table 3.6 UpdateProfile Pseudo code.....	37
Table 3.7 User Admin , Attributes , Login method .....	38
Table 3.8 GetLocation Pseudo code .....	38
Table 3.9 ManageRequests Pseudo Code .....	39
Table 3.10 ShowFeedback Pseudo Code .....	39
Table 3.11 ShowUsers Pseudo Code .....	39
Table 3.12 ConfigureDrone Pseudo Code .....	40
Table 4.1 FYP work distribution .....	41
Table 4.2 Project Development Plan .....	42
Table 5.1 Test Plan Identifier.....	46
Table 5.2 Tools for testing .....	48
Table 5.3 Modules for Unit Test.....	48
Table 5.4 Modules for Integration Testing .....	49
Table 5.5 Test Case Phone Number Signup and OTP Verification.....	51
Table 5.6 Signup and OTP Test Case .....	51
Table 5.7 OTP Verification Timeout Test Case .....	51
Table 5.8 OTP Resend Functionality Test Case .....	52
Table 5.9 Location Coordinates Retrieval Test Case.....	52
Table 5.10 Adding Medical Supplies Test Case .....	53
Table 5.11 Send Location and Emergency Details to Admin Test Case .....	53
Table 5.12 View and Update Customer Profile Test Case .....	54
Table 5.13 Feedback Submission with 5-star Rating Test Case.....	55
Table 5.14 Feedback Submission without Rating Test Case .....	56
Table 5.15 Notification on Request Acceptance Test Case .....	56

Table 5.16 Notification on Request Rejection Test Case .....	57
Table 5.17 Admin Login Test Case .....	57
Table 5.18 Invalid Admin Login Test Case.....	58
Table 5.19 Accept First Aid Request Test Case .....	58
Table 5.20 Reject First Aid Request Test Case .....	59
Table 5.21 Admin View Requests Test Case.....	59
Table 5.22 View Feedbacks Test Case .....	60
Table 5.23 Request Notification Test Case.....	60
Table 5.24 Drone Configuration in Ardupilot Mission Planner Test Case.....	61
Table 5.25 Plan Mission Test Case.....	61
Table 5.26 Place First Aid Kit and Arm Drone Test Case.....	63
Table 7.1 Acronyms.....	79

# 1 INTRODUCTION

The Autonomous First Aid Drone (AFAD) is a prototype developed to assist in saving human lives by delivering first aid kits to emergency situations quickly. The system comprises three main components: hardware, a service station (mission planner), and a mobile application. The AFAD system is cost-effective, faster, safer, and accessible, providing quick access to life-saving equipment and medical supplies in areas where traditional methods of transport are inaccessible. The system's objectives are to improve survival rates, reduce human risk, and increase accessibility. The system's significance lies in its potential to improve the efficiency and effectiveness of emergency medical aid delivery in Pakistan, especially in areas with limited access to medical aid due to natural disasters or security challenges.

## 1.1 System Introduction

The use of first aid delivery is an increasing trend worldwide, further enhanced by the COVID pandemic. In this context, a prototype of '**Autonomous First Aid Drone**' is developed in order to assist ambulances in saving human lives. The developed prototype takes less time to reach the emergency situation spot and helps to measure and transmit real time health parameters of patient to the ambulance where they can be analyzed in the lab view software. This prototype helps in major irreplaceable contribution in saving the life of a patient and thus fulfills the purpose of 'lifesaving'.

AFAD has three main components, hardware quadcopter drone, service station (mission planner) and mobile application named as "Drone Ambulance". In Hardware we have a drone like a quadcopter to deliver a first aid kit to user. In drone mobile application is used for getting user location and to put requirements for first aid. And through Ardupilot mission planner we'll plan the mission of drone.

## 1.2 Background of the System

Although there are many drone projects available and used in various field like Fire Fighting Surveillance Agriculture etc. in these field drones is very useful and reduce human power, but the existing project only focuses on one technique they are using remote controls to fly their drone. Our project is slightly different from this because we not only enhance the performance but also create an autonomous drone to deliver the first aid and it may be used in the replacement of these existing systems. The purpose of this project is to supply the First Aid to the people in case of emergency or any road accident. The transportation of First Aid very difficult for a man to hard reach areas and the areas where peoples live without mobility.

## 1.3 Objectives of the System

The objectives of an Autonomous First Aid Drone (AFAD) are:

- **Fast Response:** Drones offer a faster first aid supply in case of any emergency.
- **Cost-effective:** Using electric and autonomous drones for first aid can result in lower cost and can save lives.
- **Increase Survival Rate:** The AFAD can help improve survival rates by providing quick access to life-saving equipment and supplies.
- **Safer delivery:** Especially at times of pandemics, the need for contactless first aid supply increased to reduce the spread of diseases.
- **Reduce Human Risk:** The use of an AFAD in emergency situations can reduce the risk of harm to emergency responders and other individuals involved in the rescue effort.
- **Accessibility:** The AFAD can reach areas that are difficult to access by road or other means of transport, making it possible to provide medical aid to people who would otherwise not have access to it.

## 1.4 Significance of the System

AFAD is very useful for any rescue or hospital systems (like Rescue 1122, SAR, and NDRF etc.) because it can supply first aid much faster compared to typical supply through ambulances. In our system user gives us a location where we can deliver the first aid autonomously using a drone.

The Autonomous First Aid Drone (AFAD) can have significant importance in Pakistan, where natural disasters, such as floods and earthquakes, are a common occurrence. The AFAD can provide rapid response and quick access to life-saving equipment and medical supplies in areas where traditional methods of transport are inaccessible due to damage to roads or bridges.

In addition, Pakistan is a large country with many remote areas where access to medical aid is limited. The AFAD can reach these areas and provide emergency medical aid, reducing the risk of fatalities and improving survival rates.

Furthermore, Pakistan faces security challenges in some regions, which can make it difficult for emergency responders to access those areas. The AFAD can operate in such regions, providing emergency medical aid without putting human responders at risk. The cost-effective and accessible nature of the AFAD system can also be of significant importance in Pakistan, where many people cannot afford expensive medical aid or do not have access to it due to geographical or economic constraints.

## **2 REQUIREMENT SPECIFICATIONS**

### **2.1 Product Scope**

AFAD can be used in various fields like in the Medical field to deliver blood, medicine and first aid kit in an emergency conditions. The drone can be used in agriculture, industrial, photography area, military purpose. The metal detector system can be installing on a drone to find location of mines. The drone can be used in sprayer. There is no live tracking for AFAD after delivery of first aid it will autonomously return to the location from where it was launched.

### **2.2 Product Description**

#### **2.2.1 Product Perspective**

There are many drones available on market and also used in the different business organizations, but they are lacking in one feature or another. Medical field will get benefit from using our autonomous drone, enhance their performance and response time in case of emergency. The transportation of First Aid very difficult for a man to hard reach areas and the areas where peoples live without mobility. AFAD creates an honest medical culture and friendly environment with the user by providing first aid faster than road ambulance service.

The AFAD system's overall perspective can be illustrated through a diagram that shows the major components and their interactions with the environment. The diagram would show the drone flying autonomously to the user's location, guided by the mission planner, and delivering the first aid kit. The mobile application is used by the user to request the delivery, and the system transmits real-time health parameters of the patient to the ambulance where they can be analyzed in the lab view software. The system operates in emergency situations, where a fast response is required, and the traditional methods of transport may not be accessible.

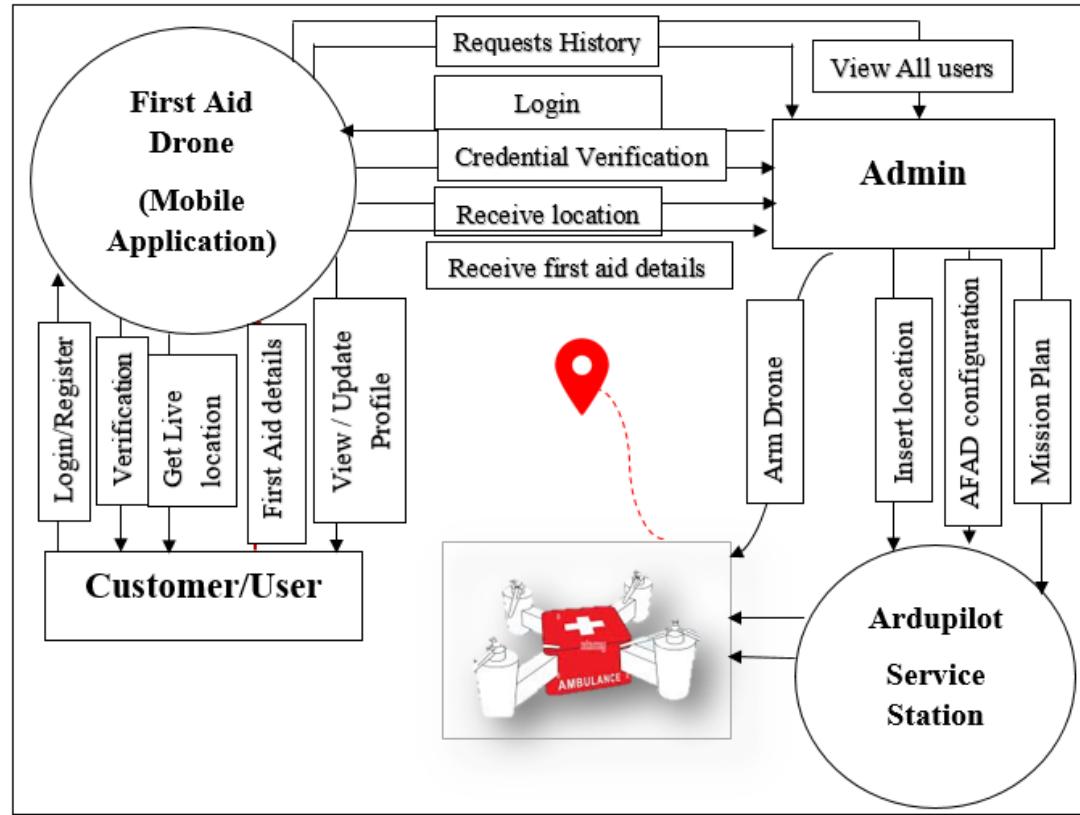


Figure 2.1 Product Perspective

### 2.2.2 Product Functionality

The major functionalities that this product has are as follows:

- Application shall allow end-users to login into the system through mobile phone number.
- Application shall provide verification through OTP on phone number.
- Application allows users to view and update their profile.
- Application shall allow user to get live location and location in latitude and longitude.
- Application shall allow user to add required first aid supplies details according to emergency type and severity.
- Application shall allow user to send the location and first aid details to the admin side.
- User will be able to provide feedback.
- Admin will receive the given location and add it to the drone and place the first aid kit.
- Admin can accept or reject the first aid request on basis of availability.
- Application shall notify user in case of acceptance or rejection of request

- Admin shall able to view all first aid requests.
- Admin shall able to view all users details and history.
- Admin will configure the drone, add location through service station and arm drone.
- Drone will fly autonomously to the location and return to location from where it was launched through RTL (Return to Launch).

### **2.2.3 Users and Characteristics**

There are mainly two users for this system Admin and user.

#### **2.2.3.1 Admin**

- Admin will manage the complete system.
- Admin will check first aid requirements and location details.
- Admin will receive the given location, plan the mission and place the first aid kit.
- Admin will view all requests, request status and user details.
- Admin will get notification of each request.
- Admin will able to accept or reject requests.
- Admin will configure drone for autonomous flight.
- Admin will plan mission by writing waypoints (Location: altitude and longitude), delays, flight modes into flight controller by using Mission Planner.
- Admin will place first aid kit with drone, arm drone and change drone modes if required to accomplish mission.

#### **2.2.3.2 User**

- User gives the location coordinates in both text and longitude-latitude to get their first aid kit.
- User gives first aid details and send it to admin side.
- User can view profile and can update it.
- User can provide feedback.
- User will be notified if the first aid request is accepted or rejected.

### **2.2.4 Operating Environment**

#### **2.2.4.1 Software Tools**

- Android Studio
- ArduPilot Mission Planner
- Draw.io
- StarUML
- Database: Firebase

#### ***2.2.4.2 Programming Language***

- Flutter Dart Language.

#### ***2.2.4.3 Operating Systems***

- Window 8 or 10
- Android phones

#### ***2.2.4.4 Hardware Specifications***

- Processor: Intel Core i5 6th Generation.
- RAM: 8 GB (ddr3 or ddr4)
- Hard Disk: 256 GB (SSD)
- Drone Kit
- APM 2.8 Flight Controller.
- GPS m8n

### **2.3 Specific Requirements**

#### **2.3.1 Functional Requirements**

The requirements of all the software system AFAD basic functions are listed here.

##### ***2.3.1.1 User Registration***

###### **Introduction**

To use our system user must have registered.

###### **Inputs**

Users have to sign-up using a phone number.

###### **Process**

A OTP will be sent to a given number and will be fetched automatically. After verification, their account will be created, and the user will be redirected to the login page where the user can log in by entering a login form.

###### **Outputs**

User's account will be created and he can perform different functionalities.

###### **Logout**

Users can log out of the system.

#### **2.3.1.2 Add Location and First Aid Details**

**Introduction:** To get required Firs Aid Supplies user must enter his exact location using google map API and also required to enter emergency type and severity.

**Inputs:** Users have to enter location in latitude and longitude and in text form. User have to select emergency type from dropdown, and enter required supplies in text field.

**Process:** When user will click location icon application will get location through API, it will take 2 seconds then user can enter other details.

**Outputs:** After clicking send a message will be shown “Ordered delivered, please wait for approval”. And the request will be sent to admin’s end.

#### **2.3.1.3 Feedback**

**Introduction:** User can provide feedback on performance of AFAD.

**Inputs:** Users can give 5-star rating. User can provide descriptive feedback.

**Process:** When user will submit the feedback it will send.

**Outputs:** After clicking send feedback will be sent to admin’s end.

#### **2.3.1.4 Update Profile**

**Introduction:** User can update his/her profile.

**Inputs:** Users can update username contact and other details.

**Process:** If user wants to update profile, simple he needs to open profile and update his information and click on save button.

**Outputs:** After clicking save information of particular user will be updated on frontend as well as backend.

#### **2.3.1.5    *Accept/Reject Request***

**Introduction:** Admin can view requests and can accept or reject any request on the basis of availability of first aid.

**Inputs:** Admin can simply click Accept or reject and status will be updated.

**Process:** When admin click any of two option Accept or reject.

**Outputs:** Status will be updated and particular user will be notified instantly.

#### **2.3.1.6    *Requests History***

**Introduction:** Admin can view all the requests from start till end.

**Inputs:** Admin will simply click the History icon on dashboard.

**Process:** All the requests will be fetched from databases saved as “cases”.

**Outputs:** Records will be shown to admin.

#### **2.3.1.7    *View Users***

**Introduction:** Admin can view all the registered users in application.

**Inputs:** Admin will simply click user’s icon on dashboard.

**Process:** All the users’ data will be fetched from firebase and shown to admin. By clicking message icon admin will be redirected to WhatsApp inbox of particular user.

**Outputs:** Records will be shown to admin.

### 2.3.2 Behavioral Requirements

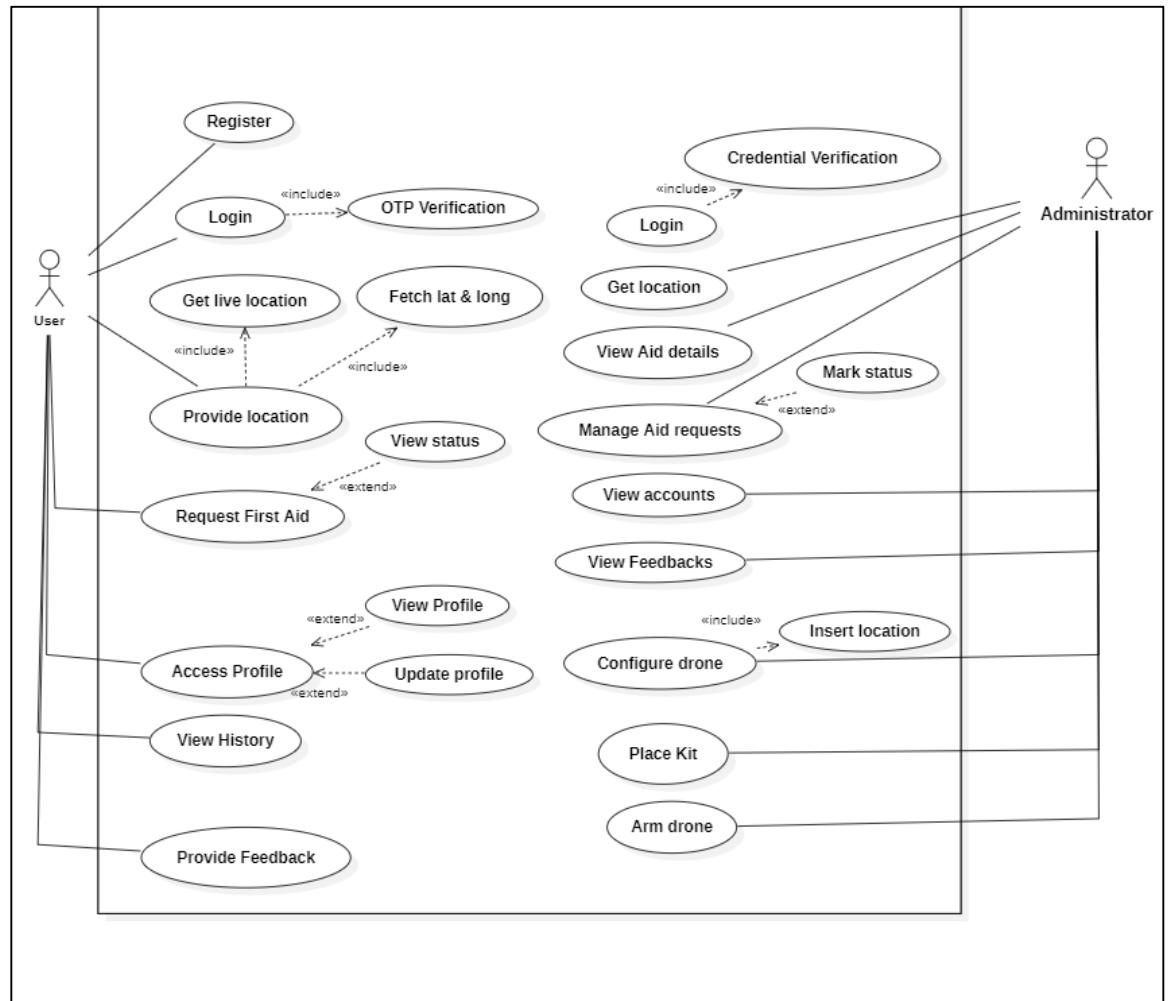


Figure 2.2 AFAD Use Case Diagram

### 2.3.2.1 Usecase#01 (Registration)

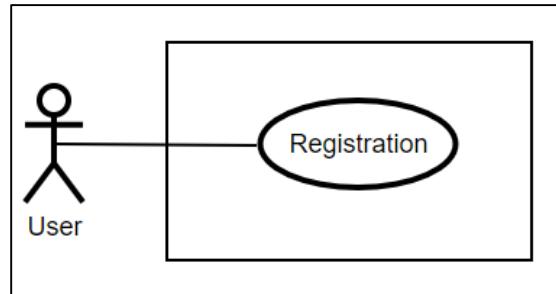


Figure 2.3 Registration Use Case

Table 2.1 Registration Use Case Description

<b>Actors</b>	User
<b>Description</b>	User can sign up using contact number.
<b>Pre-condition</b>	Pakistan number
<b>Steps</b>	User can register by entering number , email , username and profile picture.
<b>Post-condition</b>	Successful registration with the system.

### 2.3.2.2 Usecase#02 (Login)

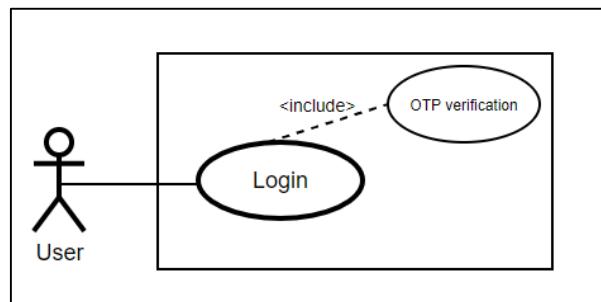


Figure 2.4 Login Use Case

Table 2.2 Login Use Case Description

<b>Actors</b>	User
<b>Description</b>	User can login by verifying number.
<b>Pre-condition</b>	One time password Verification.

<b>Steps</b>	User will enter number and get OTP on provided number and get verified.
<b>Post-condition</b>	Successfully Login with the system.

