

# **MED EVAC**

## *PROJECT REPORT*

Submitted in partial fulfillment of the requirements for the award Degree of

### **BACHELOR OF COMPUTER APPLICATIONS**

**Of the University of Calicut**

*Submitted By*

<b>LEZIN S</b>	<b>(NZAVBCA025)</b>
<b>MOHAMMED SHAMIL</b>	<b>(NZAVBCA011)</b>
<b>MUHAMMED ADNAN N</b>	<b>(NZAVBCA029)</b>
<b>ABDILLA NABEEL</b>	<b>( NZAVBCA022)</b>

**Under the supervision of**

**Mrs. JASEERA**



(Asst. Professor of Computer Science Department)

**DEPARTMENT OF COMPUTER SCIENCE**

**NASRA COLLEGE OF ARTS & SCIENCE**

**THIRURKAD -679325**

**2023– 2024**

**DEPARTMENT OF COMPUTER SCIENCE**

**NASRA COLLEGE OF ARTS & SCIENCE**

*(Affiliated to university of Calicut)*

**TIRURKAD - 67932**



**PROJECT WORK**

**CERTIFICATE**

Certified that this is a bonafied record of the project work

Done by

**MED EVAC**

**LEZIN S**

**(NZAVBCA025)**

**MOHAMMED SHAMIL**

**(NZAVBCA011)**

**MUHAMMED ADNAN N**

**(NZAVBCA029)**

**ABDILLA NABEEL**

**( NZAVBCA022)**

Submitted in partial fulfilment of the requirements for the award of the  
Degree Bachelor of Computer Application of the University of Calicut

Faculty Guide

External Examiner

Head of the Department

**DEPARTMENT OF COMPUTER SCIENCE**

**NASRA COLLEGE OF ARTS & SCIENCE**

**TIRURKAD, MALAPPURAM – 679325**



**CERTIFICATE**

This is to certify that this Project report entitled "MED EVAC" is a bonafied record of the project done by, LEZIN S, MOHAMMED SHAMIL, MUHAMMED ADNAN N, ABDILLA NABEEL of sixth semester BCA, Nasra College Of Arts And Science, Tirurkad, towards partial fulfilment of the requirement for the award of degree of Bachelor of Computer Applications University of Calicut

**Internal Guide**

**Department**

**JASEERA**

Department of Computer Science

Science

Nasra College Arts&Science, Thirurkkad

Thirurkkad

**Head of**

**MISNA VK (HOD)**

Department of Computer

Nasra College Arts&Science,

**Submitted for Viva-Voice Examination held on .....**

**INTERNAL EXAMINER**

**Name:**

**Signature:**

**Date:**

**EXTERNAL EXAMINAR**

**Name:**

**Signature:**

**Date:**

## DECLARATION

I hereby that this project work entitled 'MED EVAC' submitted at Nasra College of Arts & Science (Affiliated to University of Calicut) is a record of original work done us under the supervision and guidance of Mrs. JASEERA (Asst. Professor of Computer Science Department).

<b>LEZIN S</b>	<b>(NZAVBCA025)</b>
<b>MOHAMMED SHAMIL</b>	<b>(NZAVBCA011)</b>
<b>MUHAMMED ADNAN N</b>	<b>(NZAVBCA029)</b>
<b>ABDILLA NABEEL</b>	<b>( NZAVBCA022)</b>

Signature of Candidate

**LEZIN S:**

**MOHAMMED SHAMIL:**

**MUHAMMED ADNAN N:**

**ABDILLA NABEEL:**

## SYNOPSIS

Emergency medical services play a critical role in saving lives during accidents and medical emergencies. The efficiency of these services depends on their ability to respond quickly to incidents. To enhance the effectiveness of emergency response, "**MED EVAC**" has been proposed. This system aims to prioritize ambulances at traffic signals and provide real-time tracking to ensure timely and efficient response to emergencies.

The proposed system consists of two primary components: the Ambulance Traffic Signal Detecting System (ATSDS) and the Ambulance Tracking System (ATS). The ATSDS is designed to detect approaching ambulances and trigger pre-emption of traffic signals to allow them unobstructed passage through intersections. It uses a combination of technologies, such as Global Positioning System (GPS) to identify ambulances in real-time and communicate with traffic signals.

When an ambulance is detected, the ATSDS sends a priority signal to the traffic signal controller, causing the signal to turn green in the direction the ambulance is approaching. This enables the ambulance to navigate through the intersection without delays, reducing response times significantly.

The system by providing real-time tracking and monitoring of ambulance movements. Each ambulance is equipped with a GPS-enabled tracking device that transmits its location, speed, and status to a centralized monitoring system. Dispatchers can access this information on a digital map, allowing them to make informed decisions about dispatching the closest available ambulance to the emergency location.

The benefits of the Ambulance Traffic Signal Detecting and Tracking System are manifold. By providing priority passage through traffic signals, it reduces ambulance response times, which is crucial in critical medical situations. The real-time tracking feature enables better coordination of emergency medical services, leading to improved resource allocation and overall emergency management. Ultimately, this system has the potential to save more lives by optimizing emergency response operations and improving the efficiency of ambulance services.

## Table of Contents

<b>Chapter</b>	<b>Page no</b>
<b>1. INTRODUCTION</b>	<b>09</b>
<b>2. SYSTEM ANALYSIS</b>	<b>10</b>
2.1. Existing System	10
2.2. Proposed System	10
<b>2.3. Module Decryption</b>	<b>11</b>
2.3.1. Admin	11
2.3.2. Hospital	11
2.3.3. Officer	11
2.3.4. Driver	11
2.3.5. User	11
<b>2.4. Feasibility Study</b>	<b>12</b>
2.4.1. Operational Feasibility	12
2.4.2. Technical Feasibility	12
2.4.3. Economic Feasibility	13
<b>2.5. System environment</b>	<b>14</b>
2.5.1. Developer Requirement	14
2.5.2. User requirement	14

<b>3. METHODOLOGY</b>	<b>15</b>
<b>3.1. Introduction</b>	<b>15</b>
<b>3.2. Data flow diagram (DFD)</b>	<b>16</b>
Level 0	17
Level 1.1	18
Level 1.2	19
Level 1.3	20
Level 1.4	21
Level 1.5	22
<b>3.3. UML Diagrams</b>	<b>23</b>
3.3.1. Class Diagram	23
3.3.2. Use case diagram	24
3.3.3. Activity diagram	27
<b>3.4. Table Design</b>	<b>30</b>
3.4.1 Login Table	30
3.4.2 User Table	30
3.4.3 Ambulance Table	31
3.4.4 Hospital Table	32
3.4.5 Officer Table	32
3.4.6 Emergency info Table	33
3.4.7 Feedback Table	33
3.4.8 Location Table	33

	MED EVAC
3.4.9 Message Table	34
3.4.10 Patient Table	34
<b>3.5. Screen shoots</b>	<b>35</b>
<b>3.6. Implementation</b>	<b>46</b>
3.6.1. Testing	46
3.6.2. Implementation	46
<b>4. Conclusion</b>	<b>47</b>
<b>4.1. Conclusion</b>	<b>47</b>
4.2. Future Enhancement	47
4.2. Appendix	48
4.2.1 source code	48
5 .Biblibography	61



## INTRODUCTION

In emergencies, every second counts, especially when it comes to getting medical help. But sometimes, ambulances get stuck in traffic, delaying their arrival. That's where the Ambulance Traffic Signal Detecting and Tracking System (ATSDTS) comes in. The ATSDTS is a smart system that helps ambulances get to emergencies faster. It works in two main ways: first, it detects when an ambulance is approaching a traffic light the officer who is in charge can change the light to green so the ambulance can pass without stopping. This saves time and gets the ambulance to where it's needed quickly. Secondly, the system tracks ambulances in real-time using GPS. This means that dispatchers can see where all the ambulances are on a map and send the closest one to an emergency. It also helps ambulances avoid traffic jams and roadblocks, so they can get to the scene faster. Overall, the ATSDTS is a game-changer for emergency medical services. By making ambulances faster and more efficient, it helps save lives when every second counts.

## 2. System Study and Analysis

### 2.1. Existing System

The existing traffic signal system consists of static traffic lights that operate on predefined schedules or fixed timers. These traffic signals do not have the capability to detect approaching emergency vehicles such as ambulances. As a result, ambulances may have to pass through red light, as a result there may be a chance of safety risk or it may get stuck at red lights , causing delays in reaching emergency sites. Dispatchers rely on manual communication and estimation to coordinate ambulance responses, which can lead to inefficiencies and longer response times.

### 2.2. Proposed System

**MED EVAC** aims to revolutionize emergency response by prioritizing ambulance passage at traffic signals and providing real-time tracking and monitoring capabilities. we utilizes advanced technologies such as Global Positioning System (GPS) to detect approaching ambulances and pre-empt traffic signals, ensuring unobstructed passage through intersections. Additionally, each ambulance is equipped with a GPS-enabled tracking device that transmits its location, and status to a centralized monitoring system. Dispatchers can access this information on a digital map, enabling them to make informed decisions about dispatching the closest available ambulance to the emergency location and optimizing ambulance routes to avoid traffic congestion and roadblocks. Overall, the **MED EVAC** enhances the efficiency of emergency response operations, reduces response times, and improves patient outcomes.

## **2.3 Module Description**

### **2.3.1. Admin**

The admin can login, add and manage traffic officers, add and delete private ambulances and can track their live location if they are active, admin is the one who add all the hospitals and managing them also admin can view user feedbacks and admin is responsible for maintaining the overall functionality and performance of the med evac app

### **2.3.2. Hospital**

Hospital is also playing a major role like admin, hospital can login, add and manage their ambulance and they can communicate with each other with chat messaging,

### **2.3.3. Officer**

Officer can login, view nearest hospital and he will notify when an ambulance is near his territory to change the red light into green

### **2.3.4. driver**

Ambulance driver can login, send emergency info and patient info to the nearest hospital

### **2.3.5. User**

User have to register and login, user can view nearest ambulance and hospitals and in case of any emergency user can call the ambulance that is near him and also user can send feedbacks

## **2.4. Feasibility Study**

Feasibility is a test of system according to workability, impact on organization ability to meet user needs, and effective use of resources. An estimate is made whether the identified user may be satisfied using current software and hardware technologies. The effective from the business point of view and if it can be developed in the given existing budgetary constraints. Generally, feasibility studies precede technical development and project implementation. Feasibility study produce the overall details about a project, their importance in different fields. The three essential aspects of are involved in feasibility study promotions of the preliminary investigations technical, economic and operational feasibility.