### 2.3.2.3 Usecase#03 (Provide Location)

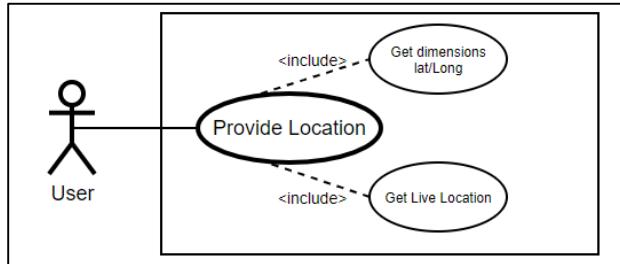


Figure 2.5 5Provide Location Use Case

Table 2.3 Provide Location Use Case Description

<b>Actors</b>	User
<b>Description</b>	User will provide location to admin.
<b>Pre-condition</b>	Google Map API should be working properly.
<b>Steps</b>	User will click location icon and wait for 2 sec. Application will get its current location and dimensions in longitude and latitude.
<b>Post-condition</b>	Successfully get location in both dimensions.

### 2.3.2.4 Usecase#04 (Request first Aid)

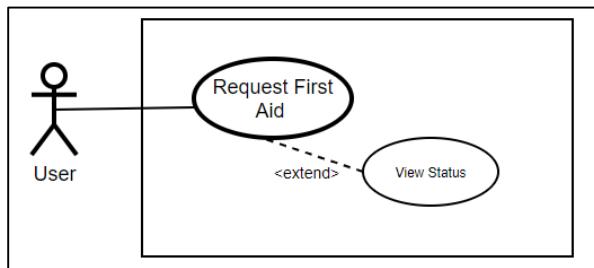


Figure 2.6 Request first Aid Use Case

Table 2.4 Request first Aid Use Case Description

<b>Actors</b>	User
<b>Description</b>	User can enter required first aid details.
<b>Pre-condition</b>	User must have some emergency.
<b>Steps</b>	User will select emergency type, severity, add description and send it to admin side.
<b>Post-condition</b>	First aid required case must be send on admin's side.

### 2.3.2.5 Usecase#05 (Access Profile)

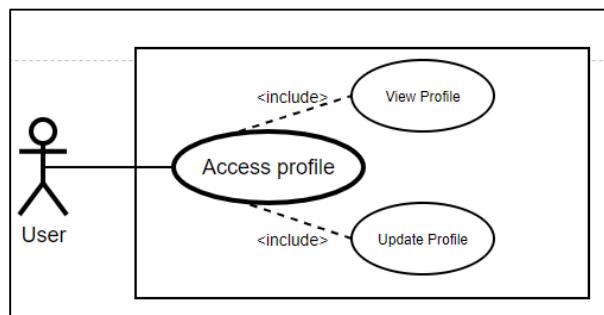


Figure 2.7 Access Profile Use Case

Table 2.5 Access Profile Use Case Description

<b>Actors</b>	User
<b>Description</b>	User can view and update profile.
<b>Pre-condition</b>	User must have an profile on application.
<b>Steps</b>	User will click on my account and update the information.
<b>Post-condition</b>	Successfully update on front-end as well as on firebase.

### 2.3.2.6 Usecase#06 (View History)

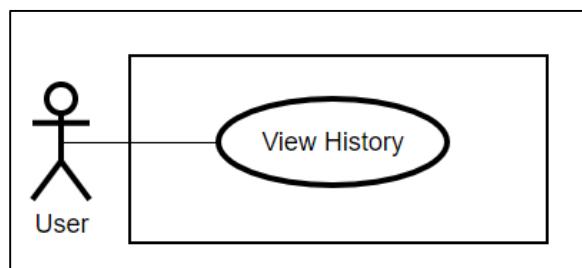
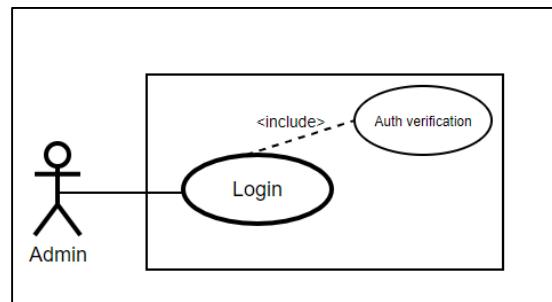


Figure 2.8 View History Use case

**Table 2.6 View History Use case Description**

<b>Actors</b>	User
<b>Description</b>	User can view his request histories.
<b>Pre-condition</b>	User have at least one request to view History.
<b>Steps</b>	User will click on Requests and can see his requests.
<b>Post-condition</b>	Successfully requests displayed.

#### 2.3.2.7 Usecase#07 (Login Admin)

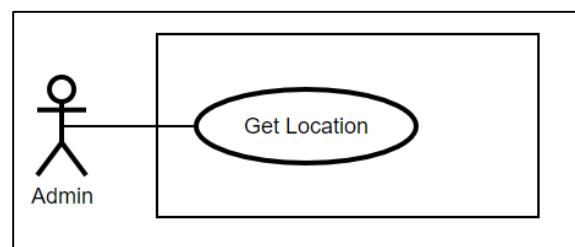


**Figure 2.9 Login Admin Use Case**

**Table 2.7 Login Admin Use Case Description**

<b>Actors</b>	Admin
<b>Description</b>	Admin can login to system.
<b>Pre-condition</b>	There should be admin login credentials in Auth.
<b>Steps</b>	Admin can login by entering provided email and password.
<b>Post-condition</b>	Successfully login to system.

#### 2.3.2.8 Usecase#08 (Get Location)

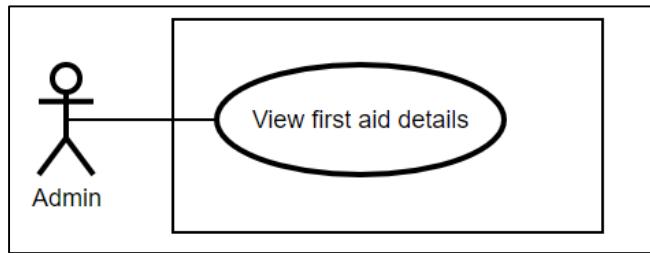


**Figure 2.10 Get Location Use Case**

**Table 2.8 Get Location Use Case Description**

<b>Actors</b>	Admin
<b>Description</b>	Admin can get location.
<b>Pre-condition</b>	User must have sent location towards admin side.
<b>Steps</b>	Admin can view location of user when a first aid request is made by user.
<b>Post-condition</b>	Successfully take location.

#### 2.3.2.9 Usecase#09 (View First Aid Details)



**Figure 2.11 View First Aid Details Use Case**

**Table 2.9 View First Aid Details Use Case Description**

<b>Actors</b>	Admin
<b>Description</b>	Admin can provide first aid according to details provided by user.
<b>Pre-condition</b>	User must have enter some details of accident.
<b>Steps</b>	By clicking on requests user can see first id requests with details i.e. Emergency type, Severity and some description.
<b>Post-condition</b>	Details viewed successfully.

### 2.3.2.10 Usecase#10 (Manage Aid Requests)

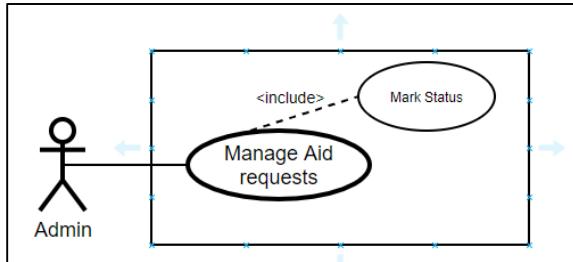


Figure 2.12 Manage Aid Requests Use Case

Table 2.10 Manage Aid Requests Use Case Description

<b>Actors</b>	Admin
<b>Description</b>	Admin can manage each request by marking it as Accepted or Discarded.
<b>Pre-condition</b>	Availability of first aid.
<b>Steps</b>	When admin have a request he will click accept or discard button.
<b>Post-condition</b>	Successfully accepted or rejected.

### 2.3.2.11 Usecase#11 (Configure Drone)

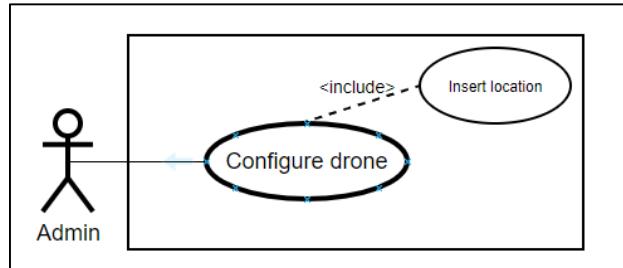


Figure 2.13 Configure Drone Use case

Table 2.11 Configure Drone Use case Description

<b>Actors</b>	Admin
<b>Description</b>	Admin can configure drone by using Ardupilot mission planner.
<b>Pre-condition</b>	All hardware components should be integrated.
<b>Steps</b>	Admin will attach drone with mission planner through USB, configure it, Add location in latitude and longitude.
<b>Post-condition</b>	Drone configured successfully.

### 2.3.2.12 Usecase#12 (Arm Drone)

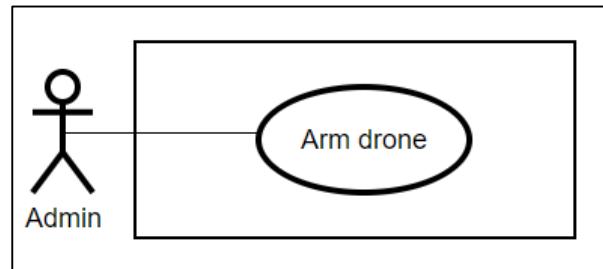


Figure 2.14 Arm Drone Use Case

Table 2.12 Arm Drone Use Case Description

<b>Actors</b>	Admin
<b>Description</b>	Admin will place First Aid Kit and Arm the drone through transmitter.
<b>Pre-condition</b>	Transmitter and receiver should be synchronized.
<b>Steps</b>	Admin will place kit , Place drone on a location , Turn transmitter on and give throttle to arm it.
<b>Post-condition</b>	Drone get armed and fly to the location inserted in location.

### 2.3.3 External Interface Requirements

#### 2.3.3.1 User Interface



Figure 2.15 Splash Screen

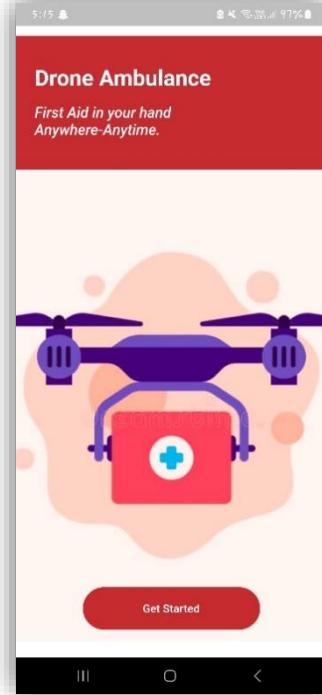


Figure 2.16 Get Started Screen

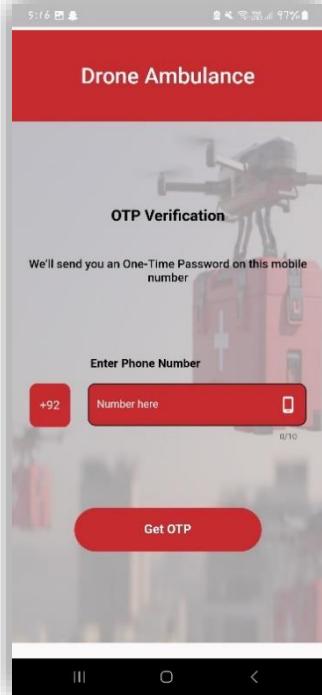


Figure 2.17 Enter Number Screen



Figure 2.18 OPT verification

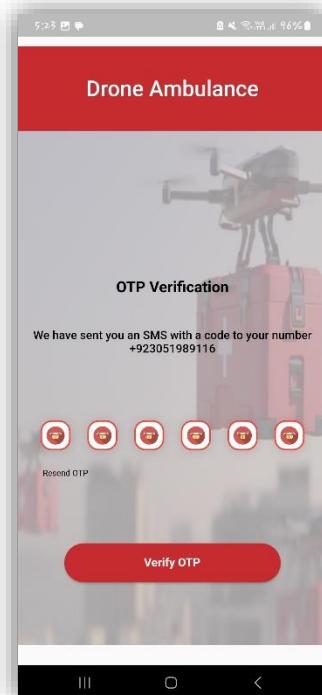


Figure 2.19 Enter OTP

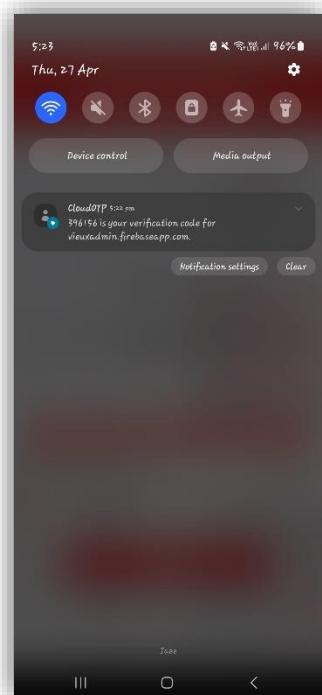
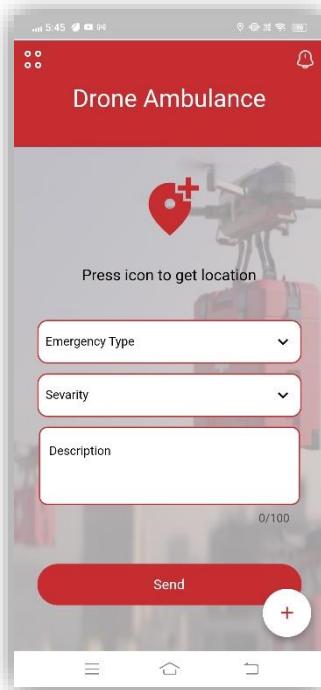


Figure 2.20 OTP notification



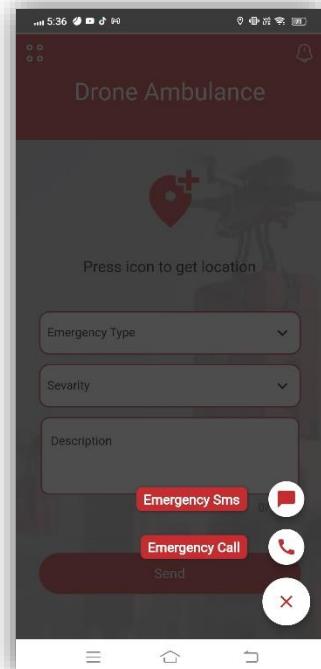
**Figure 2.23 User Registration Form**



**Figure 2.22 User Registration**



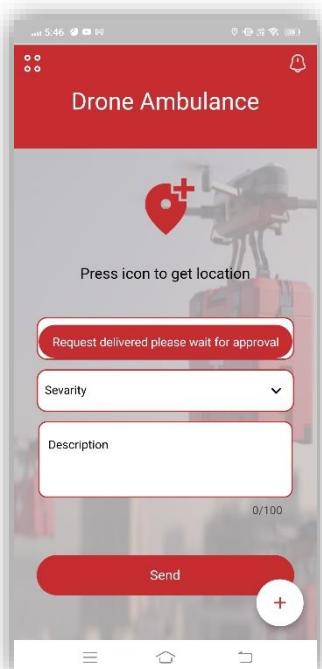
**Figure 2.21 Home Screen**



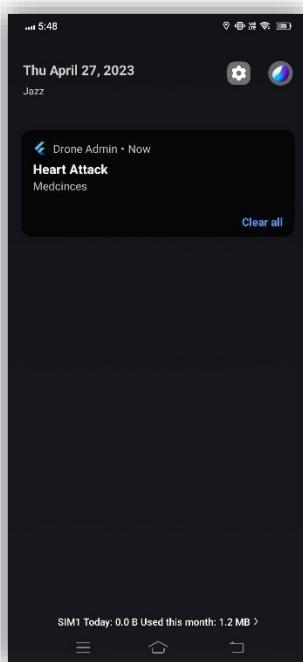
**Figure 2.26 Emergency call and message**



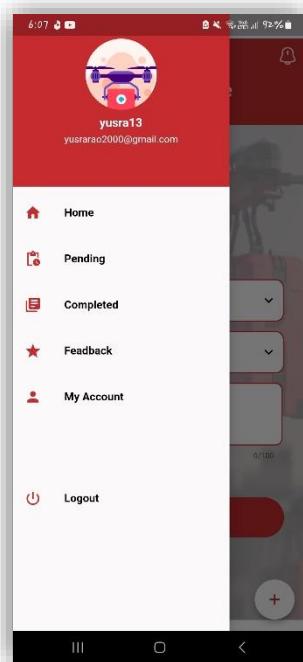
**Figure 2.25 Emergency details and Location**



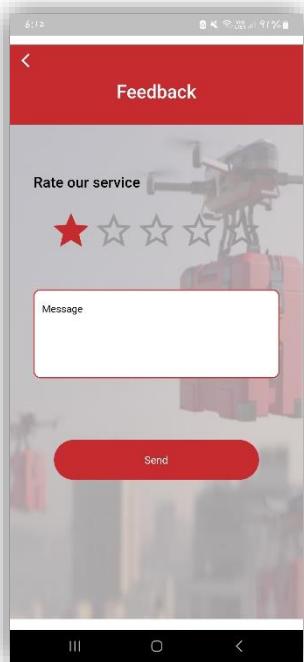
**Figure 2.24 Request sent**



**Figure 2.29 Aid Request Notification**



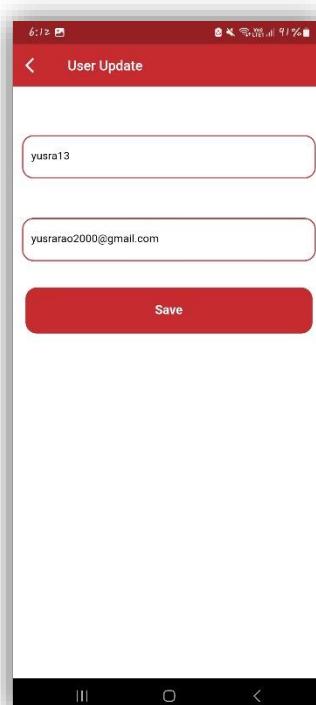
**Figure 2.28 Drawer**



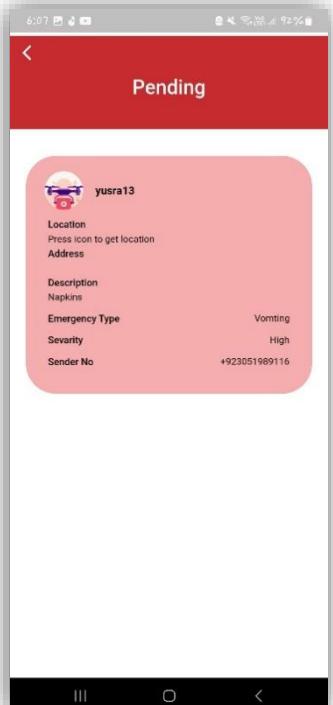
**Figure 2.27 Feedback screen**



**Figure 2.32 User Profile Screen**



**Figure 2.31 User Update Screen**



**Figure 2.30 Pending Request**

### 2.3.3.2 Admin Interface

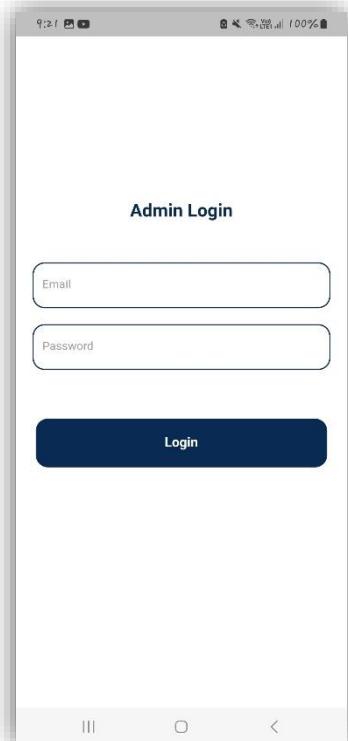


Figure 2.33 Admin Login

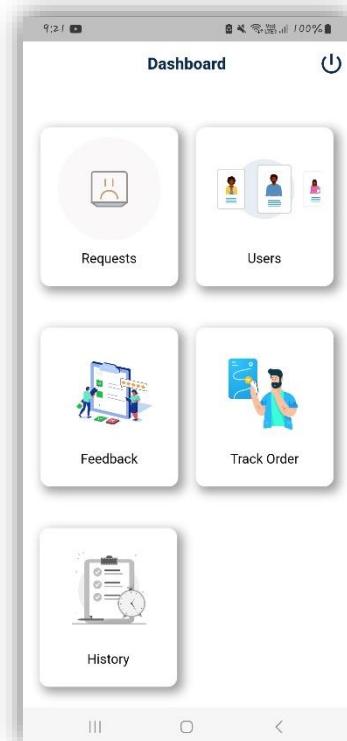


Figure 2.35 Admin Dashboard

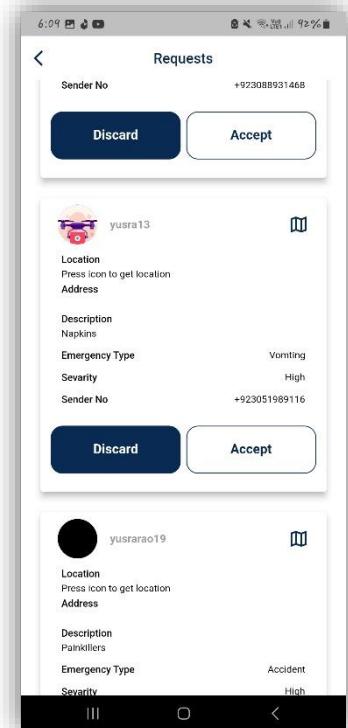


Figure 2.34 All requests

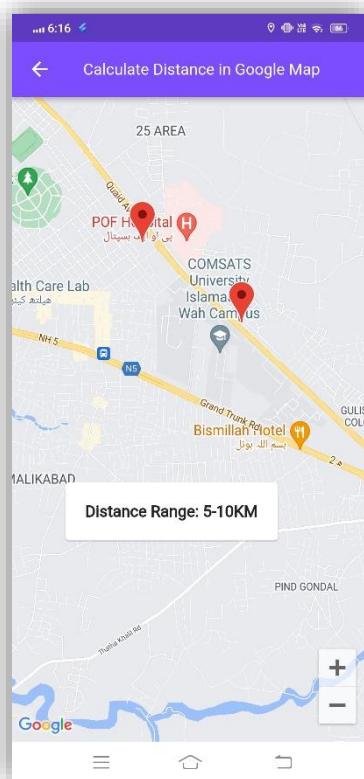


Figure 2.37 Calculate Distance

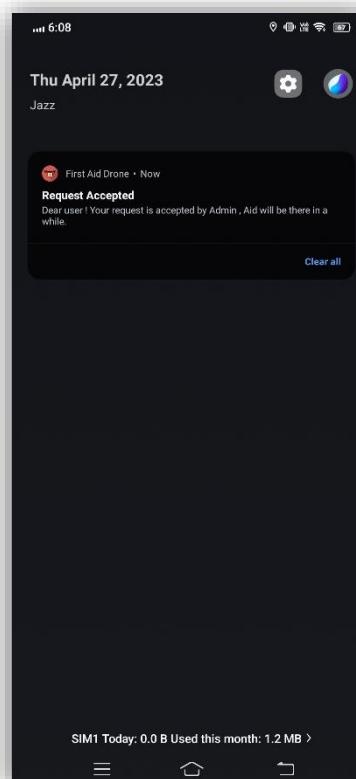


Figure 2.38 Request accepted notification

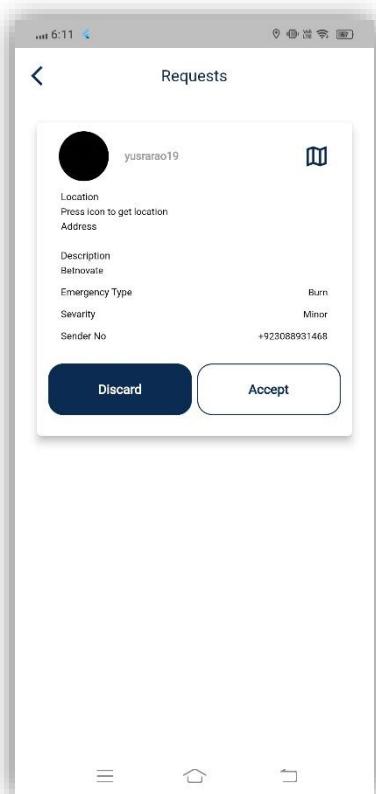
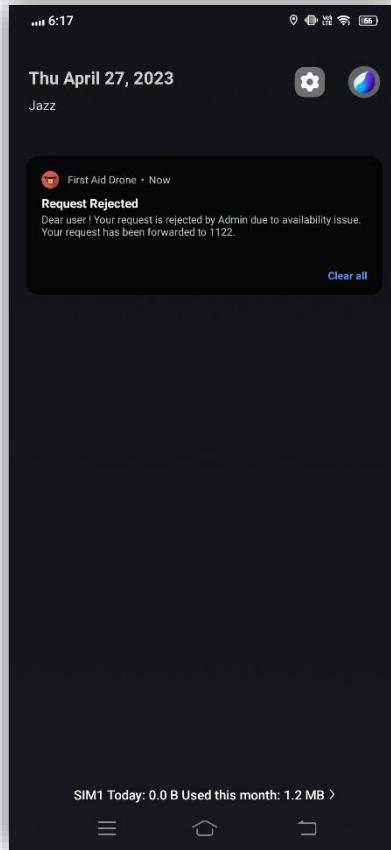
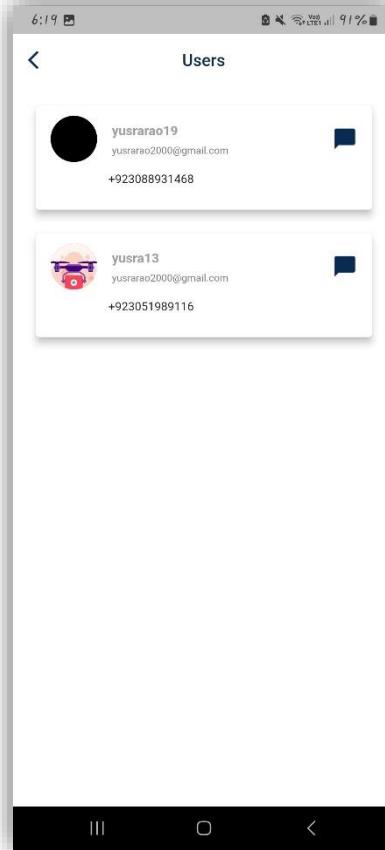


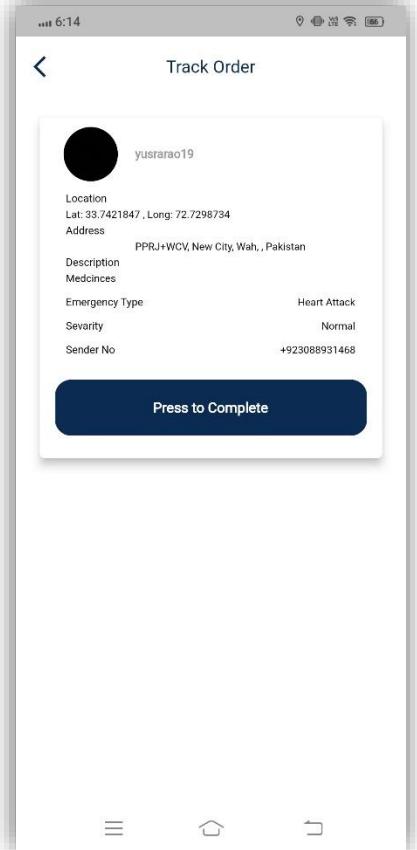
Figure 2.36 New Request



**Figure 2.41 Request Rejected Notification**

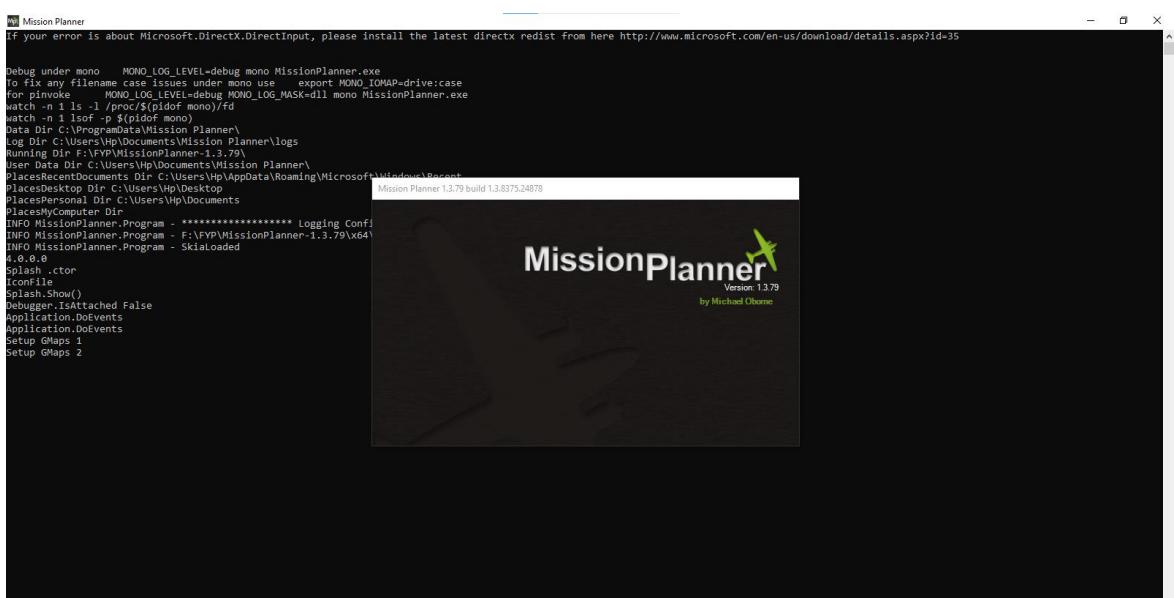


**Figure 2.40 View All users**



**Figure 2.39 Track Order**

### 2.3.3.3 Mission Planner (Server Interface)



**Figure 2.42 Mission Planner Screen**

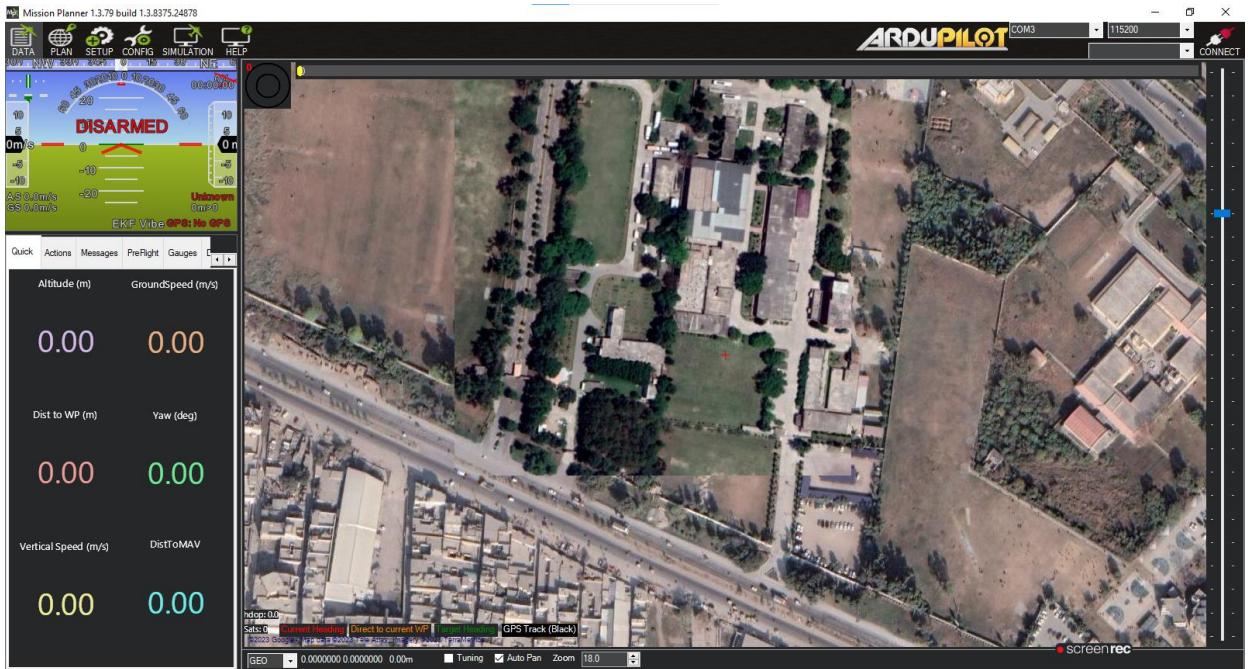


Figure 2.43 Start Screen



Figure 2.44 Plan Mission Interface

#### 2.3.3.4 Drone Interface

**Drone Frame:** - A drone frame has two main parts: the body and the arms. The body houses and protects your electronic components including flight controller, 4 ESC, BLDC motors.



Figure 2.45 Drone Frame

**BLDC motor:** - It uses an electronic controller to switch DC currents to the motor windings producing magnetic fields which effectively rotate in space and which the permanent magnet rotor follows.



Figure 2.46 Brushless DC motor

**Electric Speed Controller:** - The term ESC stands for “electronic speed control is an electronic circuit used to change the speed of an electric motor, its route, and also to perform as a dynamic brake.



Figure 2.47 Electric Speed Controller

**Flight Controller:** - This autopilot flight controller can control fixed-wing aircraft, multi-rotor helicopters, as well as traditional helicopters.



Figure 2.48 APM flight Controller

**GPS:** - GPS drones are equipped with a GPS module that allows them to know their location relative to a network of orbiting satellites. GPS helps in perform functions such as position hold, autonomous flight, return to home, and waypoint navigation.



Figure 2.49 m8n GPS

**Li-Po Battery:** -The most common batteries used in drones are lithium polymer (Lipo) batteries.



Figure 2.50 Lipo battery

**Propellers:** - Drone propellers provide lift for the aircraft by spinning and creating an airflow.



Figure 2.51 Propellers

### Circuit Diagram: -

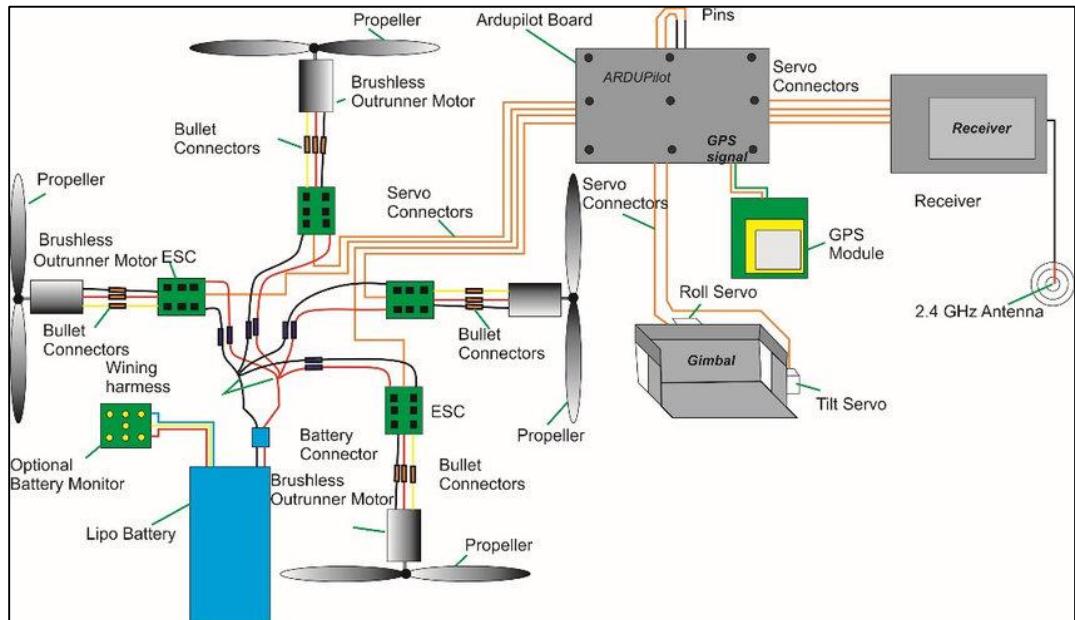


Figure 2.52 AFAD circuit Diagram

### Direction Diagram: -

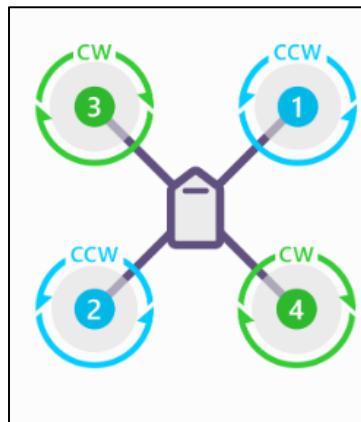


Figure 2.53 Quad Directions

### Software Interfaces: -

1. Flutter for Mobile Application
2. Ardupilot Mission Planner
3. Notification pushing API
4. Google map API.
5. Firebase will be used as a Database

**Communications Interfaces:** - The GPS module and Wi-Fi Module will use as communication interface; the Wi-Fi module provide network to the GPS to work at the live location. The microcontroller will guide the drone to fly according the instruction.

## 2.4 Non-functional Requirements

### 2.4.1 Performance Requirements

- Mobile Application should work properly if there is stable internet connection.
- Application should be fast and responsive and page loading speed must be < 5 seconds.
- Application should take < 2 seconds to get location.
- To track drone GPS module is installed but to track in a long range telemetry will be required.
- We are just developing the basic prototype of the AFAD which will work with in a Campus. It should cover area of approximately 1km.
- AFAD should make a flight autonomously without being controlled remotely.
- The drone should be able to fly at a minimum speed of 20 m/s to ensure fast delivery of the first aid kit.
- The drone should be able to carry a payload of up to 1.5 kg, including the first aid kit and any other necessary equipment.

### 2.4.2 Safety and Security Requirements

**Flight Safety:** The drone should be designed to ensure safe and stable flight operations. This includes ensuring that the drone can avoid obstacles and other aircraft and maintain safe altitude and speed.

**Payload Safety:** The first aid kit carried by the drone should be securely attached and protected to prevent any harm or damage during transport. It is important to ensure that the payload does not fall off or cause any damage to the drone or people on the ground.

**Privacy:** The AFAD should protect the privacy of individuals receiving medical aid. This includes ensuring that any personal or medical data collected by the drone is protected and not shared with unauthorized parties.

**Airspace Regulations:** The AFAD must comply with all applicable regulations governing the use of airspace for drones. This includes obtaining necessary permits and licenses, complying with flight restrictions and avoiding restricted areas.

**Emergency Stop:** The AFAD should have an emergency stop function that can be activated in case of any technical or operational issues. This should be easy to access and operate, and should result in the drone landing safely and without causing any harm or damage.

#### **2.4.3 Software Quality Attributes**

**Security:** AFAD provide secure delivery of First Aid as well as application provides security as well through OTP verification.

**Portability:** This Application will work on any Platform. i.e., Android/IOS.

**Reusability:** AFAD system is divided in such a way that if one module is needed for another system, it can be easily reusable.

i.e. Hardware modules GPS, APM can be used for any quadcopter.  
Application can be used for any delivery system with little amendments.

**Extensible:** AFAD system can be extended by using telemetry module drone will cover large range. There will be no effect on working.

**Reliability:** This system will fulfill all the needs of the project in the form of quality.

**Maintainability:** The system will be designed using professional standards and will be properly documented.

# 3 DESIGN SPECIFICATIONS

## 3.1 Introduction

This chapter includes composite structure diagram that depict the components, their relationships, and how they interact to form the system. It shows the entire system's physical structure, including subsystems, components, and their interconnections. The logical viewpoint defines the system's behavior and functionality. The UML class diagram is used to represent the logical view of AFAD. It shows the classes, their attributes, and their relationships. In interaction viewpoint sequence diagram will show the message exchange between the users. The algorithmic viewpoint defines the algorithms that the system uses to process data and respond to events.

## 3.2 Composite Viewpoint

### 3.2.1 Composite Diagram

Package diagrams are used, in part, to depict import and access dependencies between packages, classes, components, and other named elements within your system.