### **2.4.1. Operational Feasibility**

Operational feasibility examine how a project plan satisfies the requirements identified in the requirements analysis phase of system development. It helps in taking advantage of the opportunities and fulfills the requirements as identified during the development of the project. The purpose of the operational feasibility study is to determine whether the new system will be used if it is developed and implemented from users that will undermine the possible application benefits.

### **2.4.2. Technical Feasibility**

Technical feasibility study deals with the hardware and software and technology which are required to accomplish the user requirements in the system with in the allocated time and budget. This involves questions such as whether the technology needed for the system exists, how difficult it will be to build, and whether the firm has enough experience using that technology. It is concerned with the existing computer system and to what extent it can support the proposed system. The proposed system requires python and android platform only which are open source. Due to open source of languages it may not become difficult to maintain and developing of this system. The system can also be easily upgraded to the higher level with less effort and maintenance. This application is easily used with their smartphone from anywhere and user friendly. Hence the proposed work is technical

feasible. User can easily access and book the court by their smart phones, no need of any other heavy requirements.

### **2.4.3. Economic Feasibility**

This assessment typically involves a cost/ benefits analysis of the project, helping organizations determine the viability, cost, and benefits associated with a project before financial resources are allocated. In the fast paced world today there is a great need of online social networking facilities. Thus the benefits of this project in the current scenario make it economically feasible. The purpose of the economic feasibility assessment is to determine the positive economic benefits to the organization that the proposed system will provide. This involves questions such as whether the firm can afford to build the system, whether its benefits should substantially exceed its costs, and whether the project has higher priority and profits than other projects that might use the same resources. This also includes whether the project is in the condition to fulfil all the eligibility criteria and the responsibility of both sides in case there are two parties involved in performing any project. This study presents tangible and intangible benefits from the project by comparing the developments and operational costs. The technique of cost-benefit analysis is often used as a basis for assessing economic feasibility.

## **2.5. System environment**

### **2.5.1. Developer Requirement**

#### **2.5.1.1 Hardware requirement**

- Processor : Intel Core i3 and above
- RAM : 8GB
- Storage : 500GB Hard disk

#### **2.5.1.2 Software requirement**

- Operating system : Windows 10
- Front end : Python, Java
- Back end : Mysql
- IDE : Android studio, JetBrains Pycharm
- Web browser : Internet Explorer/Google chrome/Firefox

### **2.5.2. User requirement**

- An android smartphone : Version 7.0 and above
- Any Computer or laptop.

## **3 METHODOLOGY**

### **3.1. Introduction**

Agile methodology is followed in this project. Agile software development comprises various approaches to software development under which requirements and solutions evolve through the collaborative effort of self-organizing and cross-sectional teams and their customers/end users. It advocates adaptive planning, evolutionary development, early delivery and continuous improvement and it encourage rapid and flexible response to change.

### 3.2. Data flow diagram (DFD)

A data flow diagram (DFD) or a bubble chart is a graphical tool for structured analysis. DFD models a system by using external entities from which data flow to a process, which transforms the data and creates output data flows which go other process or external entities or files. Data in files may also flow to processes as inputs.

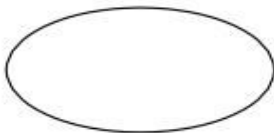
DFDs can be hierarchically organized, which help in partitioning and analyzing large systems. As a first step, one dataflow diagram can depict an entire system which gives the system overview. It is called context diagram of level0 DFD. The context diagram can be further expanded.

#### DFD Design Notation

In DFD, there are four main symbols:



Source or Destination of Data



Process



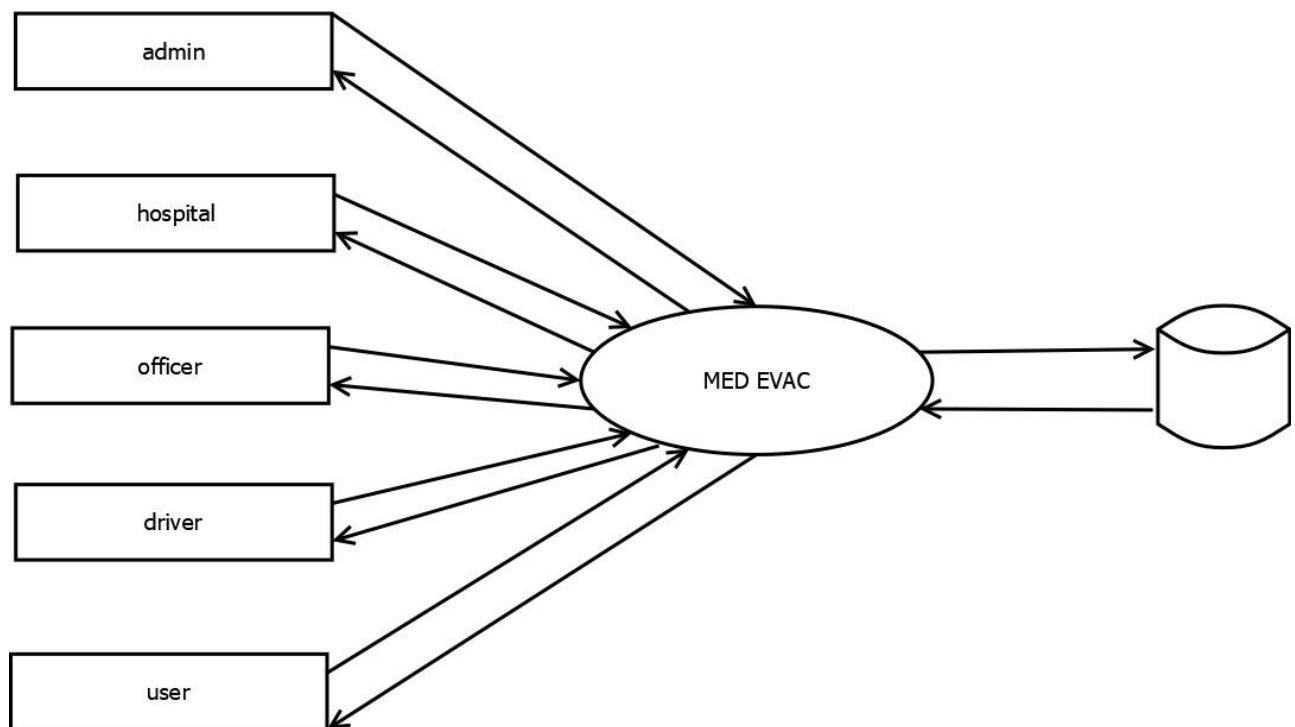
Flow of Data



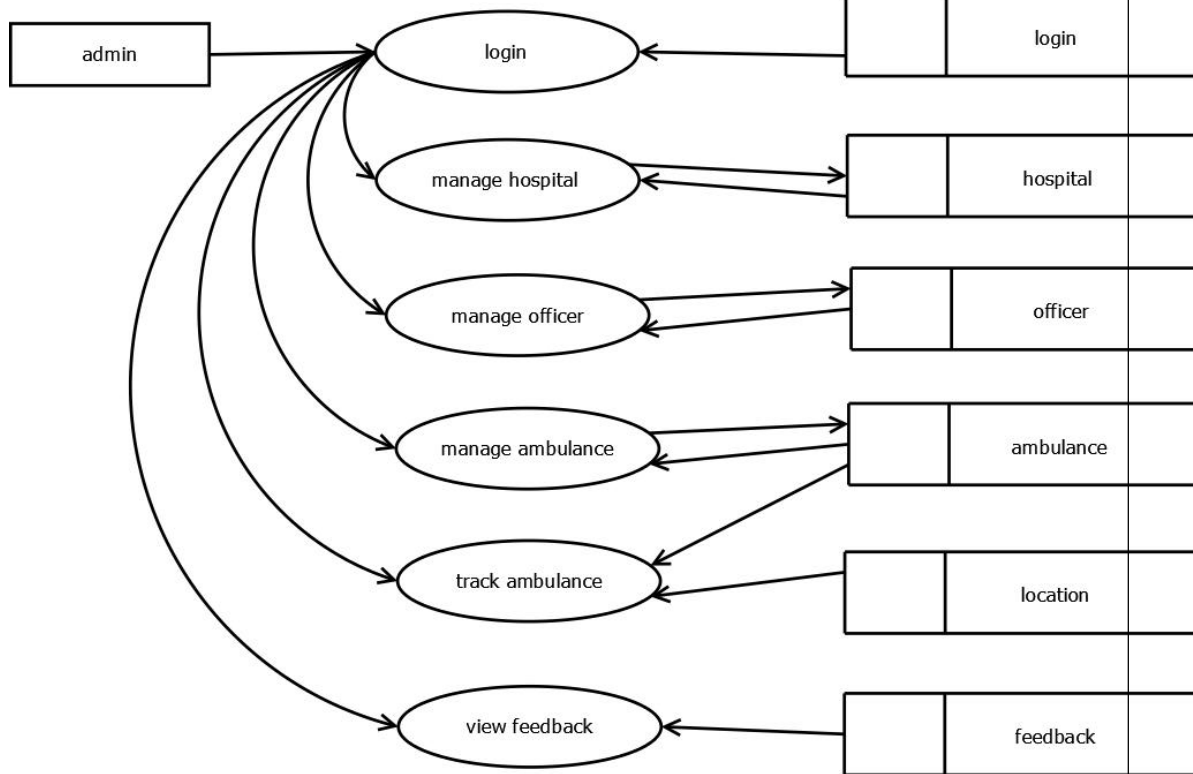
Temporary Repository of Data



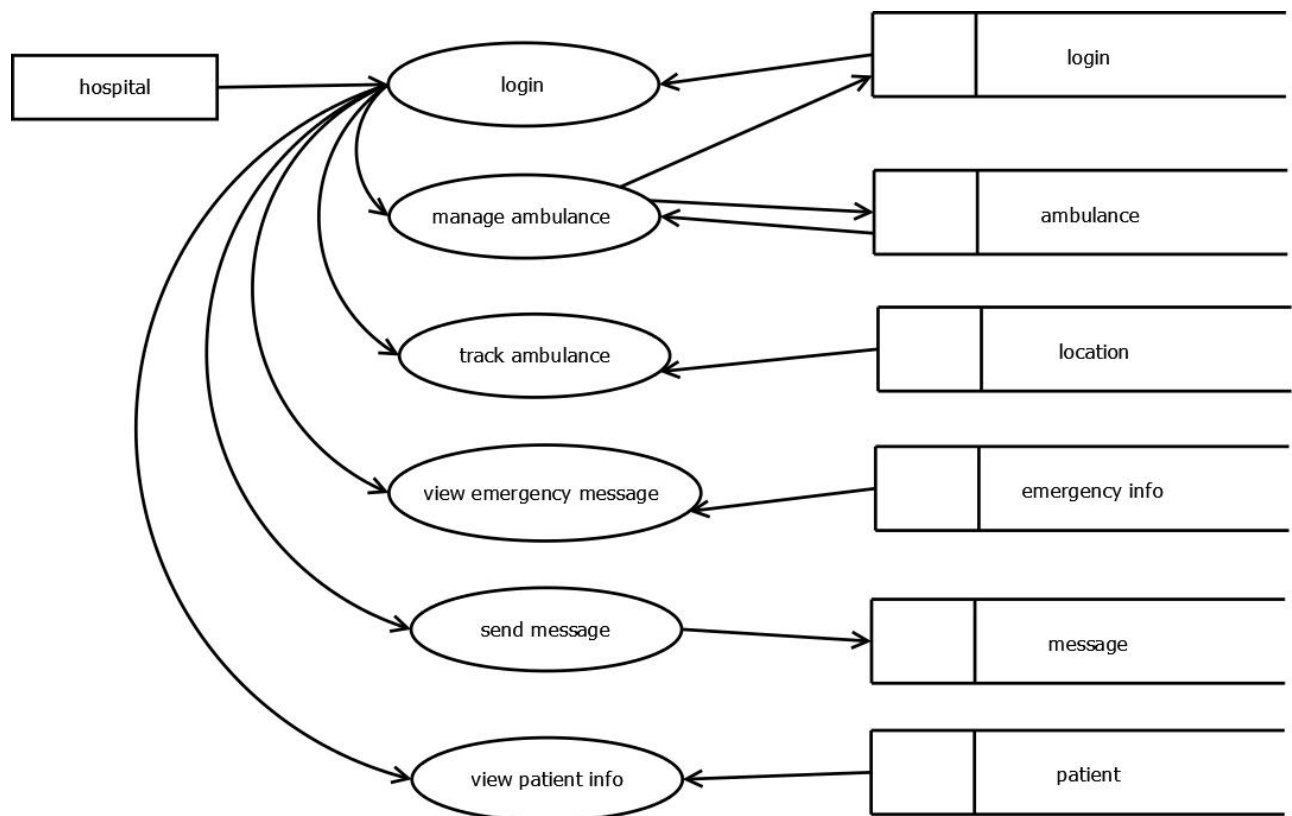
## Level 0



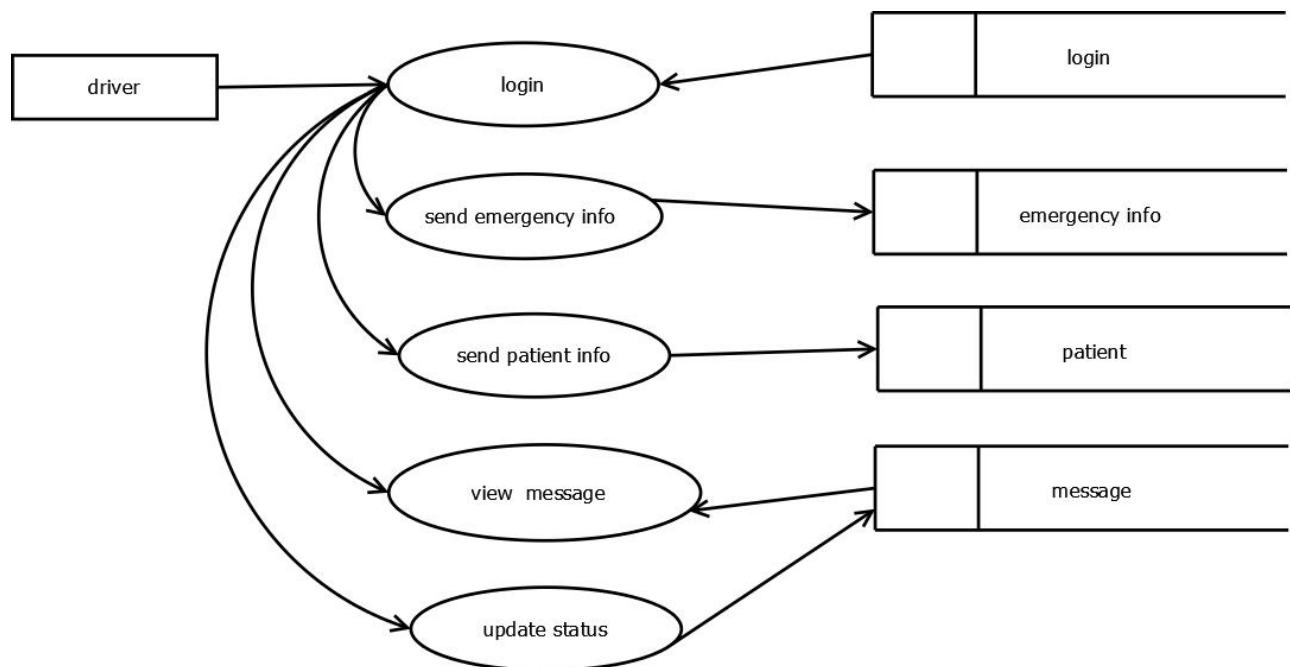
## Level 1.1



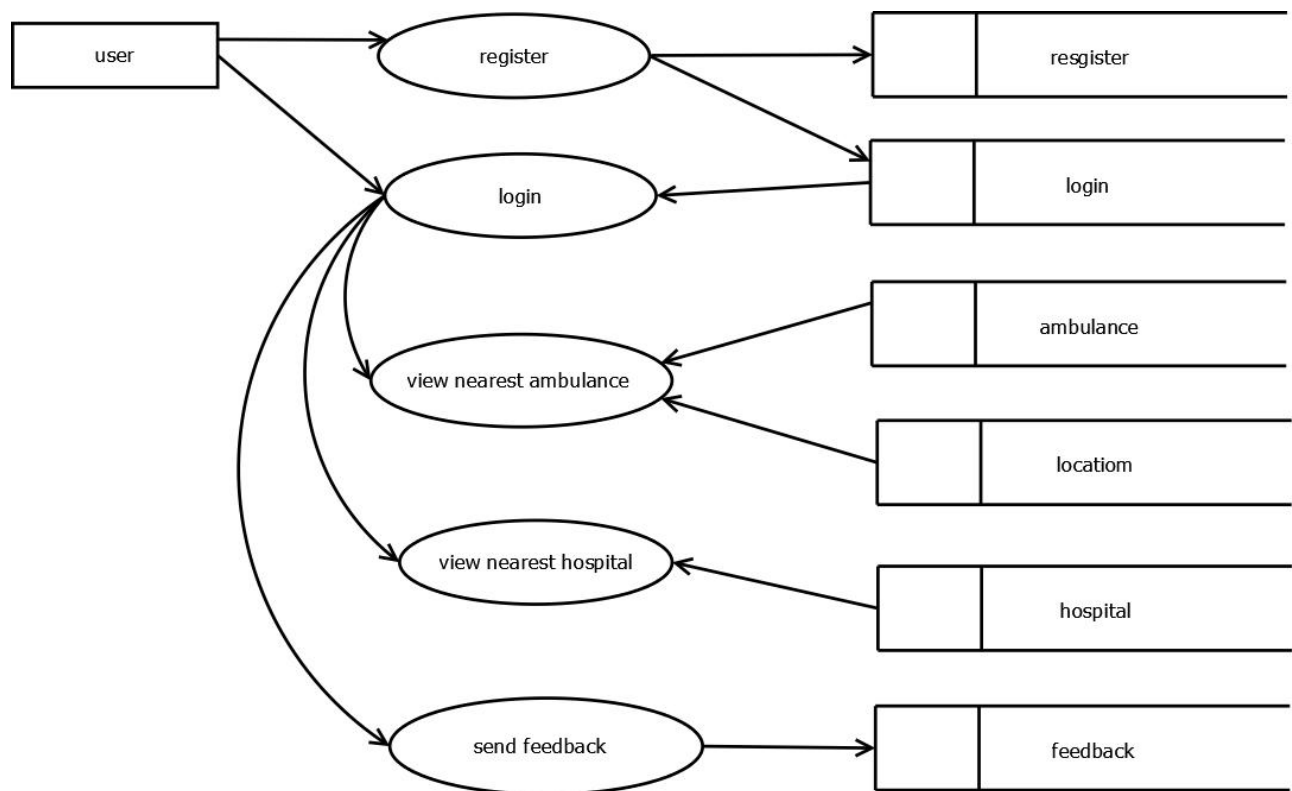
## Level 1.2



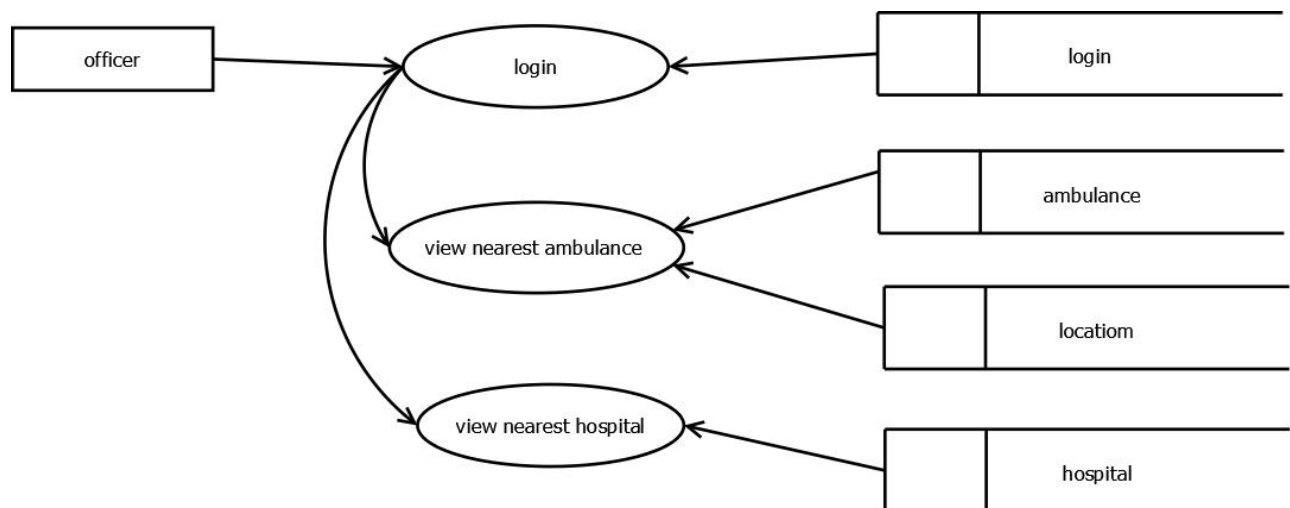
## Level 1.3



## Level 1.4

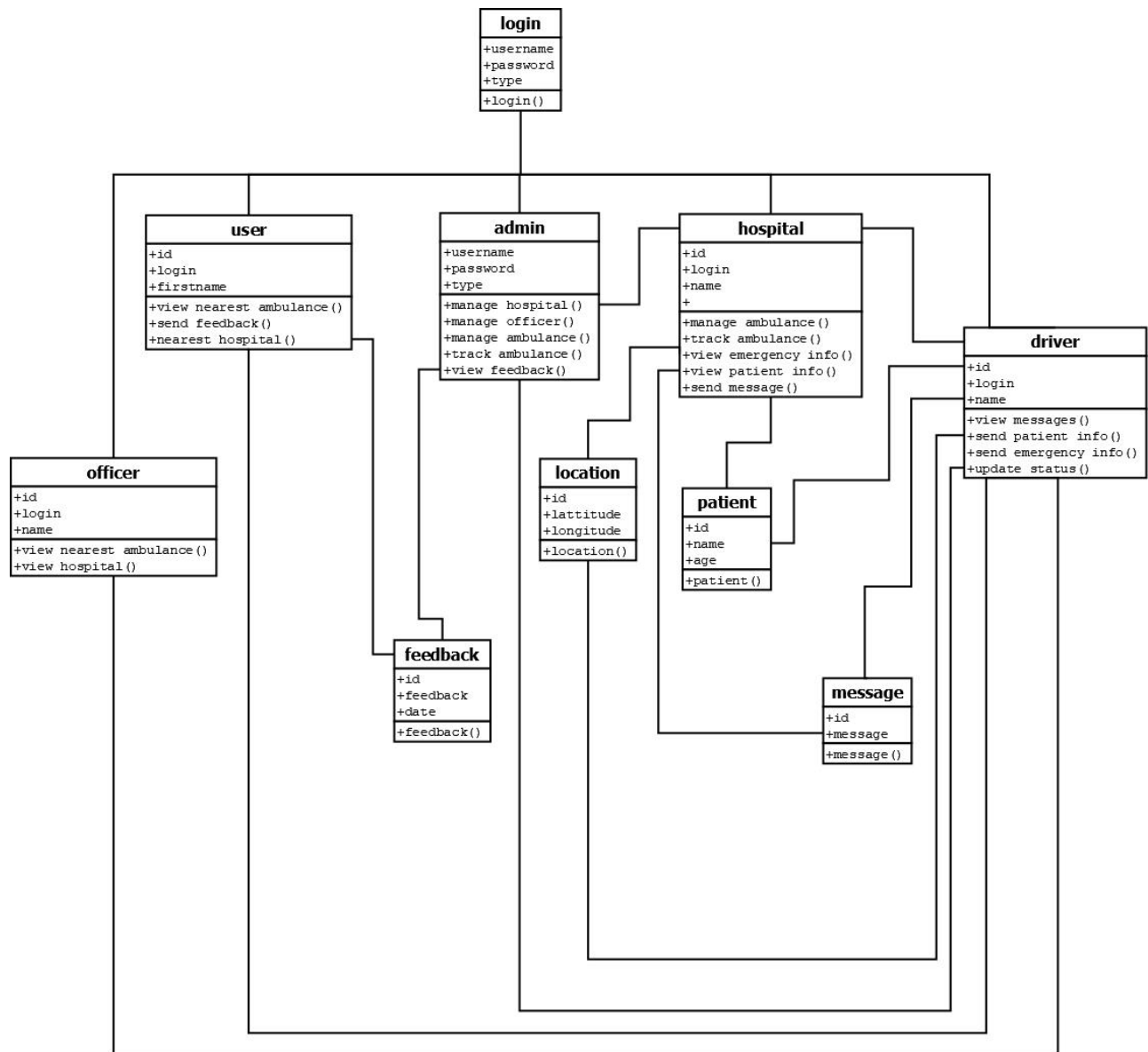


## Level 1.5



## 3.3. UML Diagrams

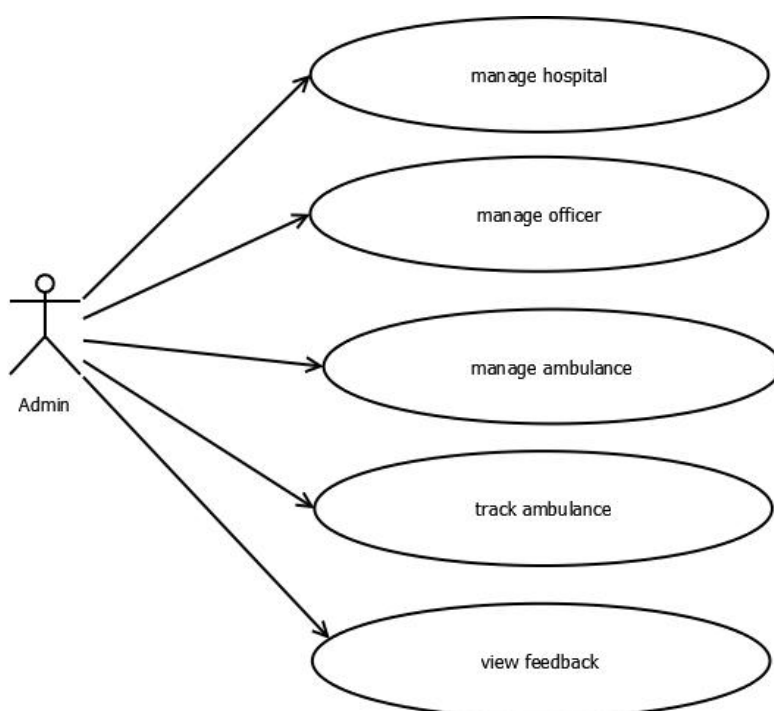
### 3.3.1 Class Diagram



### 3.3.2 Use case Diagram

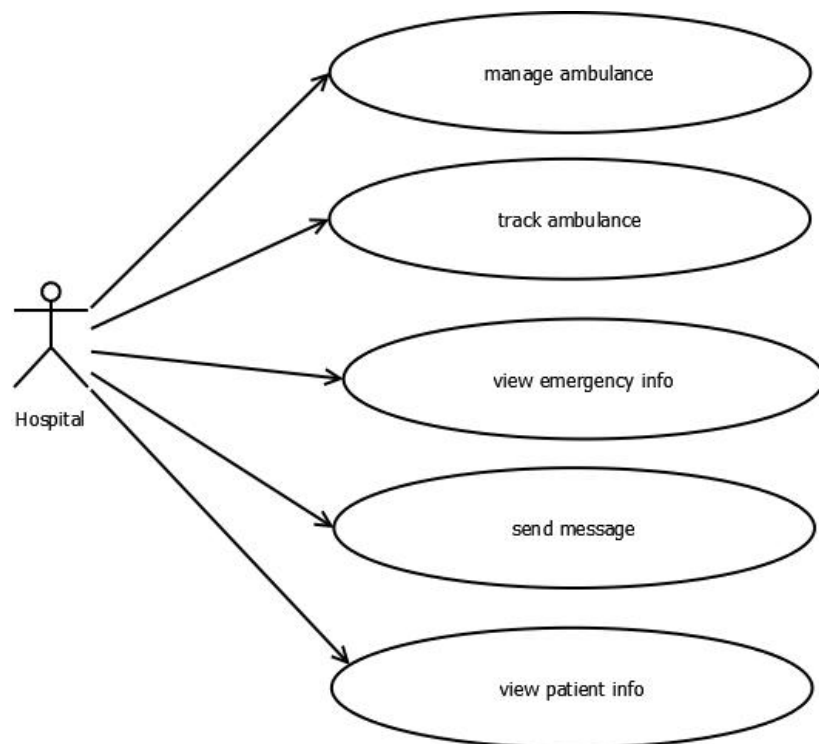
A use case diagram at its simplest is a representation of a user's interaction with the system that shows the relationship between the user and the different use cases in which the user is involved. A use case diagram can identify the different types of users of a system and the different use cases and will often be accompanied by other types of diagrams as well. The use cases are represented by either circles or ellipses. While a use case itself might drill into a lot of detail about every possibility, a use-case diagram can help provide a higher-level view of the system. It has been said before that "Use case diagrams are the blueprints for your system". They provide the simplified and graphical representation of what the system must actually do due to their simplistic nature, use case diagrams can be a good communication tool for stakeholders. The drawings attempt to mimic the real world and provide a view for the stakeholder to understand how the system is going to be designed. Siau and Lee conducted research to determine if there was a valid situation for use case diagrams at all or if they were unnecessary. What was found was that the use case diagrams conveyed the intent of the system in a more simplified manner to stakeholders and that they were "interpreted more completely than class diagrams". The purpose of the use case diagrams is simply to provide the high level view of the system and convey the requirements in laypeople's terms for the stakeholders. Additional diagrams and documentation can be used to provide a complete functional and technical view of the system.