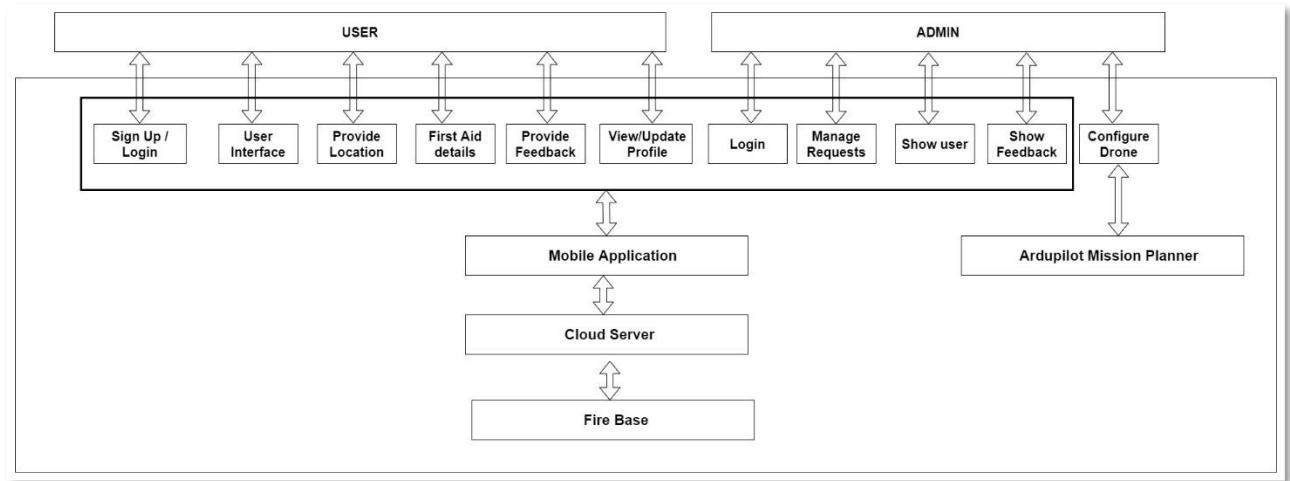


Figure 3.1 Composite Diagram

### 3.2.2 Package Diagram

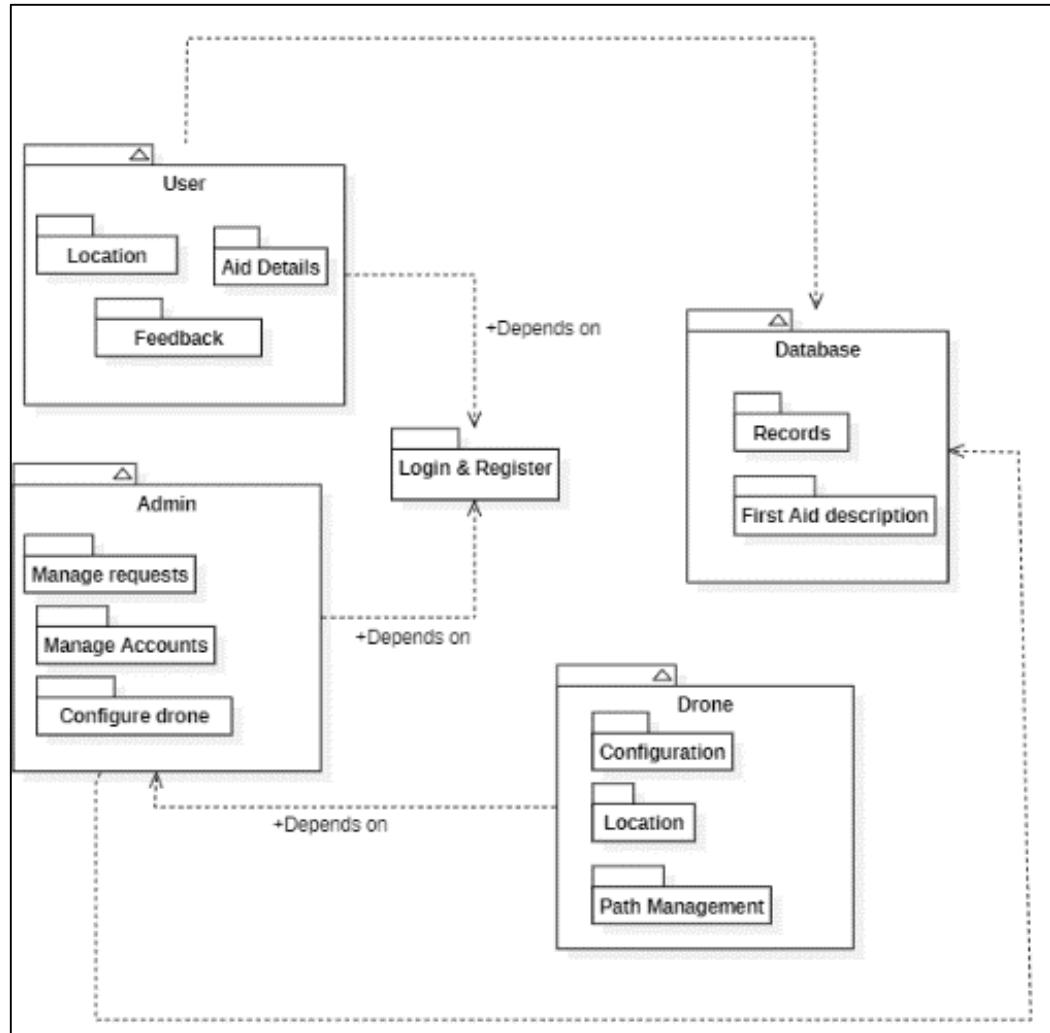


Figure 3.2 Package Diagram

### 3.3 Logical Viewpoint

A class diagram in UML is a type of static structure diagram that describes the structure of a system by showing the system's classes, their attributes, operations, and relationships among the objects. The class diagram is the basic building of object-oriented modeling.

#### 3.3.1 Class Diagram

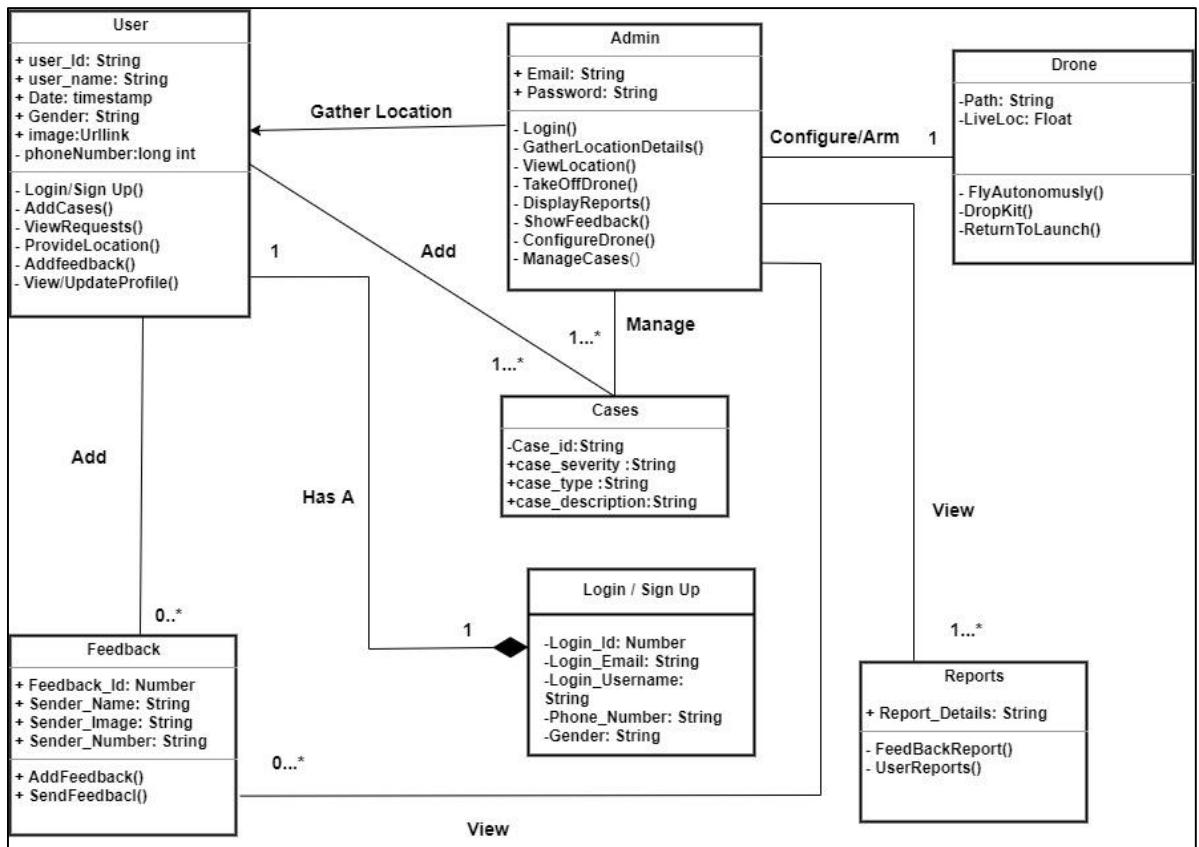


Figure 3.3 AFAD Class Diagram

## 3.4 Information Viewpoint

### 3.4.1 Entity Relation Diagram

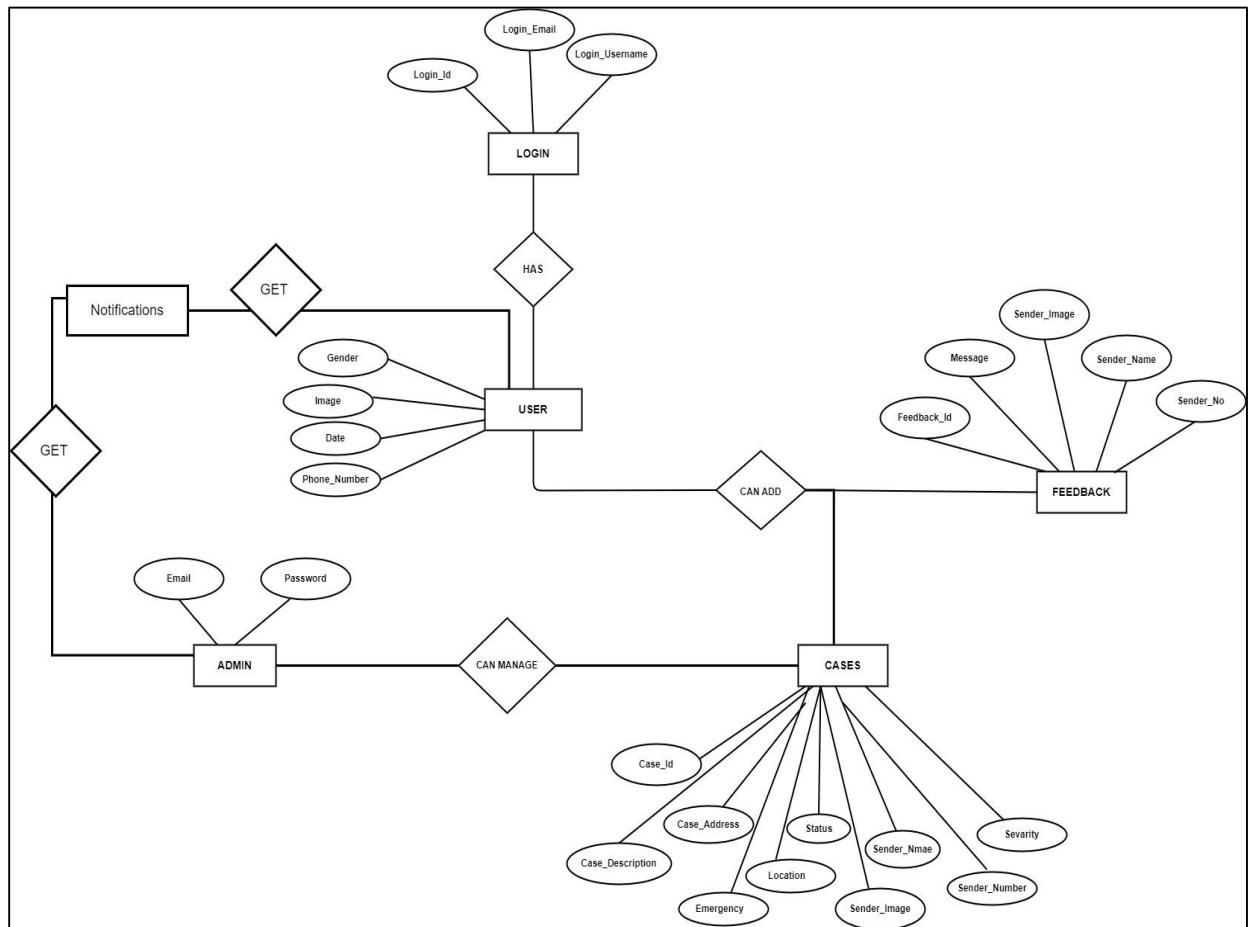


Figure 3.4 Entity Relation Diagram

## 3.5 Interaction Viewpoint

A sequence diagram simply depicts the interaction between objects in sequential order i.e., the order in which these interactions take place.

### 3.5.1 Admin Sequence Diagram: -

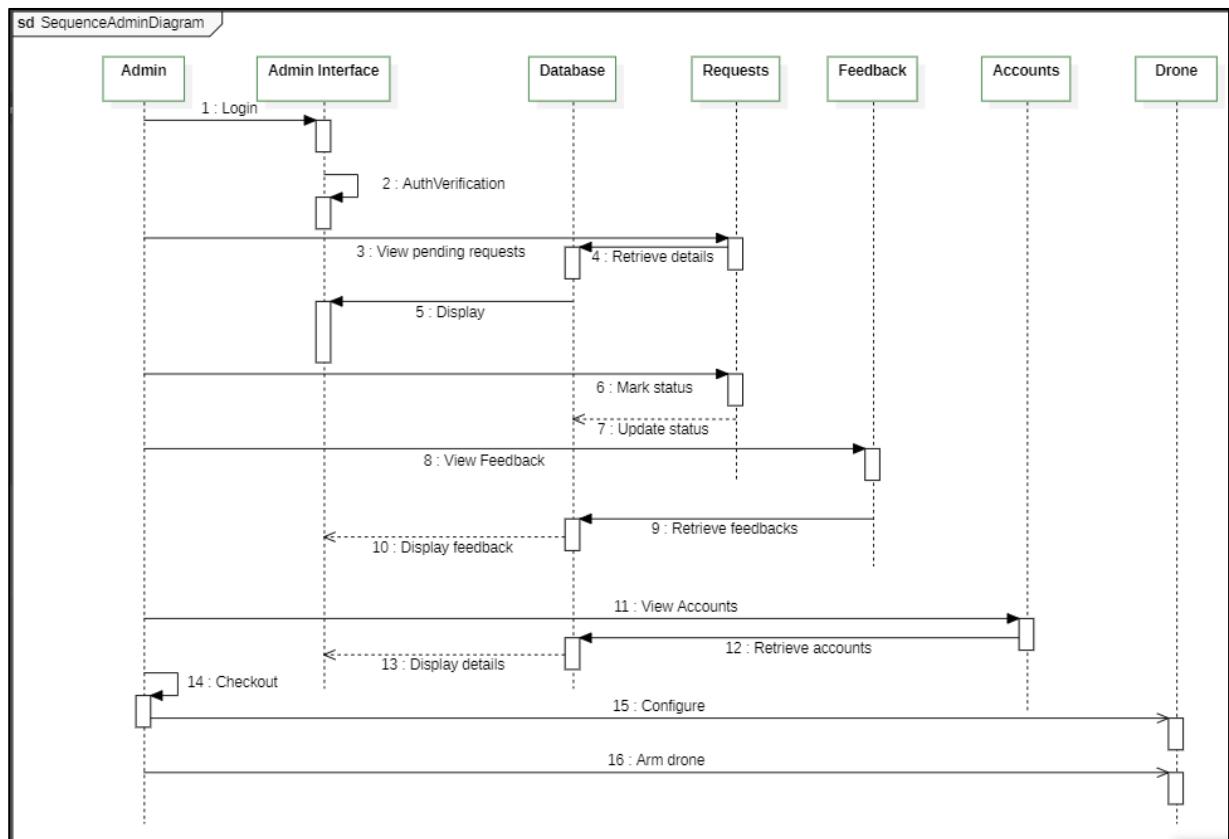


Figure 3.5 Sequence Diagram (Admin)

### 3.5.2 User Sequence Diagram

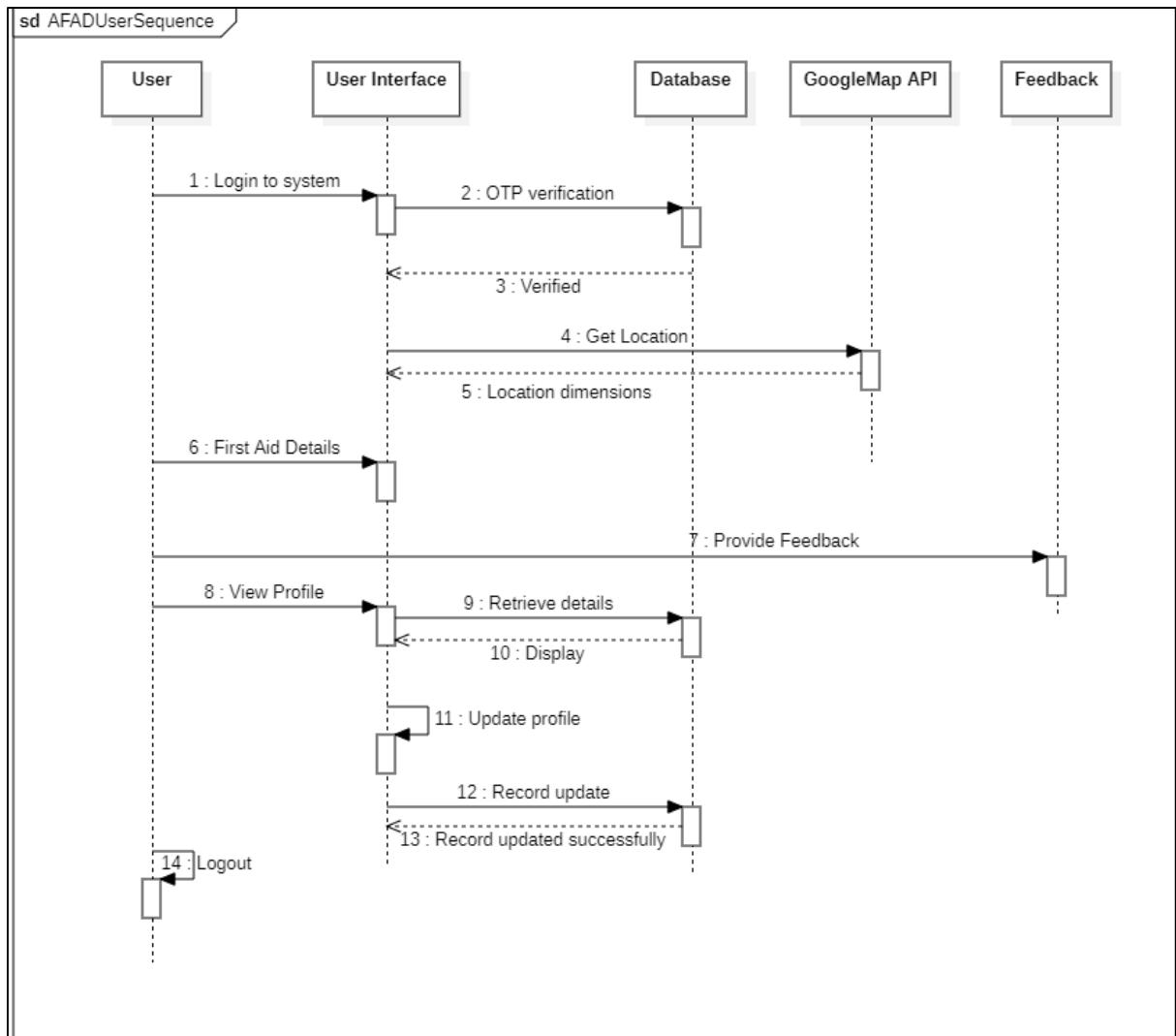


Figure 3.6 Sequence Diagram (User)

### 3.6 State Dynamics Viewpoint

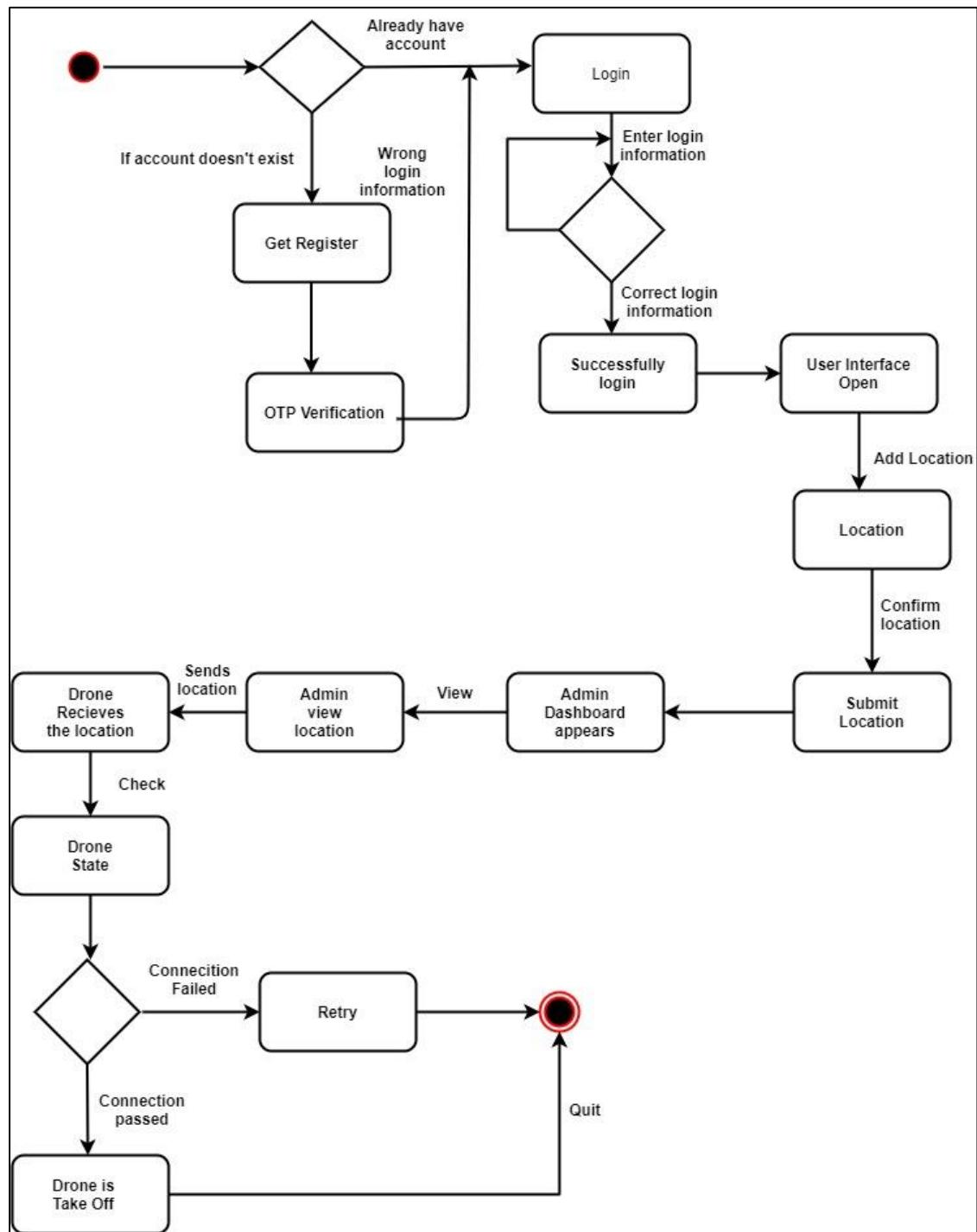


Figure 3.7 AFAD State Dynamic Diagram

## 3.7 Algorithmic Viewpoint

Pseudo code is used for creating an outline or a rough draft of a program. Pseudo code summarizes a program's flow but excludes underlying details.