#### ADMIN

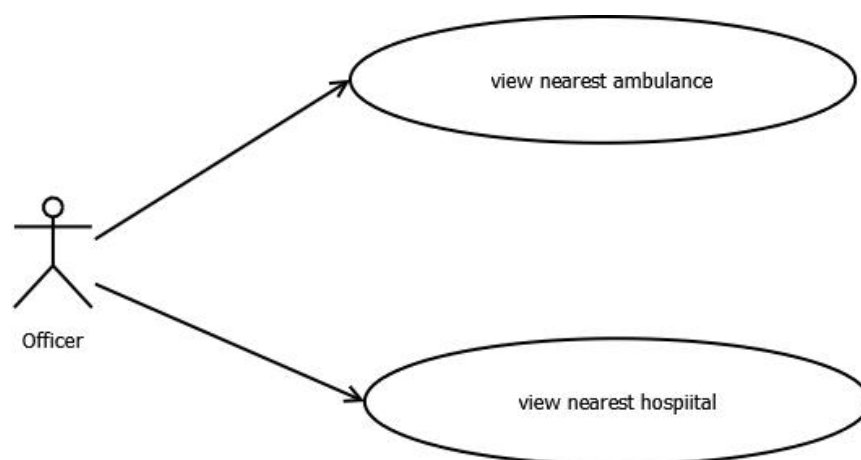


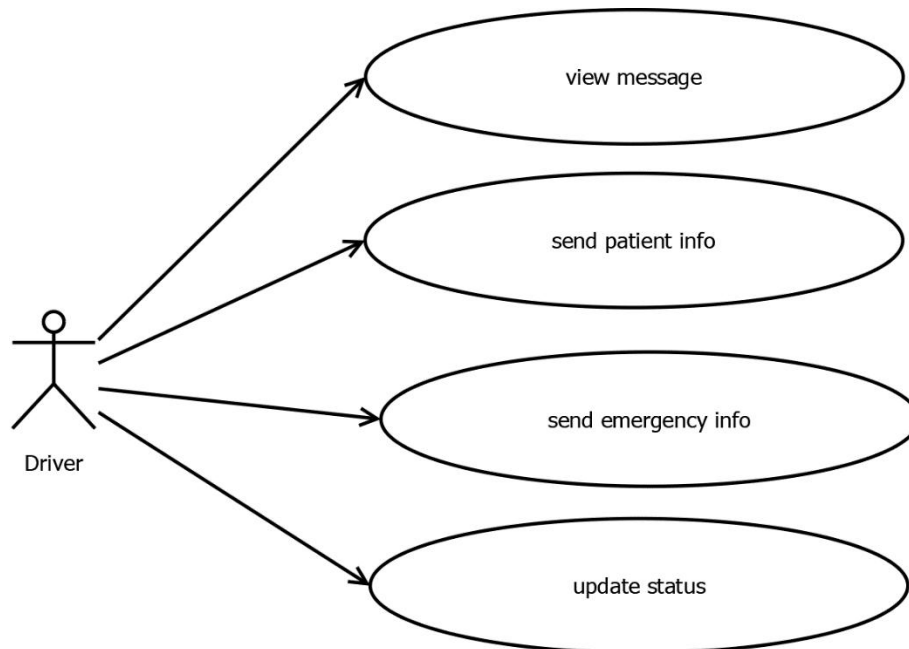
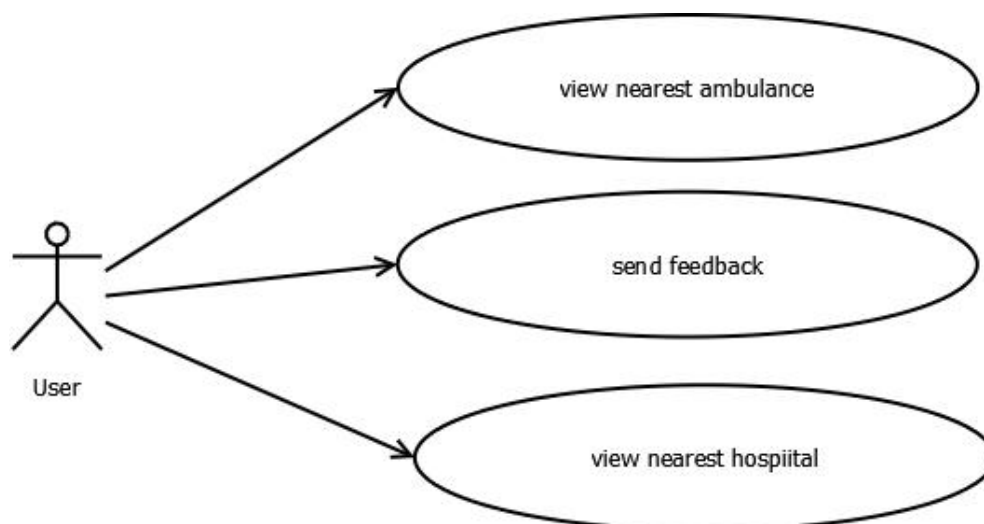


## HOSPITAL



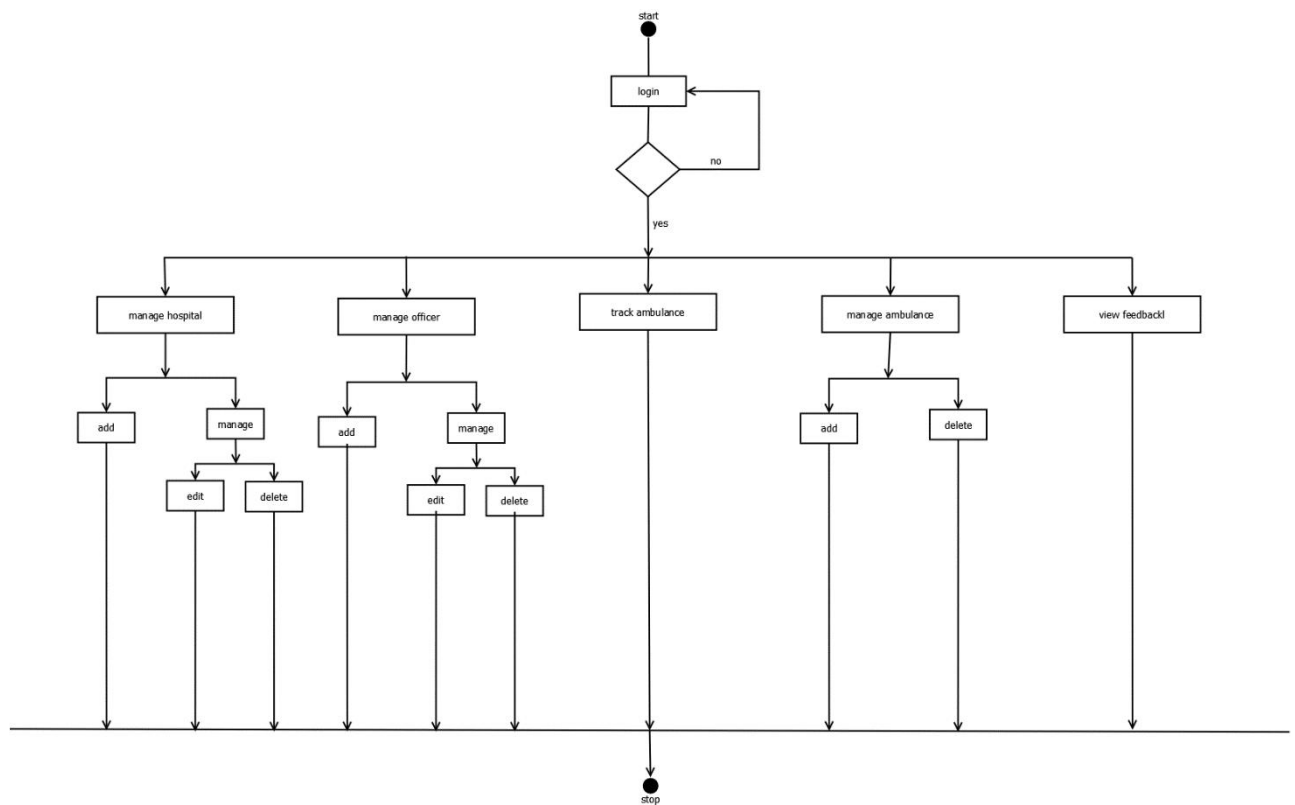
## OFFICER



**DRIVER****USER**

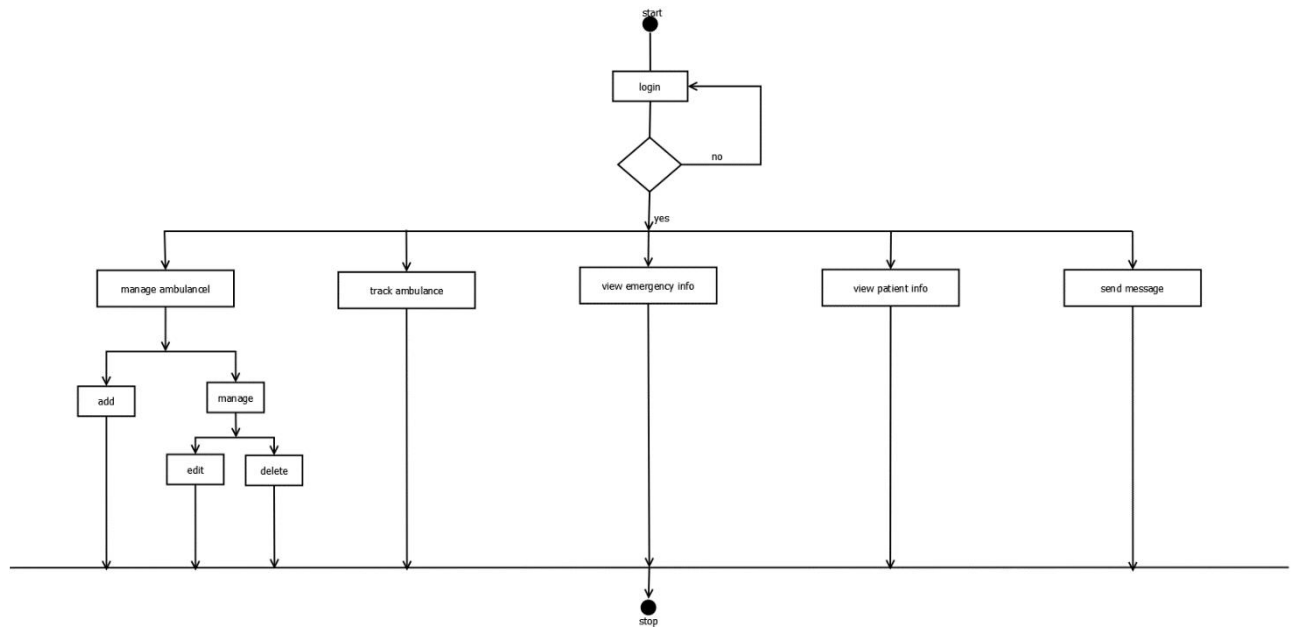
### 3.3.3 Activity Diagram

#### ADMIN ACTIVITY

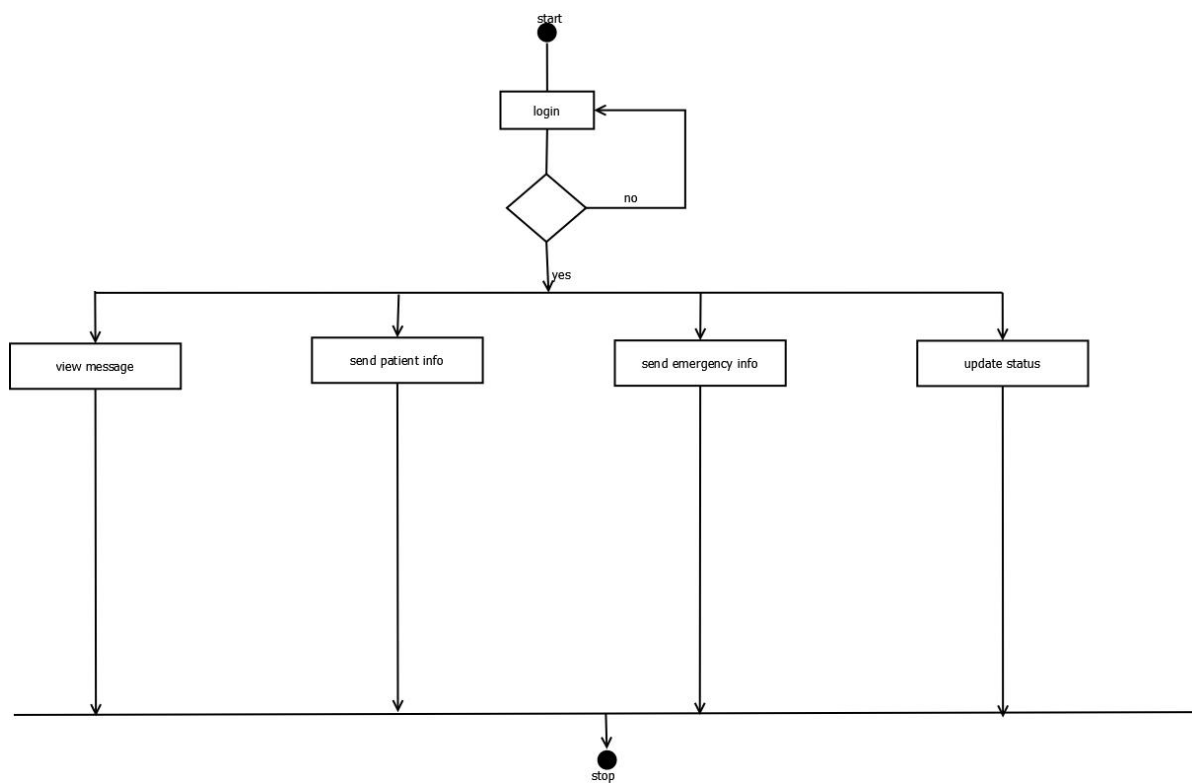


#### HOSPITAL ACTIVITY

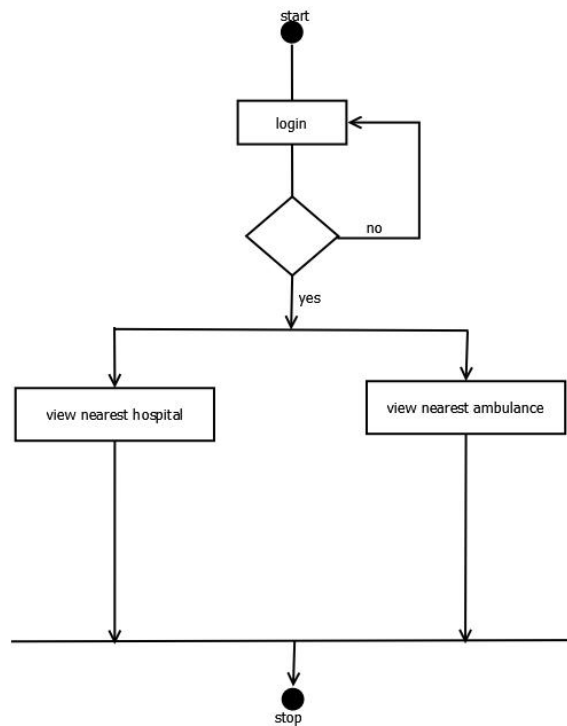
## MED EVAC



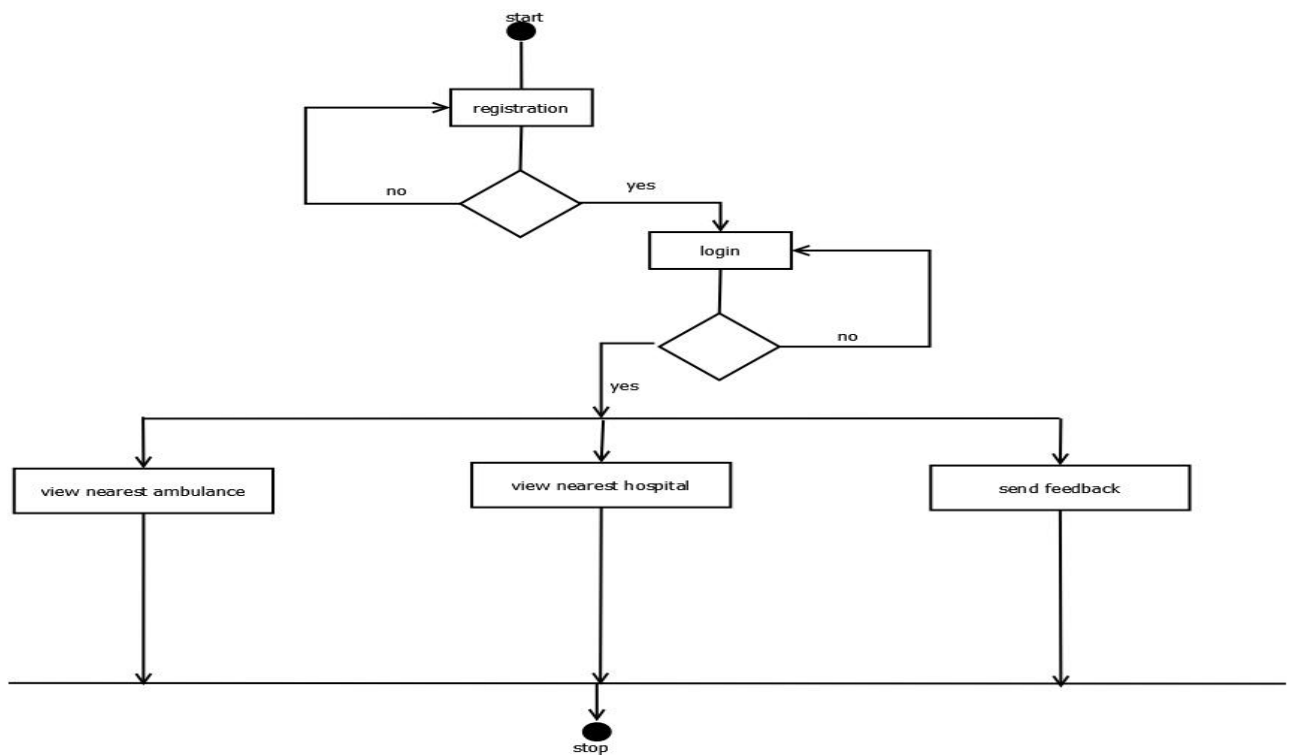
## DRIVER ACTIVITY



## OFFICER ACTIVITY



## USER ACTIVITY



## 3.4 Table Design

### 3.4.1 Login Table

this login table include User id, Username, User Type and password for login to this application. admin, hospital, officer, driver and user enter to their homepage by entering username and password and access his/her account

Attribute	Data Type	Constraint	Description
Id	BIGINT(20)	PRIMARY KEY	Login id for reference
Username	VARCHAR(50)	NOT NULL	Name of user
Password	VARCHAR(50)	NOT NULL	Password
User type	VARCHAR(50)	NOT NULL	Which type of user is login

### 3.4.2 User Table

User Table include name, place, post, pin, email, phone, user can login to his home page by entering username and password

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	User id
Name	VARCHAR(20)	NOT NULL	Name
Place	VARCHAR(20)	NOT NULL	Place
Post	VARCHAR(20)	NOT NULL	Post
Pin	BIGINT(11)	NOT NULL	Pin code
Email	VARCHAR(30)	NOT NULL	Email
Phone	BIGINT(20)	NOT NULL	Phone number
Login Id	BIGINT(20)	NOT NULL	Login id

### 3.4.3 Ambulance Table

Ambulance Table include id, name, place, post, pin, email, phone, status, driver name, photo, proof, login id and type id

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Ambulance id
name	VARCHAR(50)	NOT NULL	Ambulance name
Place	VARCHAR(50)	NOT NULL	Place
Post	VARCHAR(50)	NOT NULL	Post office
Pin	BIGINT(20)	NOT NULL	Pin number
Email	VARCHAR(50)	NOT NULL	Email
Phone	BIGINT (20)	NOT NULL	Phone number
Status	VARCHAR (20)	NOT NULL	Ambulance is active or not
Driver Name	VARCHAR(50)	NOT NULL	Name of the driver
Photo	VARCHAR(100)	NOT NULL	Photo of the driver
Proof	VARCHAR(100)	NOT NULL	Id proof of the driver
Login id	BIGINT(20)	NOT NULL	Login id
Type id	BIGINT(20)	NOT NULL	Type id

### 3.4.4 Hospital Table

Hospital Table include id, name, place, post, pin, Email, phone, location, image and login id

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Hospital id
Name	VARCHAR(50)	NOT NULL	Hospital Name
Place	VARCHAR(50)	NOT NULL	Place
Post	VARCHAR(50)	NOT NULL	Post office
Pin	BIGINT(20)	NOT NULL	Pincode
Email	VARCHAR(100)	NOT NULL	Email id
Phone	BIGINT(20)	NOT NULL	Phone number
Latitude	DOUBLE	NOT NULL	latitude
Longitude	DOUBLE	NOT NULL	longitude
Image	VARCHAR(100)	NOT NULL	Image of the hospital
Login id	BIGINT(20)	NOT NULL	Login id

### 3.4.5 Officer Table

Officer Table include id, name of the officer, place and address of the officer and contact details such as phone email and the designation of the officer and also an image of the officer

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Officer id
Name	VARCHAR(50)	NOT NULL	Officer Name
Place	VARCHAR(50)	NOT NULL	Place
Post	VARCHAR(50)	NOT NULL	Post office
Pin	BIGINT(20)	NOT NULL	Pincode
Phone	BIGINT(20)	NOT NULL	Phone number
Email	VARCHAR(100)	NOT NULL	Email id
Designation	VARCHAR(50)	NOT NULL	Designation of the officer
Login id	BIGINT(20)	NOT NULL	Login id
Photo	VARCHAR(100)	NOT NULL	Image of the Officer



### 3.4.6 Emergency info Table

Emergency info Table include id, emergency info, date and the id of ambulance.