### 3.7.1 Algorithmic Viewpoint of User

**Table 3.1 User class , Attributes , Login method**

<b>Class Name:</b> User	
<b>Brief Description:</b> This class describes the responsibility and functionalities of User. User can log in to perform further action by using this class.	
Attributes (fields)	Attributes Description
ID	ID is used to identify this entity uniquely.
Username	Username is used to store the name of the user.
Contact number	Phone number will be stored.
Email	Gmail will be saved in email.
Profile	Image URL will be saved in profile.
Methods (operation)	Methods Description
<b>Login ()</b>	A method used to perform log in when user wants to login the system.
	<b>Pseudo Code</b> <ol style="list-style-type: none"><li>1. Get the phone number from the user</li><li>2. Verify phone number by sending OTP. If (EnteredOTP == CloudOTP)</li><li>3. Display the registration form.</li><li>4. If (form==filled)</li><li>5. Display the user interface</li><li>6. Else</li><li>7. Resend OTP</li></ol>

**Table 3.2 ProvideLocation Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to provide location.
	<b>Pseudo Code</b>
<b>ProvideLocation ()</b>	<ol style="list-style-type: none"><li>1. Click the location icon.</li><li>2. Wait for 2 seconds.</li><li>3. Application will get location in two dimensions' longitude, latitude in current location in text.</li></ol>

**Table 3.3 AddCase Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to provide location.
	<b>Pseudo Code</b>
<b>AddCase ()</b>	<ol style="list-style-type: none"><li>1. Select emergency type.</li><li>2. Select severity of emergency case.</li><li>3. Enter requirement description.</li><li>4. Click send button to send it to admin side.</li></ol>

**Table 3.4 ProvideFeedback Pseudo code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to provide feedback to service.
	<b>Pseudo Code</b>
<b>ProvideFeedback ()</b>	<ol style="list-style-type: none"><li>1. Select feedback option.</li><li>2. Rate the AFAD service using 5-star rating.</li><li>3. Give a comment description.</li><li>4. Click send button to send it to admin side.</li></ol>

**Table 3.5 ViewRequest Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
<b>ViewRequest ()</b>	A method used view requests made by user and status whether it is accepted , rejected or completed.
	<p style="text-align: center;"><b>Pseudo Code</b></p> <ol style="list-style-type: none"> <li>1. Select request option.</li> <li>2. If (User have pending request)</li> <li>3. Display details.</li> <li>4. Else</li> <li>5. No record to show</li> </ol>

**Table 3.6 UpdateProfile Pseudo code**

<b>Method</b> (operation)	<b>Method Description</b>
<b>UpdateProfile ()</b>	A method used allow user to modify details in firebase.
	<p style="text-align: center;"><b>Pseudo Code</b></p> <ol style="list-style-type: none"> <li>1. Select profile.</li> <li>2. Update email, username.</li> <li>3. Save the information.</li> </ol>

### 3.7.2 Algorithmic Viewpoint of Admin

**Table 3.7 User Admin , Attributes , Login method**

<b>Class Name:</b> Admin	
<b>Brief Description:</b> This class describes the responsibility and functionalities of Admin. User can log in to perform further action by using this class.	
Attributes (fields)	Attributes Description
Email	Gmail will be saved in email.
Password	Password will be stored in password.
Methods (operation)	Methods Description
	A method used to perform log in when admin wants to login the system.
Pseudo Code	
Login ()	<ol style="list-style-type: none"><li>1. Get the Email and password from the admin.</li><li>2. Verify these credentials from the auth file.</li><li>3. If one of these fields is incorrect “Invalid Information”</li><li>4. Else “Successfully Login”</li><li>5. Display admin dashboard.</li></ol>

**Table 3.8 GetLocation Pseudo code**

<b>Method</b> (operation)	<b>Method Description</b>
<b>Method</b> (operation)	A method used to get location. A method used to accept or reject any request
	<b>Pseudo Code</b>
<b>GetLocation ()</b> <b>ManageRequests ()</b>	<p>1. Click on requests and view location of user.</p> <p>1. Click on requests.</p> <p>2. If (First aid == available)</p> <p>3. Accept request</p> <p>4. Else</p> <p>5. Discard request.</p>

**Table 3.9 ManageRequests Pseudo Code**

**Table 3.10 ShowFeedback Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to display feedback given by users to AFAD service.
<b>ShowFeedback ()</b>	<b>Pseudo Code</b> <p>1. Select feedback option.</p> <p>2. If (feedback count&gt;0)</p> <p>3. Show feedbacks.</p> <p>4. Else</p> <p>5. No record available.</p>

**Table 3.11 ShowUsers Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to display users record that are registered to AFAD system.
<b>ShowUsers ()</b>	<p style="text-align: center;"><b>Pseudo Code</b></p> <ol style="list-style-type: none"> <li>1. Select users option.</li> <li>2. If (users count&gt;0)</li> <li>3. Show users.</li> <li>4. Else</li> <li>5. No record available</li> </ol>

**Table 3.12 ConfigureDrone Pseudo Code**

<b>Method</b> (operation)	<b>Method Description</b>
	A method used to configure drone so that it can fly to particular location.
<b>ConfigureDrone ()</b>	<p style="text-align: center;"><b>Pseudo Code</b></p> <ol style="list-style-type: none"> <li>1. Connect drone flight controller with Ardupilot mission planner.</li> <li>2. Insert location in latitude and longitude.</li> <li>3. Create waypoints.</li> <li>4. Path management.</li> <li>5. Active Return to Launch mechanism.</li> </ol>

## 4 DEVELOPMENT AND TOOLS

This chapter is the detailed account on the tools used in the development of the project with the development plan. Development plan includes all the phases that

were involved in building AFAD system from the scratch. Different languages and frameworks were used in the project. At the end, conclusion and future work assure the placement of this project in upcoming times. Highest motivation of planning is to get work done to meet academic deadlines and to keep our work aligned. Moreover, there are several tools and ways to perform a task. This section best describes choices made by our own team.

## 4.1 Introduction

Our project is collaboration of Mobile application, Mission Planner and hardware (Drone) that includes various modules to perform the tasks like add Location of User, First Aid Request, Aid Delivery. Our system gets the location from the user in Latitude and Longitude and save location coordinate in database and our admin will get those coordinate and Aid details of user then place those coordinates in Mission planner after that kit will be attached with drone will fly to intended location autonomously.

Tools used to develop AFAD system are: -

- Android Studio
- Firebase
- Mission Planner
- APM Flight Controller
- GPS

For development of AFAD we use devices: -

- Laptop (HP core i5 ProBook).
- Laptop (HP core i7 10<sup>th</sup> generation)

## 4.2 Development Plan

This project is developed by a team of two members.

1. YUSRA TAHIR
2. SHAHZEB NOOR

### 4.2.1 Project Modules Distribution

**Table 4.1 FYP work distribution**

Project Members	Names & Enrollments	Member wise Module Information
1	YUSRA TAHIR	<ul style="list-style-type: none"> <li><b>1. Three chapters of SRDS/Report</b></li> <li><b>2. Mobile Application (Admin Module)</b> <ul style="list-style-type: none"> <li>• Request Accept/Reject</li> <li>• Integration of notification pushing API.</li> <li>• View History</li> <li>• View Users</li> <li>• Map integration</li> <li>• Firebase (Backend)</li> </ul> </li> <li><b>3. Hardware (Drone)</b> <ul style="list-style-type: none"> <li>• Flight controller integration &amp; configuration (Autonomous flight)</li> <li>• GPS integration</li> <li>• Return to launch (RTL)</li> <li>• ESC calibration</li> <li>• Power Module installation</li> </ul> </li> </ul>
t	SHAHZEB NOOR	<ul style="list-style-type: none"> <li><b>1. Three chapters of SRDS/Report</b></li> <li><b>2. Mobile Application (User Module)</b> <ul style="list-style-type: none"> <li>• Login/signup</li> <li>• Update User</li> <li>• Location API</li> <li>• Feedback</li> </ul> </li> <li><b>3. Hardware (Drone)</b> <ul style="list-style-type: none"> <li>• Receiver/Transmitter</li> <li>• BLDC motors / Propellers installation</li> <li>• Battery Connections.</li> </ul> </li> </ul>

The development plan of this project has been divided into the following parts:

**Table 4.2 Project Development Plan**

### 4.3 Development Tools

ID	Duration	Duration	Start	Finish	%Done
1	Project planning	15 days	Tue Aug 30, 2022	Wed Sep 14, 2022	100%
	Proposal formulation	15 days	Thu Sep 15, 2022	Thu Sep 29, 2022	100%
2	Review and Approve Proposal	7 days	Mon Oct 10, 2022	Fri Oct 21, 2022	100%
	Gather Requirements	2 weeks	Fri Oct 21 , 2022	Thu Nov 3 , 2023	100%
3	Figma Design UI/UX	2 days	Fri Nov 4, 2023	Sun Nov 6, 2023	100%
	UML Diagrams	5 days	Mon Nov 7,2022	Mon Nov 14, 2022	100%
4	Database Design	10 days	Tue Nov 15, 2022	Fri Nov 25, 2022	100%
	AFAD Application Frontend	10 days	Sat Nov 26, 2022	Mon Dec 5, 2022	100%
5	AFAD Application Backend	5 days	Tue Dec 6 , 2022	Sun Dec 11 , 2022	100%
	Firebase Connectivity	5 days	Mon Dec 12 , 2022	Fri Dec 16 , 2022	
6	SRDS/Demo Release 1 25% work	15 days	Sun Dec 18, 2022	Sun Jan 1, 2023	100%
	Review and Approve SRS	6 days	Wed Jan 4, 2023	Tue Jan 10 , 2023	100%
7	Drone parts installment , Configuration	60 days	Thu Jan 15 , 2023	Thu Mar 16, 2023	100%
	Testing including Test Plans and Test Cases	30 days	Mon Mar 20 , 2023	Tue Apr 18 , 2023	100%
8	Project Finalization Report of FYP	8 days	Thu Apr 20 , 2023	Fri Apr 28 , 2023	100%

The development tools used for this project are listed below: -

- Flutter: 3.0 or above
- Android studio 11.0.13

- Android Emulator
- Mission planner 1.3.79
- Draw.io
- Figma
- Capcut / InShot

#### **4.4 Conclusion and Future Work/Extensions**

In conclusion, the AFAD system designed through UML diagrams has the potential to improve the efficiency and effectiveness of emergency response efforts. The system includes various components, such as the admin interface, the user interface, and the communication interfaces. The system architecture and design can be further improved and refined based on the specific needs and requirements of the emergency response organization.

Future work and extensions for AFAD could include integrating additional functionalities, such as real-time tracking of emergency vehicles and responders, automated alerts and notifications to relevant parties, and machine learning algorithms for predictive analytics. The system could also be expanded to support international emergency response efforts, with multilingual interfaces and cross-border communication capabilities. Also with the use of telemetry range can be enhanced up to 20km with real-time tracking, and the system will provide real-time battery status updates to the admin to avoid mission failures due to low battery. Ultrasonic sensors can be installed in drone to prevent any crash.

Overall, AFAD has the potential to significantly enhance emergency response operations and improve outcomes for those affected by emergencies and disasters. Continued development and refinement of the system can further increase its impact and effectiveness.

## **5 QUALITY ASSURANCE**

In quality assurance phase which is mainly based on Test plan including testing strategies and types of testing applied to ensure the reliability and accuracy of the application to give the user a great and error free learning experience. Since

satisfaction of end user is a first and foremost priority, thus to ensure it, a proper testing mechanism was devised and the results are tabulated in the form of test cases and to trace each test case against desired functional requirement a requirement traceability matrix have been devised which include test case ID against each and every functional requirement desired by user.

## 5.1 Introduction

Quality assurance is an essential process that ensures that the final product or service meets the desired quality standards. In the case of AFAD, quality assurance is crucial to ensuring that the system is reliable, efficient, and meets the requirements of the end-users. The QA process should involve testing and verifying the system's functionalities, identifying and addressing potential issues or bugs, and ensuring that the system is secure and user-friendly. Quality assurance should also include documentation of the testing and verification process and ongoing monitoring of the system's performance to ensure that it continues to meet the desired quality standards. By implementing a robust quality assurance process, AFAD can ensure that it delivers a high-quality and reliable system that meets the needs of its users. The AFAD system will pass through three levels of testing i.e., unit testing, system or integration testing and acceptance testing in each level multiple functions need to be tested:

### **Documentation Items: -**

- System Requirement Specification.
- Design Specification.

### **Mobile Application Items: -**

- AFAD's Application Customer module (Frontend / Backend).
- AFAD's Application Admin module (Frontend/Backend).

### **Hardware Items: -**

- Signal levels and frequencies of transmitter and receiver at various test points.
- Configuration of GPS, Flight controllers and Brushless motors.

## 5.2 Traceability Matrix

Requirement Traceability Matrix														
Req-ID	Test Case ID	UFR1-TC01	UFR1-TC02	UFR1-TC03	UFR1-TC04	UFR2-TC01	UFR2-TC02	UFR3-TC01	UFR4-TC01	UFR5-TC01	UFR5-TC02	UFR6-TC01	UFR6-TC02	# of Test cases for respective requirement
UFR1		✗	✗	✗	✗									3
UFR2						✗	✗							2

**Figure 5.1 Traceability Matrix (User FR)**

Req-ID	Test Case ID	AFR1-TC01	AFR1-TC02	AFR2-TC01	AFR2-TC02	AFR3-TC01	AFR4-TC01	AFR5-TC01	AFR6-TC01	AFR7-TC01	AFR8-TC01	# of Test cases for respective requirement
AFR1		X	X									2
AFR2				X	X							2
AFR3						X						1
AFR4							X					1
AFR5								X				1
AFR6									X			1
AFR7										X		1
AFR8											X	1

**Figure 5.2 Traceability Matrix (Admin FR)**

## 5.3 Test Plan

### 5.3.1 Test Plan Identifier

**Table 5.1 Test Plan Identifier**

Date	Version	Description	Author
March 15 , 2023	1.0	Autonomous First Aid Drone - TP 1.0	Yusra Tahir Shahzeb Noor

### 5.3.2 References

Documents that support test plan for AFAD are:

- IEEE 829 standards and guidelines.
- Software Requirement Design Specification.
- Software Design Document.
- Software Plan.
- Project Proposal.

### 5.3.3 Introduction

This test plan describes the testing approach and overall framework that will drive the testing of the AFAD Version 1.0 – that includes software (mobile application) and hardware (drone) testing supports the following objectives:

### 5.3.4 Objectives: -

- The main aim for the making to this document is to enlist and plan all the testing techniques which will be used to test the software.
- It will also include a proactive approach to find possible defects or risks and based on the techniques of this document we will be lowering that risk.
- This document will be executed before the actual implementation of the product to save time in the future and to get to know all possible risks in advance.
- To communicate to the responsible parties, the items to be tested, set expectations around schedule, and define environmental needs.
- To communicate important aspects like test estimation, test scope, test Strategy, so it can be reviewed by Management Team and re-used for other projects.
- To define how the tests will be conducted.

### 5.3.5 Software Risk Issues

The critical areas of Autonomous First Aid Drone (AFAD) system are:

- A.** Ability to integrate Notification pushing and Google map API with application.
- B.** Unavailability of hardware modules i.e. Telemetry, GPS etc.
- C.** Integration issues of Admin and Customer module.
- D.** Additional Resources Required.
- E.** Hardware configuration issues.
- F.** Obsolescence of libraries.
- G.** Aggressive deadlines.
- H.** Database crash.

### 5.3.6 Testing Approach (Strategy)

#### 5.3.6.1 *Test Support Tools:* -

The tools that will be employed to support this Test Plan are:

**Table 5.2 Tools for testing**

Tool Type	Tool Name	Open Source
<b>Test Execution tool</b>	<ul style="list-style-type: none"> <li>• Test Project</li> <li>• Test Complete</li> <li>• Appium</li> </ul>	✓
<b>Static Analysis Tool</b>	Synopsis Coverity	✓
<b>Configuration Management Tool</b>	Chef	✓

### 5.3.7 Testing Levels: -

The testing for the AFAD system will consist of Unit, System/Integration (combined) and Acceptance test levels. It is hoped that there will be at least one full time independent test person for system/integration testing.

#### 5.3.7.1 *Unit Test:* -

Unit Testing will be done by the developer and will be approved by the development team leader. Proof of unit testing (test case list, sample output, data printouts, defect information) must be provided by the programmer to the team leader before unit testing will be accepted and passed on to the test person. All unit test information will also be provided to the test person.

**Table 5.3 Modules for Unit Test**

Scenario ID #	Unit Test Scenario Description
<b>C U S T O M E R - M O D U L E</b>	
<b>01</b>	Login/Sign up verification through OTP.
<b>02</b>	Google Map API to take location.
<b>03</b>	View Profile and Update.
<b>04</b>	Feedback module.
<b>A D M I N I S T R A T O R - M O D U L E</b>	
<b>05</b>	Receive the given location, and medicine details.
<b>06</b>	View all requests, request status and user details.
<b>07</b>	Accept and reject requests.

#### **5.3.7.2 System/Integration Testing: -**

System/Integration Testing will be performed by the test manager and development team leader with assistance from the individual developers as required. Tester will apply the specified tools to test the application. Application will enter into System/Integration test after all critical defects have been corrected.

**Table 5.4 Modules for Integration Testing**

Scenario ID #	Integration Test Scenario Description
<b>01</b>	Integration of Google map API with Customer module.
<b>02</b>	Integration of Customer and Administrator module to send details.
<b>03</b>	Integration of feedback send and receive option.

#### **5.3.7.3 Acceptance Testing: -**

Acceptance Testing will be performed by the actual end users with the assistance of the test manager and development team leader. The acceptance test will be completion of the System/Integration test process. Application will enter into Acceptance test after all critical and major defects have been corrected.

#### **5.3.8 Features to Be Tested**

#### **5.3.8.1 Customer Module Features: -**

**UFR1:** As a customer signup through phone number and verification through OTP.

**UFR2:** As a customer, add location coordinates in both text and longitude-latitude to get their first aid kit along with medical supplies details.