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	id
Info	VARCHAR(2000)	NOT NULL	Emergency info
Date	DATE	NOT NULL	Date
Ambulance id	BIGINT(20)	NOT NULL	Ambulance id

### 3.4.7 Feedback Table

Feedback Table include id, feedback written by the user with date and user id

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	id
Feedback	VARCHAR(200)	NOT NULL	Feedback of user
Date	DATE	NOT NULL	Date
User id	BIGINT(20)	NOT NULL	user id

### 3.4.8 Location Table

Location table includes id, location of the ambulances and ambulance id

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Location id
Latitude	DOUBLE	NOT NULL	Latitude
Longitude	DOUBLE	NOT NULL	Longitude
Ambulance id	BIGINT(20)	NOT NULL	Ambulance id

### 3.4.9 Message Table

Message Table include message id, message send by the hospital with date and hospital id and ambulance id

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Message id
Message	VARCHAR(2000)	NOT NULL	Message
Date	DATE	NOT NULL	Date
Ambulance id	BIGINT(20)	NOT NULL	Ambulance id
Hospital id	BIGINT(20)	NOT NULL	Hospital id

### 3.4.10 Patient Table

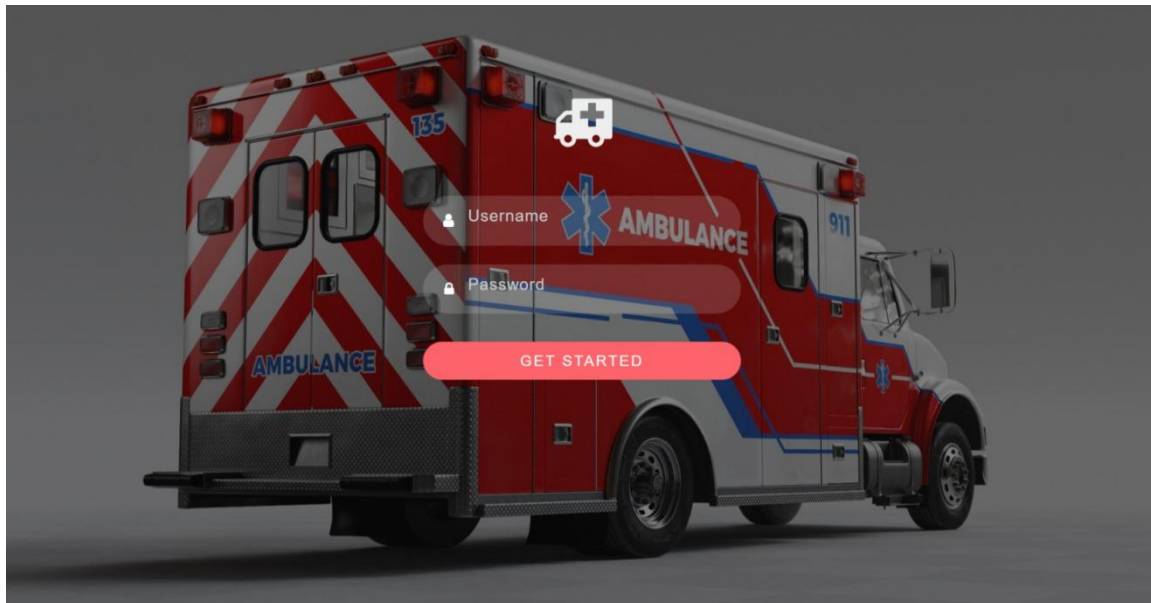
Patient Table include patient id, patient name, place and case with date and ambulance id and hospital id.

Attribute	Datatype	Constraint	Description
Id	BIGINT(20)	NOT NULL	Patient id
Name	VARCHAR(50)	NOT NULL	Name of patient
Case	VARCHAR(100)	NOT NULL	Case
Place	VARCHAR(50)	NOT NULL	Place of the patient
Date	DATE	NOT NULL	Date
Ambulance id	BIGINT(20)	NOT NULL	Ambulance id
Hospital id	BIGINT(20)	NOT NULL	Hospital id

## 3.5 Screen shoots

### Login page

By using Login page, Admin and Hospitals can login to homepage through web application. They can enter username and password to enter their home page.



### Admin Home page


Admin can view his homepage by entering to his page. He can access and manage his account by this Web application. He can manage hospital, ambulance and officers. also he can view user feedbacks . and he can track all the active ambulances



## Manage hospital

Admin can add, edit and delete hospital and can track them

Hospital


Name	Address	Email	phone	Location	
 Baby Memorial	Calicut Calicut 673579	babymemorial@gmail.com	9654328612	<a href="#">Track</a>	<input type="button" value="Edit"/> <input type="button" value="Delete"/>
					<input type="button" value="Add"/>

## Add hospital

Name	<input type="text"/>
Photo	<input type="button" value="Choose File"/> No file chosen
Place	<input type="text"/>
Post	<input type="text"/>
Pin	<input type="text"/>
Email	<input type="text"/>
Phone	<input type="text"/>
Latitude	<input type="text"/>
Longitude	<input type="text"/>
Username	<input type="text"/>
Password	<input type="text"/>
<input type="button" value="Add"/>	


## Manage ambulance

Admin can add and delete private ambulance

		Ambulance	<input type="text"/>	Search			
Name	Driver	Address	Proof	Email	Phone	Status	
 Amb1	Arun	Calicut Calicut 654321	<a href="#">Download</a>	amb1@gmail.com	9876543212	<a href="#">Delete</a>	
		<a href="#">Add</a>					

## Manage Officer

Admin can add, edit and delete Officer

		Name	<input type="text"/>	search			
SINo	Name	Address	Phone	Email	Designation		
1	 Ajith	Calicut Westhill 654321	9876543213	ajith@gmail.com	Traffic	<a href="#">Edit</a>	<a href="#">Delete</a>
		<a href="#">Add</a>					

## Track ambulance

Admin can track ambulance in Realtime if its active

		Ambulance	<input type="text"/>	Search		
Ambulance		Location		Status		
Amb1		<a href="#">Track</a>		pending		

## View feedbacks

Admin can view feedback written by users and view them and sort them according to date

Date	<input type="text" value="dd-mm-yyyy"/>	<input type="button" value="Search"/>
Feedback	User	Date
Helpful	Aswin	Jan. 18, 2024

## Hospital home

Hospital can add and manage their ambulances and they can track them, also hospital can view the emergency info and patient info that is send by the ambulance and can replay with essential directions or message with chatting facility.



## Emergency info

Hospital can view the emergency info send by the ambulance

Ambulance <input type="text"/>			Search
Ambulance	Info	Date	
Amb2	Reach in 5 mins	Jan. 18, 2024	
Amb2	Need 3 Stretchers	Jan. 19, 2024	
Amb2	huuuu	March 7, 2024	

## Send message


Hospital can replay with messages when ambulance send an emergency info and can sort by ambulance name.

Ambulance		<input type="text"/>	<button>Search</button>
Ambulance	Message	Date	
Amb2	Straight to casualty...	Jan. 18, 2024, 3:35 p.m.	<button>Delete</button>
Amb2	Enter through Second gate	Jan. 19, 2024, 3:37 p.m.	<button>Delete</button>
Amb2	Inject 1 dose of insuline	Jan. 19, 2024, 3:38 p.m.	<button>Delete</button>
<div><button>Send New</button></div>			

Ambulance	<input type="text" value="Amb2"/>
Message	<input type="text"/>
Send	

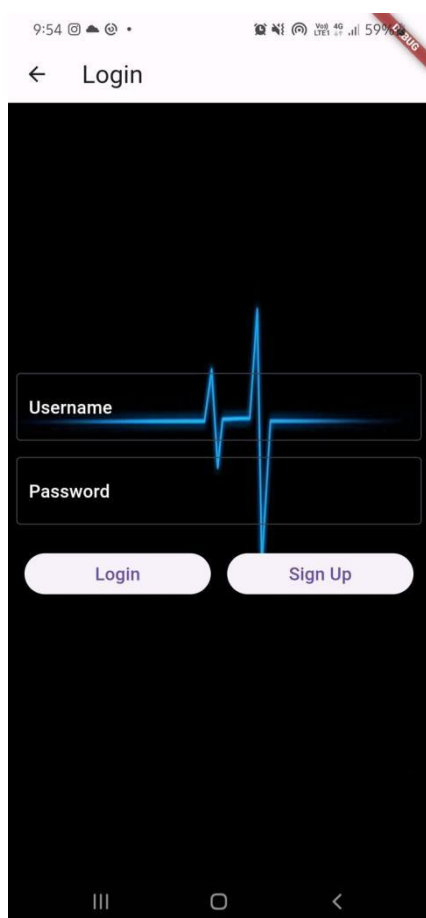
## View patient info

Hospital can view patient info send by ambulance while ambulance attended an emergency

Date <input type="text" value="dd-mm-yyyy"/> 				<input type="button" value="Search"/>
Ambulance	Patient Name	Case	Place	
Amb2	Ram	Accident	Calicut	
Amb2	Aswin	Nothing	Calicut	
Amb2	yy	uu	hh	
Amb2	ram	accident	calicut	
Amb2	hari	accident	clt	