**UFR3:** As a customer, fetched location and entered first aid detail could send to admin side.

**UFR4:** As a customer, can view profile and can update it.

**UFR5:** As a customer, can provide feedback.

**UFR6:** As a customer, notified when a request is accepted or rejected.

#### **5.3.8.2 Administrator Module Features: -**

**AFR1:** As an admin, admin can login using valid email and password credentials.

**AFR2:** As an admin, accept and reject first aid requests.

**AFR3:** As an admin, view all requests, request status and user details.

**AFR4:** As an admin, view feedbacks.

**AFR5:** As an admin, notified for each first aid request along with user, location and first aid details.

**AFR6:** As an admin configure drone for autonomous flight.

**AFR7:** As an admin, plan mission by writing waypoints (Location: altitude and longitude), delays, Flight modes into flight controller by using Mission Planner.

**AFR8:** As an admin, place first aid kit with drone, arm drone and change drone modes if required to accomplish mission.

#### **5.3.9 Test Cases**

Test Cases are a set of specified inputs with implied conditions and a procedure needed for testing and an expected output against it.

**Table 5.5 Test Case Phone Number Signup and OTP Verification**

<b>Test ID</b>	UFR1-TC01
<b>Test name</b>	Phone Number Signup and OTP Verification
<b>Date of test</b>	March 21, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can sign up for the Drone Ambulance service using their phone number and complete the verification process through OTP
<b>Input</b>	Phone Number: 3088931468
<b>Expected output</b>	User receives OTP via SMS and can enter it to complete the verification process
<b>Actual output</b>	User receives OTP via SMS and enters it to complete the verification process
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.6 Signup and OTP Test Case**

<b>Test ID</b>	UFR1- TC02
<b>Test name</b>	Signup and OTP
<b>Date of test</b>	March 21, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	To verify the functionality of customer signup through phone number and verification through OTP
<b>Input</b>	Phone Number: +923088931468, OTP: 690123
<b>Expected output</b>	Application Should Display Signup form
<b>Actual output</b>	Signup form is displayed
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.7 OTP Verification Timeout Test Case**

<b>Test ID</b>	UFR1- TC03
<b>Test name</b>	OTP Verification Timeout

<b>Date of test</b>	March 21, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that the OTP verification process times out after a certain period.
<b>Input</b>	Phone Number: +923088931468
<b>Expected output</b>	System should prompt an error message "OTP Verification Timeout"
<b>Actual output</b>	System prompts an error message "OTP Verification Timeout"
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.8 OTP Resend Functionality Test Case**

<b>Test ID</b>	UFR1- TC04
<b>Test name</b>	OTP Resend Functionality
<b>Date of test</b>	March 21, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can request for OTP resend.
<b>Input</b>	Phone Number: +923088931468, Request for Resend
<b>Expected output</b>	User should receive a new OTP via SMS.
<b>Actual output</b>	User receives a new OTP via SMS.
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.9 Location Coordinates Retrieval Test Case**

<b>Test ID</b>	UFR2- TC01
<b>Test name</b>	Location Coordinates Retrieval
<b>Date of test</b>	March 25, 2023

<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can retrieve their location coordinates through the Google Maps API in both text and longitude-latitude formats
<b>Input</b>	Enable current location for Application and press location icon
<b>Expected output</b>	Location coordinates should be displayed in both text and longitude-latitude formats
<b>Actual output</b>	Location coordinates displayed in both text and longitude-latitude formats
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.10 Adding Medical Supplies Test Case**

<b>Test ID</b>	UFR2- TC02
<b>Test name</b>	Adding Medical Supplies
<b>Date of test</b>	2023-05-01
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can add medical supplies details to their first aid kit
<b>Input</b>	Emergency type, Severity, Medical supplies details
<b>Expected output</b>	Medical supplies details should be added to the customer's first aid kit
<b>Actual output</b>	Medical supplies details are added to the customer's first aid kit
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.11 Send Location and Emergency Details to Admin Test Case**

<b>Test ID</b>	UFR3- TC01
----------------	------------

<b>Test name</b>	Send Location and Emergency Details to Admin
<b>Date of test</b>	March 25, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can send their location and medical supplies details to the admin using the Send button and receive a token confirmation message
<b>Input</b>	<ol style="list-style-type: none"> <li>1. Longitude: 73.056209, Latitude: 33.684422,</li> <li>2. Emergency type: Accident Severity: Medium Description: Bandage</li> <li>3. Enter Send button</li> </ol>
<b>Expected output</b>	<ol style="list-style-type: none"> <li>1. Details sent on admin side</li> <li>2. User should receive a token "Your request is successfully sent to admin. Wait for approval"</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>1. Details sent on admin side</li> <li>2. User receives a token "Your request is successfully sent to admin. Wait for approval"</li> </ol>
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.12 View and Update Customer Profile Test Case**

<b>Test ID</b>	UFR4- TC01
<b>Test name</b>	View and Update Customer Profile
<b>Date of test</b>	March 26, 2023

<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can view and update their profile information, including username, and email.
<b>Input</b>	Customer logs in and navigates to their profile page.
<b>Expected output</b>	<ol style="list-style-type: none"> <li>1. Profile page should display current username, phone, and email. Customer can edit and save any changes made.</li> <li>2. Updated user information in firebase</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>1. Profile page displays current username, phone, and email. Customer is able to successfully edit and save any changes made to their profile.</li> <li>2. Updates user information in firebase</li> </ol>
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.13 Feedback Submission with 5-star Rating Test Case**

<b>Test ID</b>	UFR5- TC01
<b>Test name</b>	Feedback Submission with 5-star Rating
<b>Date of test</b>	March 26, 2023
<b>Name of application</b>	Drone Ambulance

<b>Description</b>	Verify that customers can submit feedback with a 5-star rating to the admin side
<b>Input</b>	Rating: 5 stars, Feedback message: "Great service!".
<b>Expected output</b>	<ol style="list-style-type: none"> <li>1. Feedback should be sent to admin side.</li> <li>2. System should store the feedback and rating in firebase</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>1. Feedback send to admin side.</li> <li>2. System stores the feedback and rating</li> </ol>
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.14 Feedback Submission without Rating Test Case**

<b>Test ID</b>	UFR5- TC02
<b>Test name</b>	Feedback Submission without Rating
<b>Date of test</b>	2023-05-01
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that customers can submit feedback without a rating
<b>Input</b>	Feedback message: "Great service!".
<b>Expected output</b>	<ol style="list-style-type: none"> <li>3. Feedback should be sent to admin side.</li> <li>4. System should store the feedback and rating in firebase</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>3. Feedback send to admin side.</li> <li>4. System stores the feedback and rating</li> </ol>
<b>Test Role (Actor)</b>	Customer / End User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.15 Notification on Request Acceptance Test Case**

<b>Test ID</b>	UFR6- TC01
<b>Test name</b>	Notification on Request Acceptance
<b>Date of test</b>	March 27, 2023

<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that the customer is notified through a push notification when their request is accepted by an admin
<b>Input</b>	Username , Phone#
<b>Expected output</b>	Customer should receive a push notification with message "Your request has been accepted by admin and aid will be there in a while."
<b>Actual output</b>	Customer receives a push notification with message "Your request has been accepted by admin 456 and aid will be there in a while."
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.16 Notification on Request Rejection Test Case**

<b>Test ID</b>	UFR6- TC02
<b>Test name</b>	Notification on Request Rejection
<b>Date of test</b>	March 27, 2023
<b>Name of application</b>	Drone Ambulance
<b>Description</b>	Verify that the customer is notified through a push notification when their request is rejected.
<b>Input</b>	Username , Phone#
<b>Expected output</b>	Customer should receive a push notification with message "Your request has been rejected by admin due to availability issues. Your request has been transferred to 1122 emergency number."
<b>Actual output</b>	Customer receives a push notification with message "Your request has been rejected by admin due to availability issues. Your request has been transferred to 1122 emergency number."
<b>Test Role (Actor)</b>	Customer / End-User
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.17 Admin Login Test Case**

<b>Test ID</b>	AFR1- TC01
<b>Test name</b>	Admin Login
<b>Date of test</b>	March 27, 2023
<b>Name of application</b>	Drone Ambulance.

<b>Description</b>	Verify that an admin can log in using valid email and password credentials.
<b>Input</b>	Enter valid email and password.
<b>Expected output</b>	Application should redirect to the admin dashboard.
<b>Actual output</b>	Application redirects to the admin dashboard.
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.18 Invalid Admin Login Test Case**

<b>Test ID</b>	AFR1- TC02
<b>Test name</b>	Invalid Admin Login
<b>Date of test</b>	March 27, 2023
<b>Name of application</b>	Drone Ambulance.
<b>Description</b>	Verify that the system does not allow an invalid email address or password
<b>Input</b>	Enter invalid email and valid password
<b>Expected output</b>	Application should prompt an error message "Invalid Email Address".
<b>Actual output</b>	Application prompts an error message "Invalid Email Address".
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.19 Accept First Aid Request Test Case**

<b>Test ID</b>	AFR2- TC01
<b>Test name</b>	Accept First Aid Request
<b>Date of test</b>	March 28, 2023
<b>Name of application</b>	Drone Ambulance.

<b>Description</b>	Verify that an admin can accept a first aid request
<b>Input</b>	Select request option and accept a particular request.
<b>Expected output</b>	<ol style="list-style-type: none"> <li>1. Request should be accepted</li> <li>2. Case status should be changed</li> <li>3. The request should no more visible in requests</li> <li>4. Notification should be pushed to user</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>1. Request accepted</li> <li>2. Case status changed to accepted</li> <li>3. The request is no more visible in request</li> <li>4. Notification pushed to user.</li> </ol>
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.20 Reject First Aid Request Test Case**

<b>Test ID</b>	AFR2- TC02
<b>Test name</b>	Reject First Aid Request
<b>Date of test</b>	March 28, 2023
<b>Name of application</b>	Drone Ambulance.
<b>Description</b>	Verify that an admin can reject a first aid request
<b>Input</b>	Select request option and reject/discard a particular request.
<b>Expected output</b>	<ol style="list-style-type: none"> <li>1. Request should be discarded</li> <li>2. Notification should be pushed to user</li> </ol>
<b>Actual output</b>	<ol style="list-style-type: none"> <li>1. Request accepted</li> <li>2. Notification pushed to user.</li> </ol>
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.21 Admin View Requests Test Case**

<b>Test ID</b>	AFR3- TC01
<b>Test name</b>	Admin View Requests
<b>Date of test</b>	March 29, 2023
<b>Name of application</b>	Drone Ambulance.

<b>Description</b>	Verify that the admin can view all requests, request status and user details
<b>Input</b>	Click on "Requests" button
<b>Expected output</b>	Application should show all requests along with their status and user details should be displayed
<b>Actual output</b>	Showing all requests along with their status and user details should be displayed
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.22 View Feedbacks Test Case**

<b>Test ID</b>	AFR4- TC01
<b>Test name</b>	View Feedbacks
<b>Date of test</b>	March 29, 2023
<b>Name of application</b>	Drone Ambulance.
<b>Description</b>	Verify that the admin can view all feedbacks
<b>Input</b>	Click on "Feedback" button
<b>Expected output</b>	Application should show all feedback along with username.
<b>Actual output</b>	Showing all feedback along with username.
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.23 Request Notification Test Case**

<b>Test ID</b>	AFR5- TC01
<b>Test name</b>	Request Notification
<b>Date of test</b>	April 2, 2023
<b>Name of application</b>	Drone Ambulance.
<b>Description</b>	Verify that admin is notified for each first aid request along with user, location, and first aid details

<b>Input</b>	User submits a first aid request
<b>Expected output</b>	Admin should receive a notification with user, location, and first aid details
<b>Actual output</b>	Admin receives a notification with user, location, and first aid details
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.24 Drone Configuration in Ardupilot Mission Planner Test Case**

<b>Test ID</b>	AFR6- TC01
<b>Test name</b>	Drone Configuration in Ardupilot Mission Planner
<b>Date of test</b>	April 5, 2023
<b>Name of application</b>	AFAD
<b>Description</b>	Verify that the admin can configure the drone in Ardupilot Mission Planner for autonomous flight.
<b>Input</b>	Connect drone to Ardupilot Mission Planner, configure flight plan, Save configuration
<b>Expected output</b>	Drone configuration should be saved and the drone is ready for autonomous flight.
<b>Actual output</b>	Drone configuration saved and the drone is ready for autonomous flight
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.25 Plan Mission Test Case**

<b>Test ID</b>	AFR7-TC01
<b>Test name</b>	Plan Mission
<b>Date of test</b>	April 10, 2023
<b>Name of application</b>	AFAD
<b>Description</b>	Verify that admin can plan a mission using Mission Planner by writing waypoints, delays, and flight modes into the flight controller
<b>Input</b>	<ol style="list-style-type: none"> <li>1. Select home</li> <li>2. Waypoints: (33.642047, 73.081719), (33.640781, 73.082623), (33.641295, 73.084669),</li> <li>3. Delay: 10 seconds, on landing waypoints Flight</li> <li>4. Mode: Auto</li> </ol>
<b>Expected output</b>	Mission should be upload to the flight controller and drone takes off and follows the planned mission
<b>Actual output</b>	Mission is uploaded to the flight controller and drone takes off and follows the planned mission
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

<b>Test ID</b>	AFR8- TC01
----------------	------------

<b>Test name</b>	Place First Aid Kit and Arm Drone
<b>Date of test</b>	2023-05-01.
<b>Name of application</b>	AFAD
<b>Description</b>	Verify that the admin can place a first aid kit on the drone, arm the drone, and change the drone mode if required to accomplish the mission
<b>Input</b>	First Aid Kit, Arm Drone, Change Drone Mode
<b>Expected output</b>	The drone is armed, the first aid kit is placed, and the drone is in the appropriate mode for the mission
<b>Actual output</b>	The drone is armed, the first aid kit is placed, and the drone is in the appropriate mode for the mission
<b>Test Role (Actor)</b>	Admin
<b>Test verified by</b>	Ms. Mehwish Mukhtar (Supervisor)

**Table 5.26 Place First Aid Kit and Arm Drone Test Case**

## **6 USER MANUAL**

This chapter would serve as help for the user using AFAD system. It contains all the necessary information with pictures for giving the user a clear idea of how to use this application. The purpose of the manual is to get the user to know the sequence of steps to perform a specific task. It also includes the requirements of the application.

### **6.1 Introduction**

A user manual is required for Autonomous First Aid Drone and Application (AFAD) to ensure that users have a clear understanding of how to operate and use the system safely and effectively. The AFAD system involves complex technology and requires specialized knowledge to operate. The user manual provides instructions on how to use the system, including information on how to control the drone, interact with the application, and follow safety protocols. It also provides troubleshooting guidelines in case of any issues or errors encountered while using the system. Overall, a user manual is essential to ensure that users can utilize the AFAD system to its full potential while maintaining safety and efficiency.

### **6.2 Hardware/Software Requirements for the System**

#### **6.2.1 Hardware Requirements**

- PC (desktop or laptop)
- CPU must be 5th generation or higher.
- Minimum 8 GB of RAM
- Smartphone 6.2.2

#### **6.2.2 Software Requirements**

- Operating System (Windows) must be version 7 or above.
- An android operating system should version 4.2 minimum. 6.2.3

#### **6.2.3 Other requirements**

- High speed Internet Connectivity should be available.
- The system must have space to install updates.

### **6.3 Installation guide for Application**

To install an Android Application “Drone Ambulance” from an APK file, please follow the steps below:

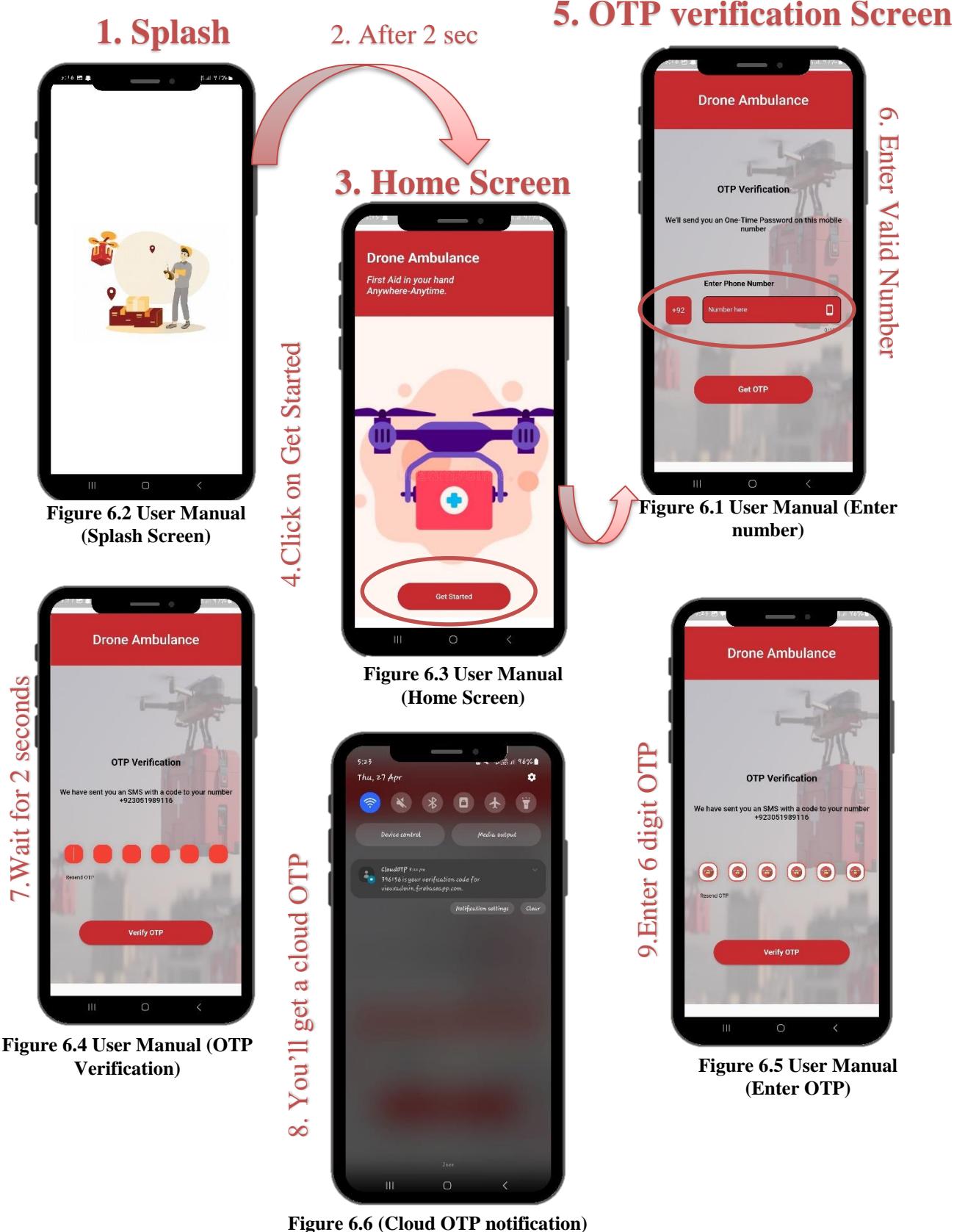
1. Go to the Settings menu on your Android device.
2. Scroll down and select Security or Privacy, depending on the Android version and device manufacturer.
3. Enable the "Unknown Sources" option. This allows you to install applications that are not from the Google Play Store.
4. Download the APK file from a trusted source.
5. Open the File Manager app on your Android device and navigate to the folder where the APK file is saved.
6. Tap on the APK file to start the installation process.
7. If prompted, review the permissions the application is requesting, and tap "Install" to proceed with the installation.
8. Wait for the installation to complete.
9. Once the installation is finished, you can open the application from the app drawer on your Android device.
10. There are two APK's generated one for admin side and other for customer. User only need to install customer side APK.

Note: It is important to be cautious when installing applications from unknown sources. Make sure to download the APK file from a trusted source, and to review the permissions the application is requesting before proceeding with the installation.

### **6.4 Operating Manual**

A complete operating manual has been given in this section to access all the functionalities of the system. The system is composed of different functionalities that are given below.

#### 6.4.1 Mobile Application User Manual



## 10. Registration Form

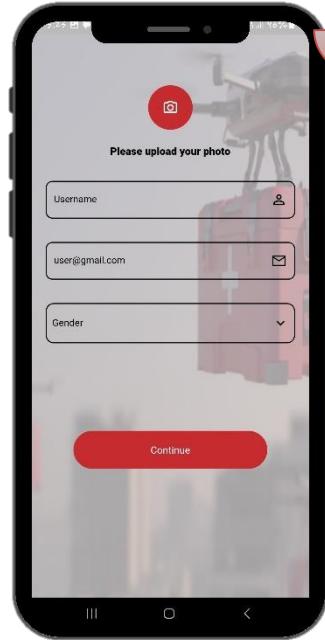


Figure 6.9 User Manual  
(Registration Form)

## 12. Drawer Screen

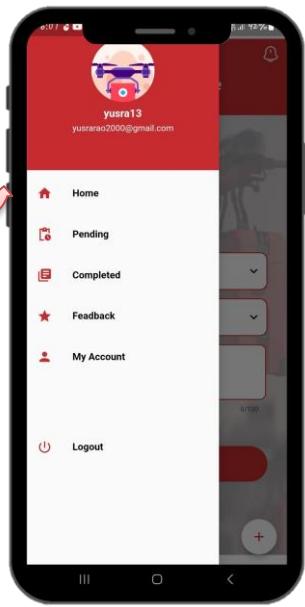


Figure 6.8User Manual  
(Drawer Screen)

11. Enter Required Fields

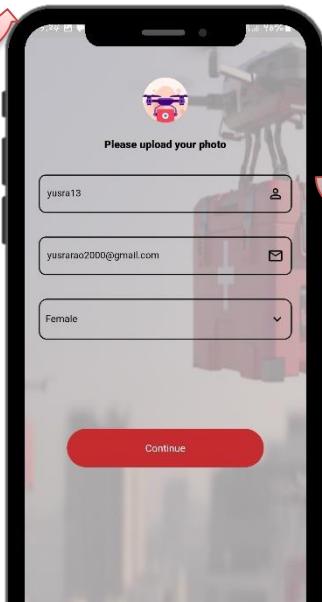


Figure 6.7 User Manual(User  
Registration)

13. Emergency Button is available

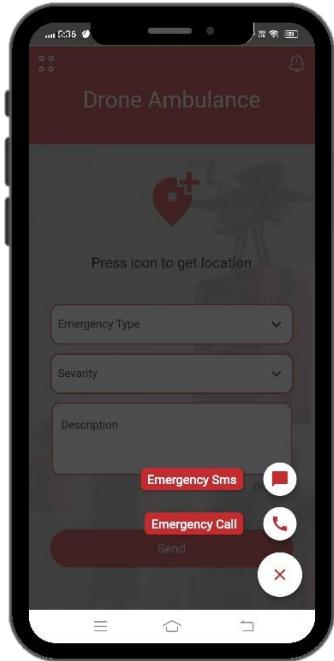


Figure 6.10 User Manual  
(Emergency Button)

14. Press location icon  
15. Enter Required Fields

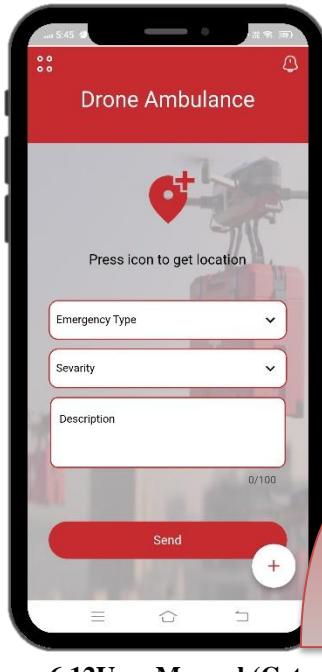


Figure 6.12User Manual (Get  
location and emergency details)

Lat: 33.7421586 , Long: 72.729856  
PPRJ+WCV, New City, Wah., Pakistan

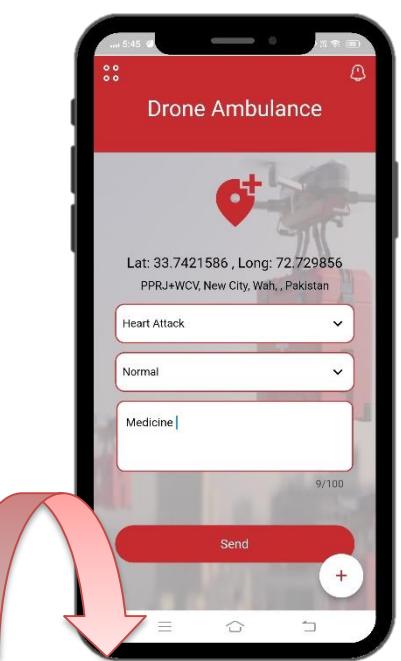


Figure 6.11 User Manual (Details  
Entry)

16. Request will have sent to Admin side

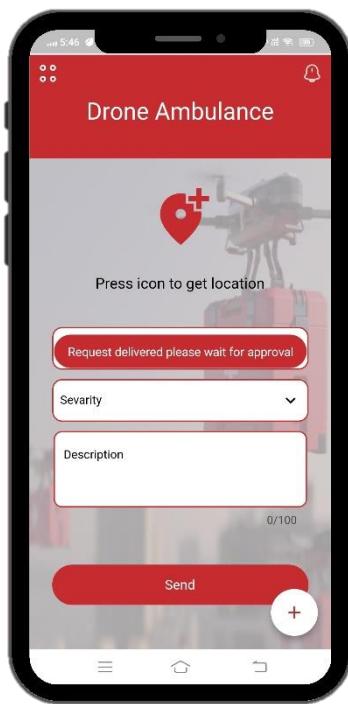


Figure 6.13 User Manual (Request sent)

## 17. Pending Screen

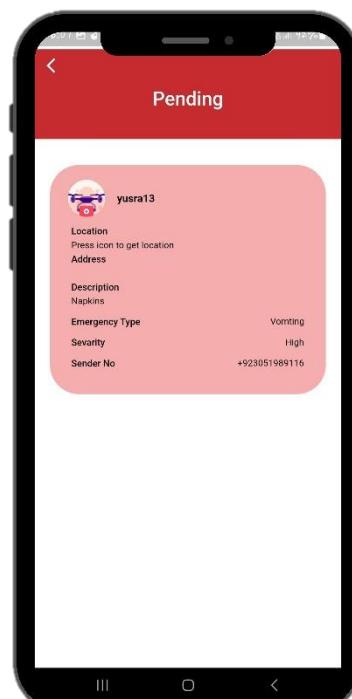


Figure 6.15 User Manual (Pending requests)

## 18. Feedback Screen

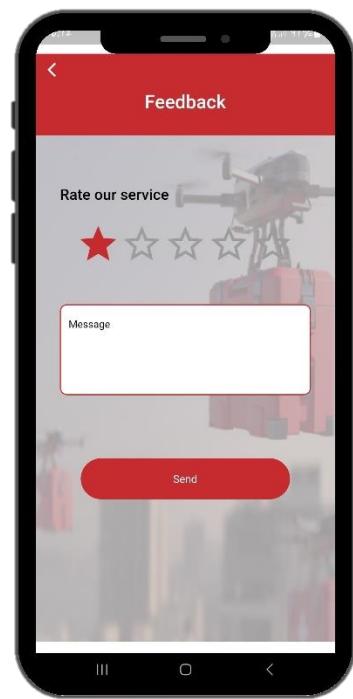


Figure 6.14 User Manual (Feed Back)

19. Give 5-star rating & Write message

## 20. User Profile

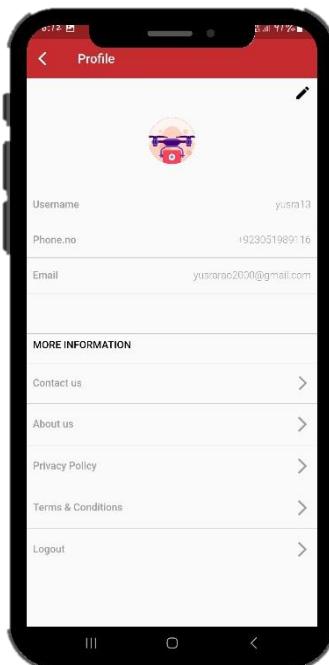


Figure 6.17 User Manual (View Profile)

## 21. Update Profile

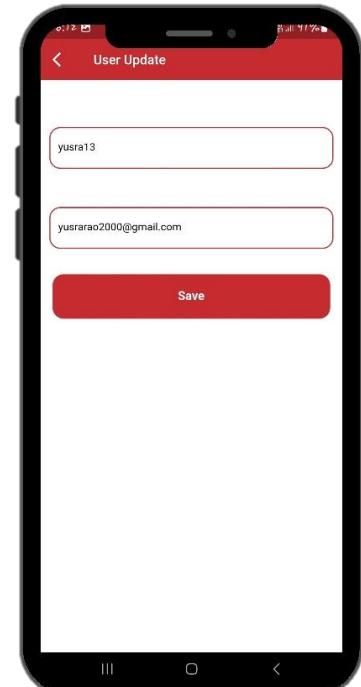


Figure 6.16 User Manual (Update Profile)

22. Enter Updated Username and Email

## 1. Admin Login

2. Enter valid Email and password

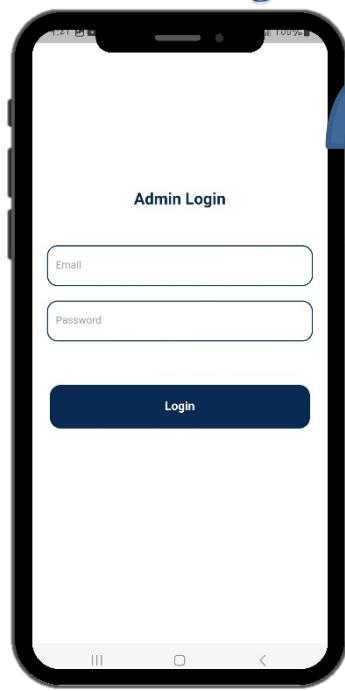


Figure 6.19 User Manual (Admin Login)

## 3. Admin Dashboard

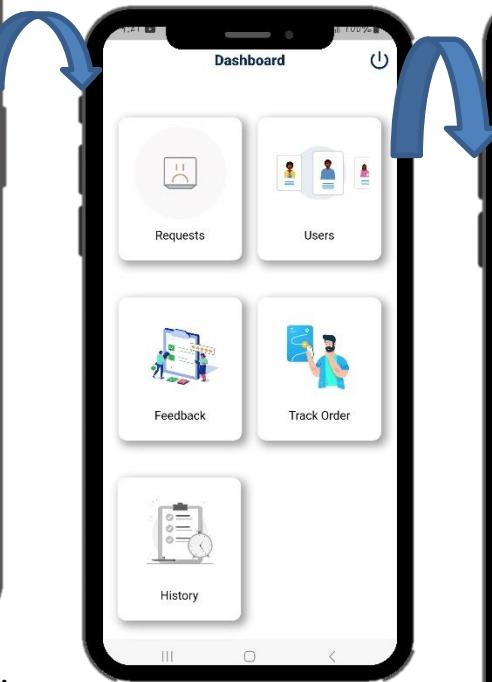


Figure 6.18 User Manual(Admin Dashboard)

## 5. Request Screen

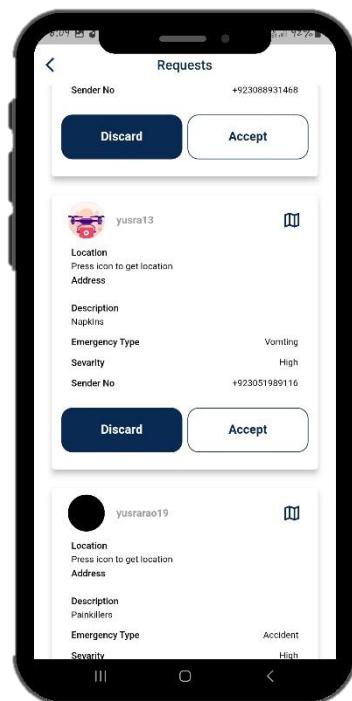


Figure 6.22 User Manual(Requests Screen)

6. When a request rejected  
Notification to user

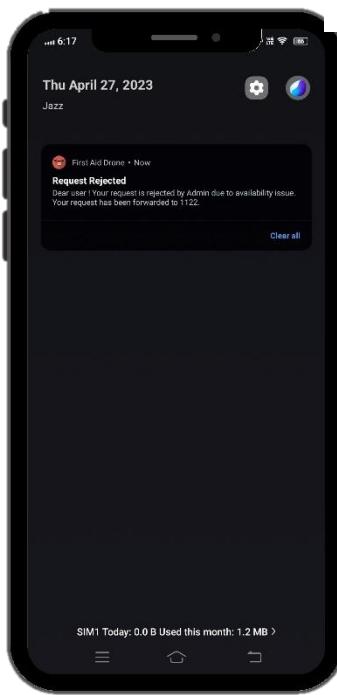


Figure 6.21 ( Request Rejected Notification)

4. Notification from user for first aid request  
Notification to user

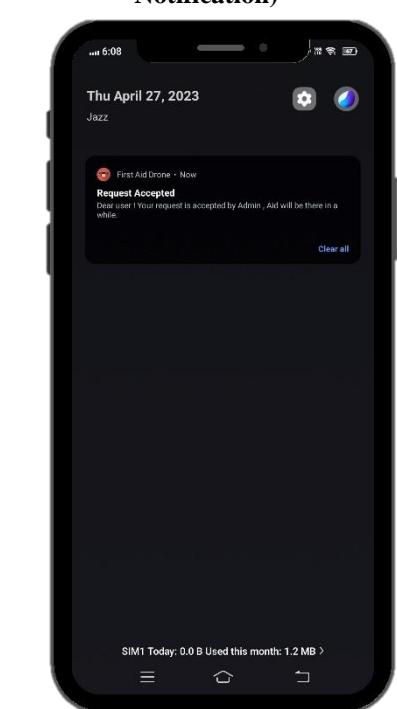


Figure 6.23 Request Accepted Notification

## 8. View Users

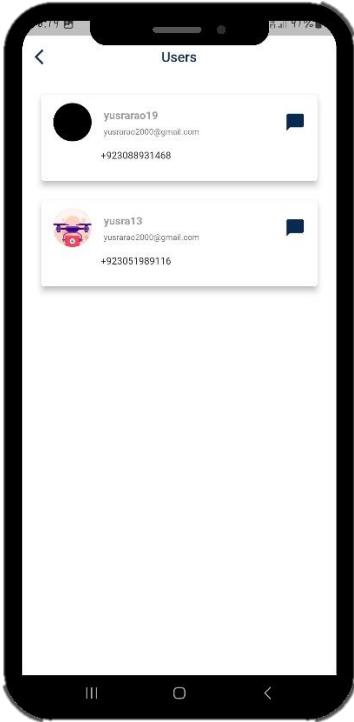


Figure 6.27 View all users

## 11. Aid Request

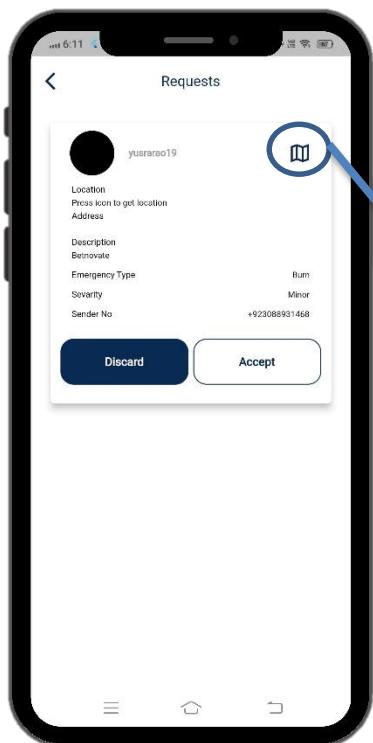


Figure 6.24 Google Map

## 9. Request History

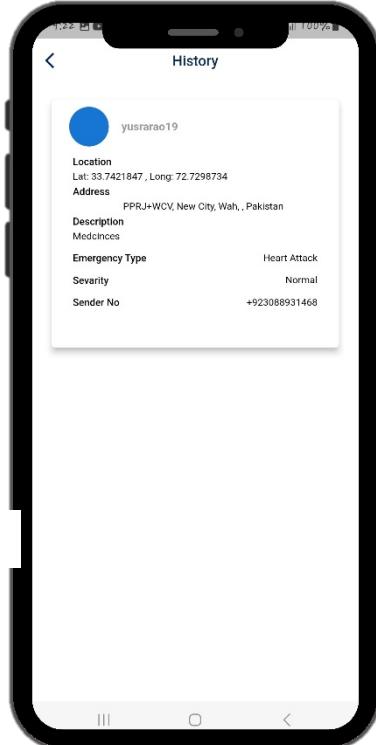


Figure 6.26 Requests history

12. Google map to check distance

## 10. Track Order

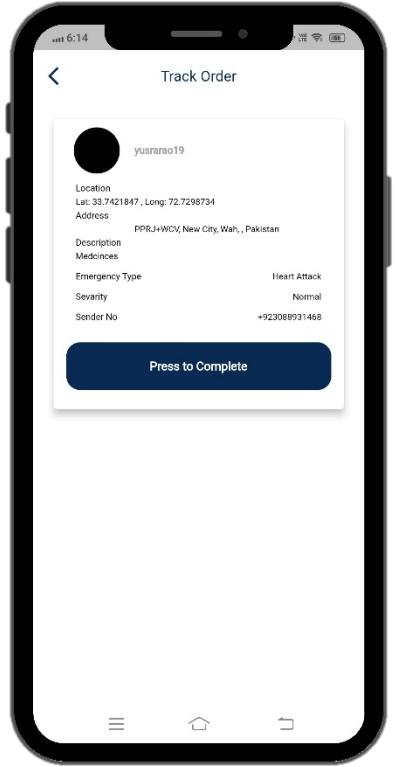


Figure 6.25 Track Order

## 13. Google Map

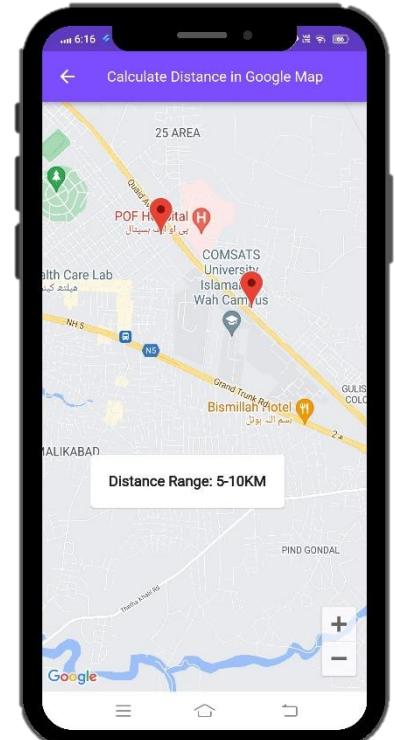


Figure 6.28 Distance Calculation

#### 6.4.2 Drone User Manual (Autonomous Flight)

1. Connect Drone's Flight Controller with Laptop USB port.

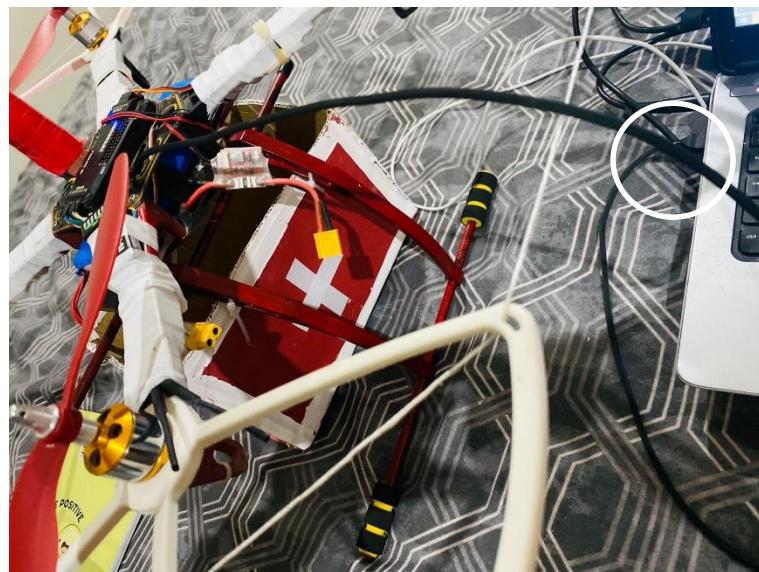


Figure 6.29 Flight Controller connection

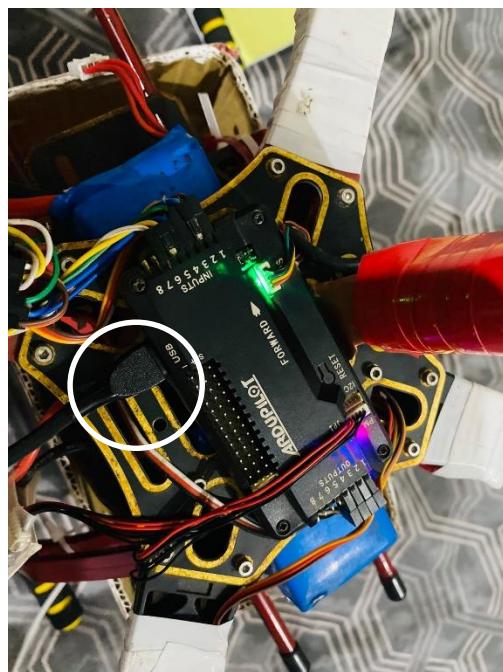
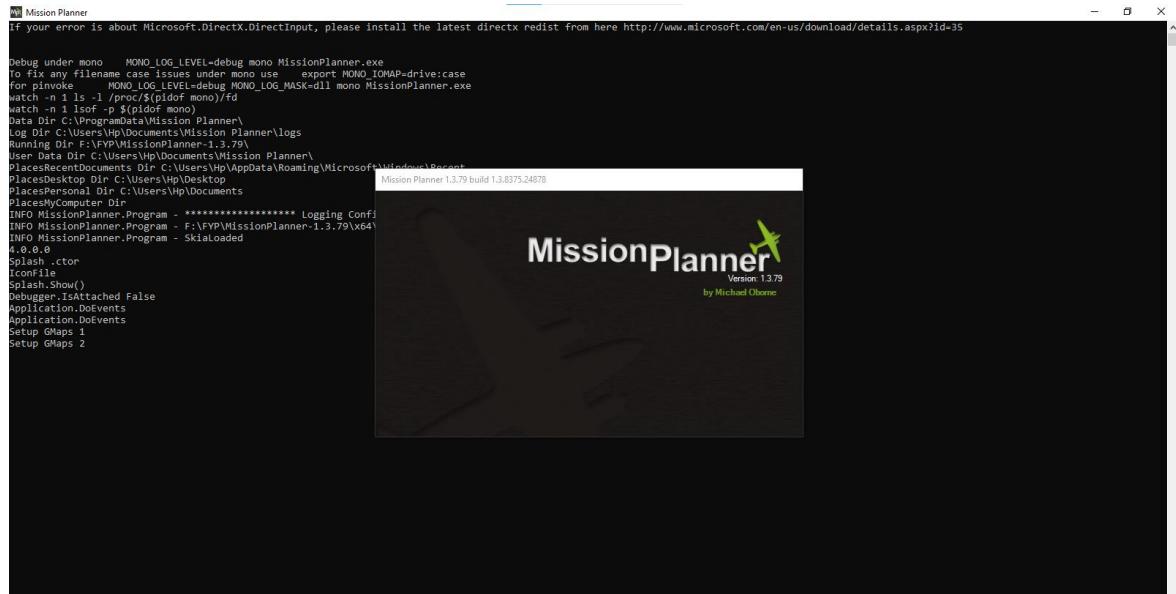
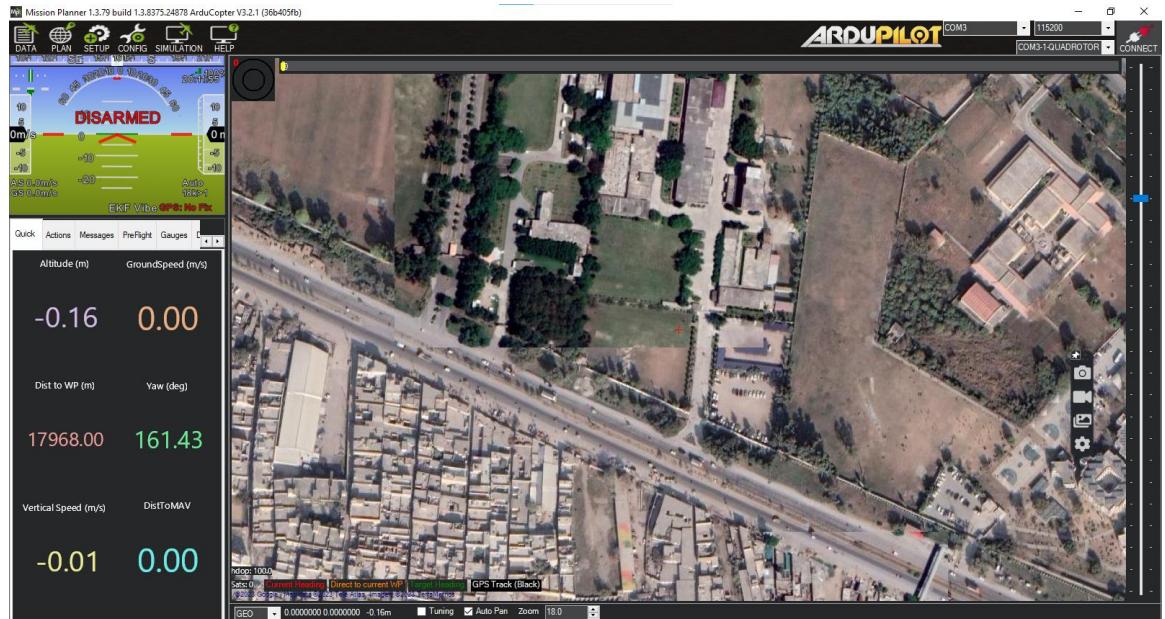


Figure 6.30 USB port in Flight Controller

## 2. Install and open Mission Planner.



**Figure 6.31 Startup Mission Planner**



**Figure 6.32 Mission Planner Interface**

### 3. Build connection between drone and Mission Planner.

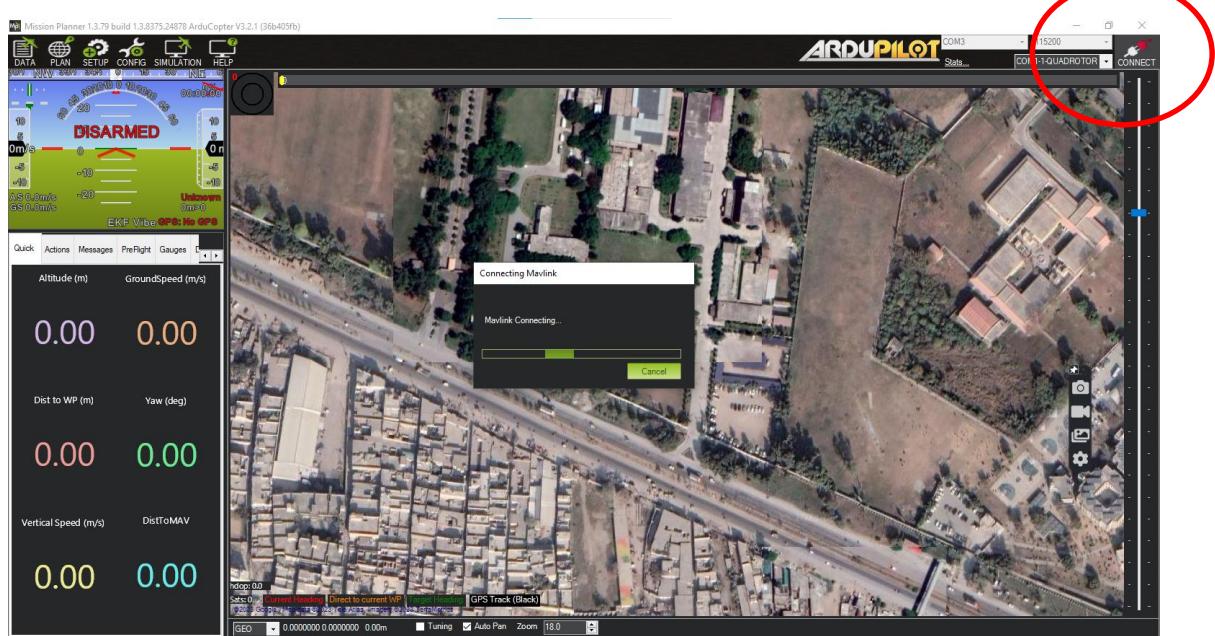


Figure 6.33 Build Connection

### 4. Connection has been built.

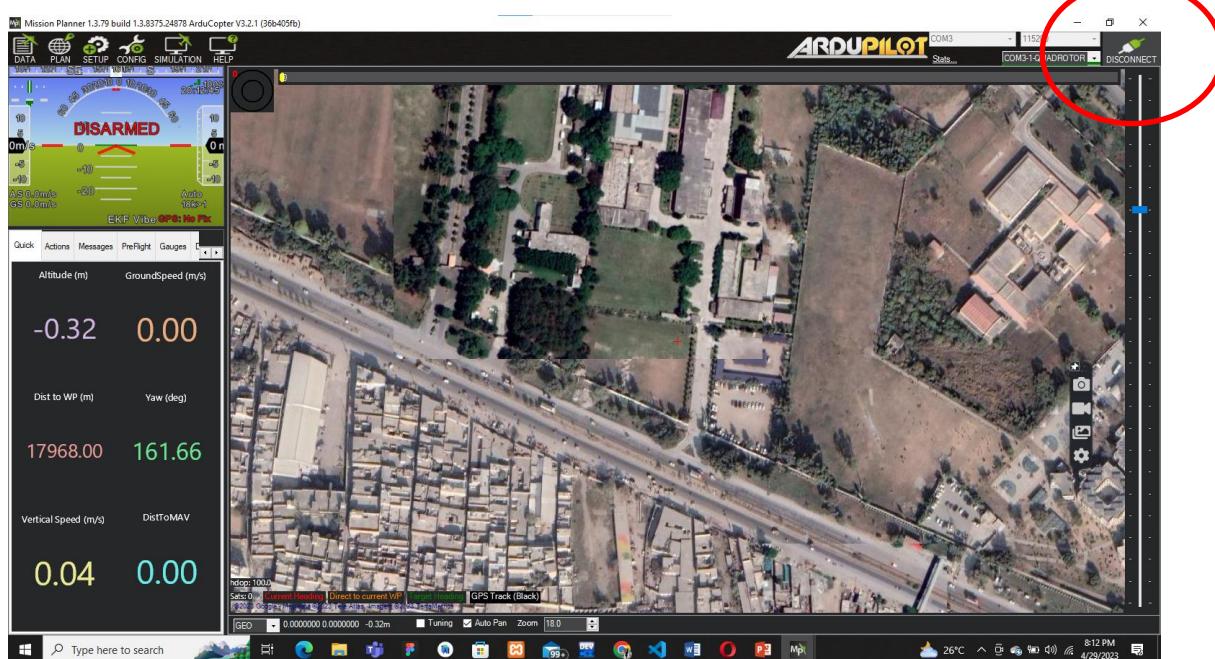


Figure 6.34 Connected

5. Wait for the GPS lock to have an exact stable drone location. GPS will lock when GPS light in flight controller turns solid blue.

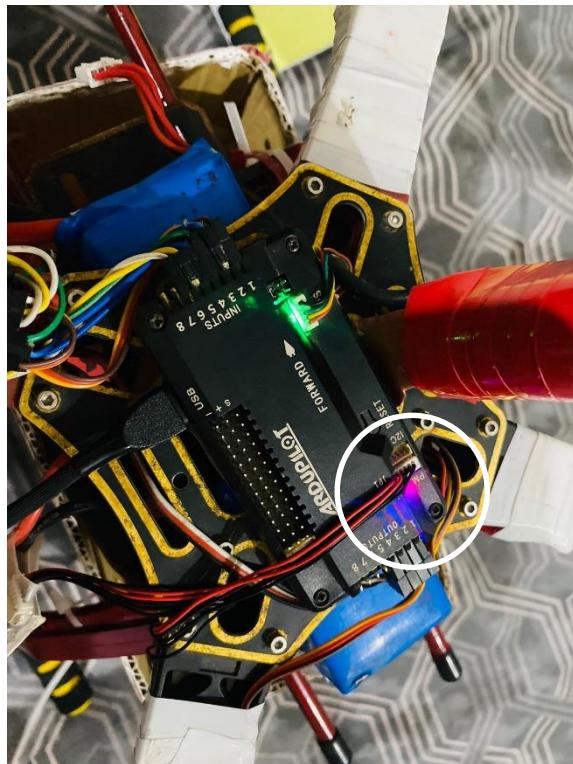


Figure 6.35 GPS Locked

6. When GPS get locked click on “PLAN” option from Mission Planner ribbon.

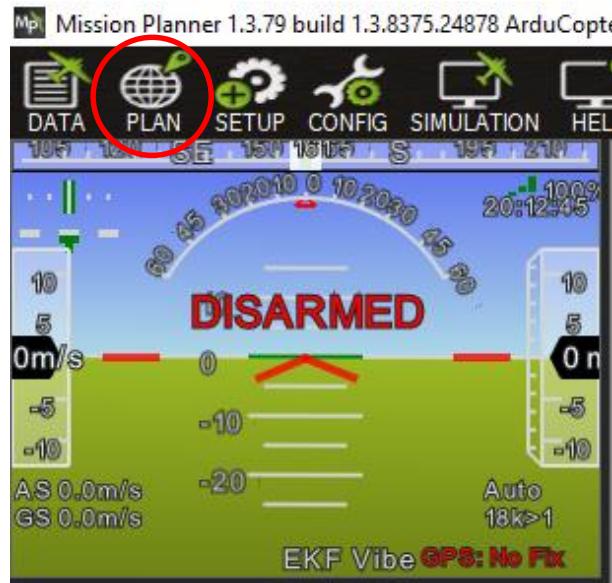


Figure 6.36 PLAN mission

7. Select the home location and make waypoints by setting parameters.

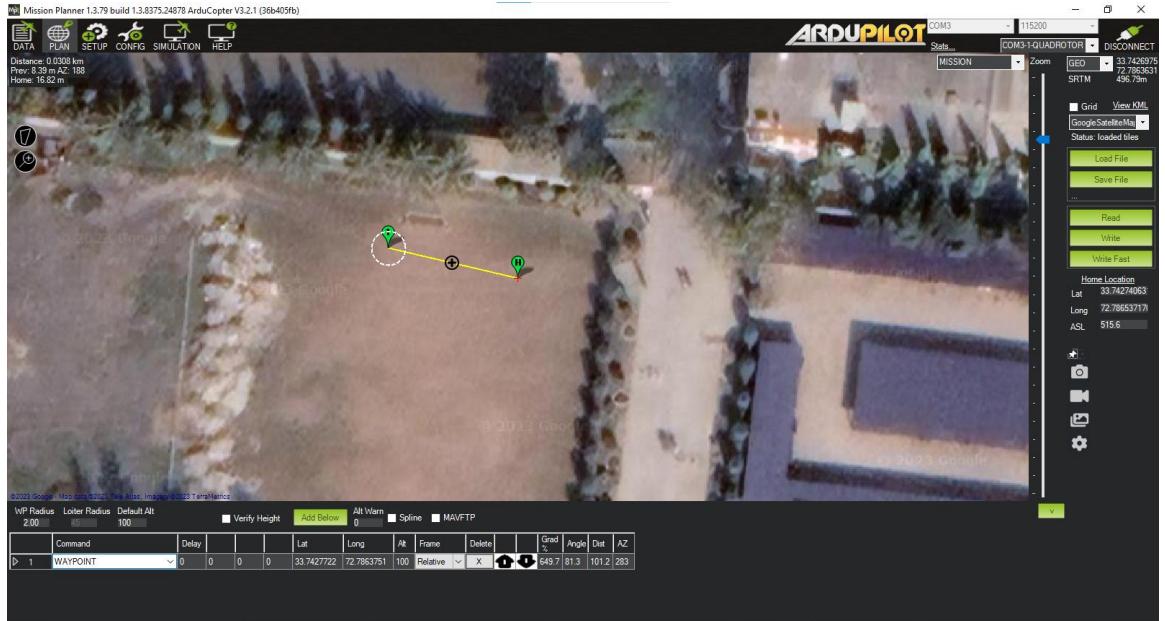


Figure 6.37 Selecting WP1

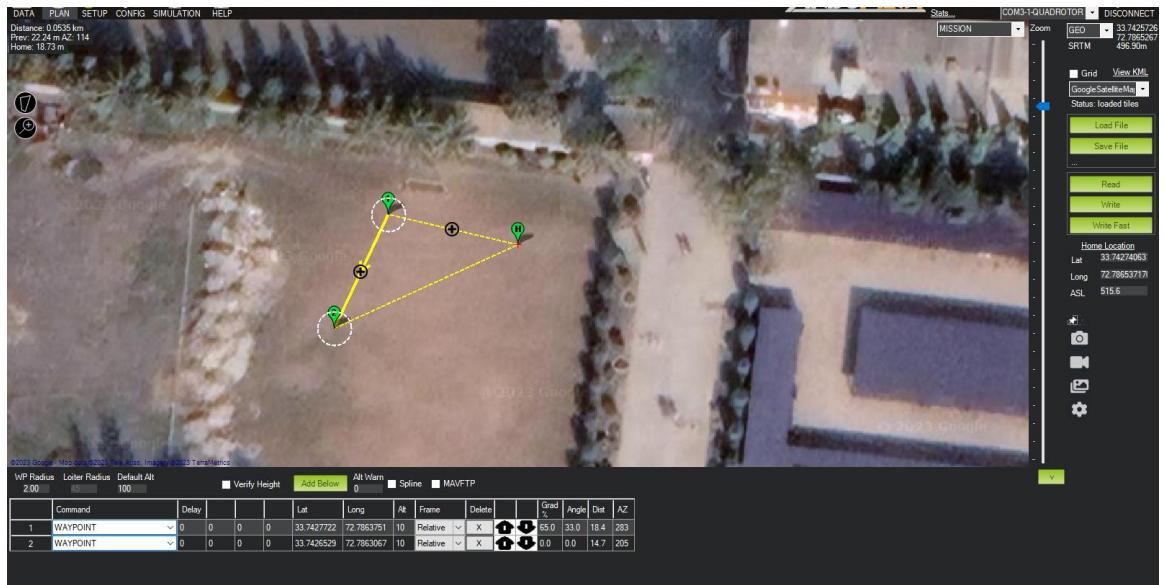
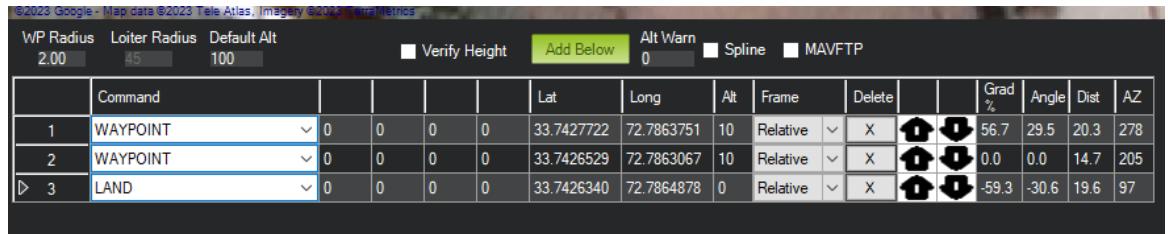


Figure 6.38 Selecting WP2

8. You can edit the parameters for example Altitude, Longitude, Latitude, Speed, and Delay at each Waypoint by double click.



The screenshot shows the Mission Planner interface with the following parameters:

	Command	Lat	Long	Alt	Frame	Delete	Grad %	Angle	Dist	AZ
1	WAYPOINT	33.7427722	72.7863751	10	Relative	X	56.7	29.5	20.3	278
2	WAYPOINT	33.7426529	72.7863067	10	Relative	X	0.0	0.0	14.7	205
▷ 3	LAND	33.7426340	72.7864878	0	Relative	X	-59.3	-30.6	19.6	97

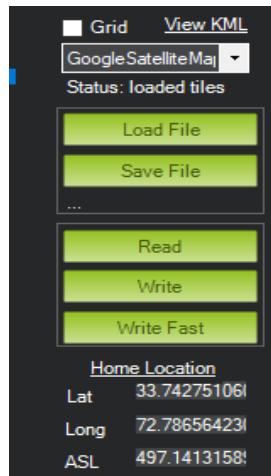
**Figure 6.39 Parameters**

9. Make waypoints and select a landing location where Aid is required.



**Figure 6.40 Selecting Land point**

10. Write WP's from right ribbon to the flight controller and Drone will be ready to take autonomous flight.



**Figure 6.41 Writing WP's to Flight Controller**

11. Plug in the battery terminals.



**Figure 6.42 Plug Battery Terminals**



**Figure 6.43 Battery Plugs**

12. Arm the drone using FLYSKY FS i6s transmitter.



**Figure 6.44 Transmitter FLYSKY FS i6s**

13. Drone is ready for autonomous flight.



**Figure 6.45 AFAD**



**Figure 6.46 Autonomous Flight**

## 7 GLOSSARY

This chapter involves appending additional information, data, or materials to a report can be useful in providing supplementary details that may not fit in the main body of the report. In the case of Final Year Project Report, appendices can be helpful in providing additional technical information about the drone's design, specifications, and performance.

Additionally, by including appendices, the we as authors of report can provide interested parties with more in-depth information that may be useful for future research, analysis, or development of similar system.

### 7.1 Definition, Abbreviation, Acronyms

**Table 7.1 Acronyms**

Term	Acronyms
<b>AFAD</b>	Autonomous First Aid Drone
<b>GPS</b>	Global positioning System
<b>API</b>	Application Programming Interface
<b>OTP</b>	One Time Password
<b>UFR</b>	User Functional Requirement
<b>AFR</b>	Admin Functional Requirement
<b>WP</b>	Way Point