## Login page

Android app login page for user, driver and officer





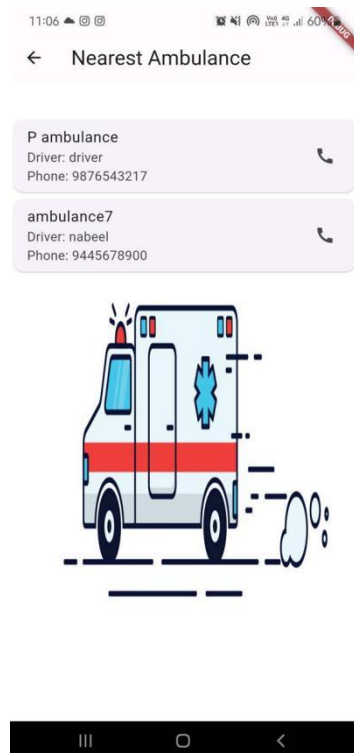
## Nearest hospital

The user and the officer can view nearest hospital in the app



## Nearest ambulance

User and officer can view nearest ambulance around them



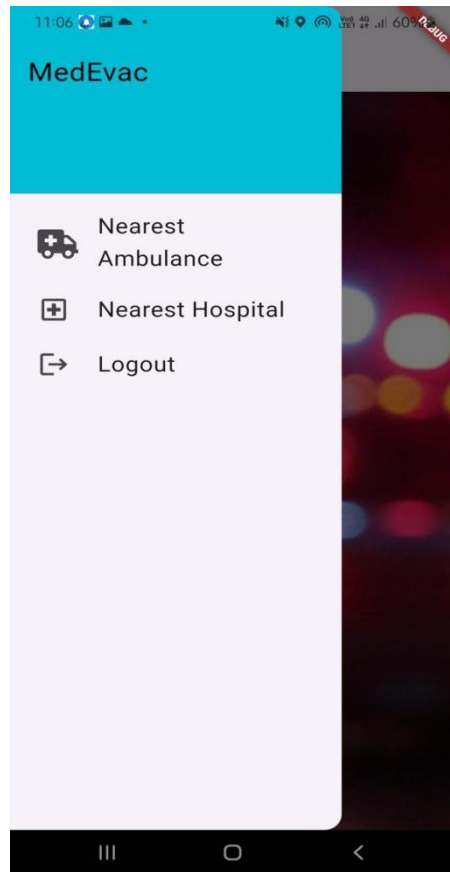
## Feedback

User can send feedback directly to admin



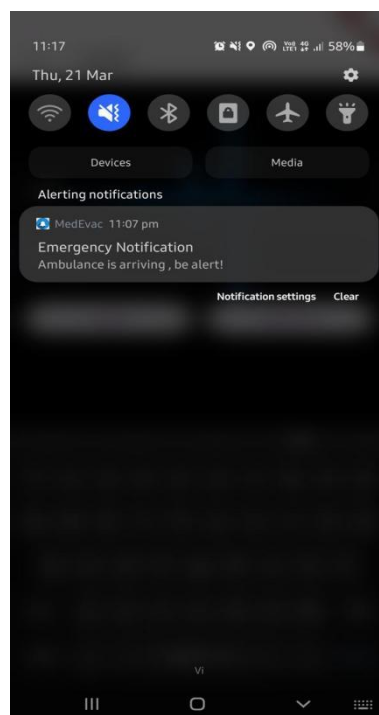
## Officer dash

Officer can view nearest ambulance and nearest hospital



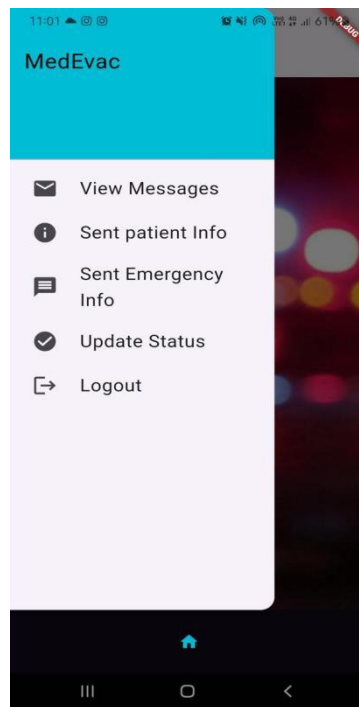
## Officer notification

Officer will get a notification when any ambulance is around him so he could change the red signal to green.



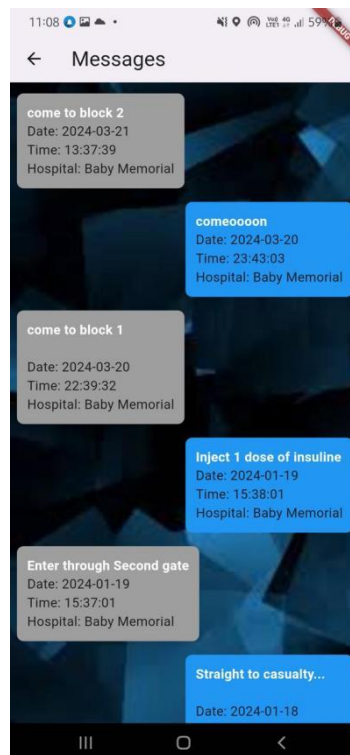
## Driver dash

Driver can view messages, send patient info, send emergency info and can update his status whether he is active or no non active



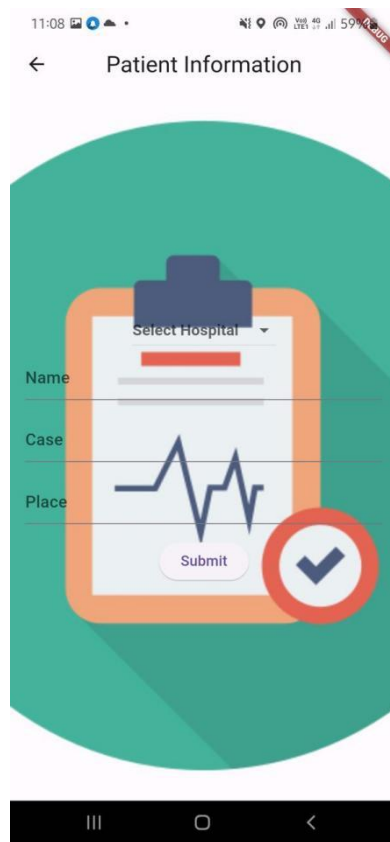
## View messages

Driver can view messages send by hospital



## Send patient info

Driver can send patient information



The screenshot shows a mobile application interface titled "Patient Information". At the top, there is a status bar with the time 11:08, signal strength, and battery level at 59%. Below the title bar, there is a large green rounded rectangle containing a white clipboard icon. The clipboard has a "Select Hospital" dropdown menu, a "Name" input field, a "Case" input field, and a "Place" input field. A blue ECG line is drawn across the clipboard. At the bottom of the clipboard is a "Submit" button. To the right of the clipboard is a red circular button with a white checkmark. The bottom of the screen shows the standard Android navigation bar with three icons: a square, a circle, and a triangle.

## Send emergency info

Driver can send emergency information to the hospital

11:08 5G 59%

← Emergency Information

Emergency Information

Submit

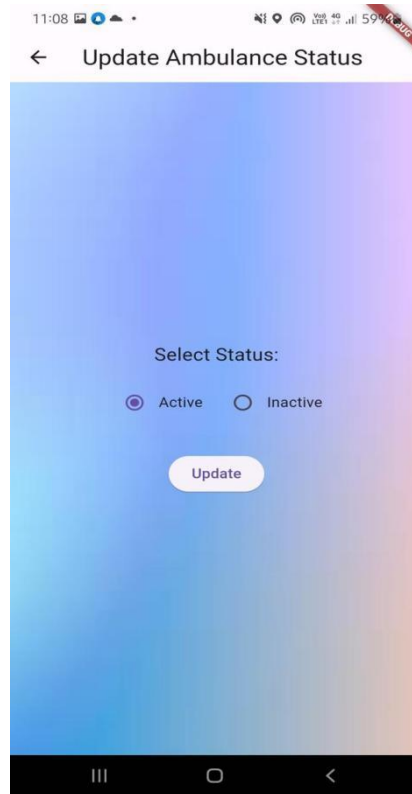
IN CASE OF EMERGENCY

READYTOGOEMERGENCY.COM

PAGE 2

## Update status

Driver can update status



## 3.6 Testing and Implementation

### 3.6.1 Testing

Testing is an important step in the software engineering process that could view rather than constructive. Testing is the process of executing a program with the intent of finding an error . a good test is that has probability to find an as yet undiscovered error.

- A good case is one that has a high probability of finding an unpredictable error.
- A successful case is one that has a high probability of finding an unpredictable error.
- A good test case is one that provides solution to that unpredictable error
- A test plan entailed the following activities. We prepare list plan.
- We specified condition for users acceptance testing.
- We prepared list data for program testing .
- Also we prepared list data transaction plan testing.
- Then we planned user training.
- Our programs were compiled and assembled.

### 3.5.2 Implementation

Implementing MED EVAC involves several key steps. Firstly, a comprehensive system architecture must be designed, outlining the components and interfaces

required for effective operation. Necessary hardware components, such as GPS-enabled tracking devices for ambulances and traffic signal pre-emption equipment, must be procured, along with corresponding software systems. Integration of these hardware and software components is crucial, requiring rigorous testing to ensure seamless communication and functionality. Once tested, the project can be deployed at selected intersections, with traffic signal pre-emption equipment installed and GPS tracking devices configured in ambulances. Training sessions should be conducted for traffic signal operators to familiarize them with the system's operation and protocols. Continuous monitoring of system performance and regular maintenance are essential to ensure ongoing functionality. Feedback from users and performance metrics should be gathered to evaluate the system's impact and identify areas for optimization. As the system proves successful, it can be scaled up for broader deployment and potential integration with other emergency response systems, contributing to improved efficiency and effectiveness of emergency medical services.



# Chapter 4

## 4.1 Conclusion

In conclusion, the MED EVAC represents a significant advancement in emergency response technology, with the potential to greatly enhance the efficiency and effectiveness of ambulance services. By prioritizing ambulance passage at traffic signals and providing real-time tracking and monitoring capabilities, MED EVAC aims to reduce response times, optimize resource allocation, and ultimately save more lives during critical medical situations. Through careful implementation, monitoring, and optimization, this innovative system can revolutionize the way emergency medical services operate, leading to improved patient outcomes and bolstering the resilience of healthcare systems. As the MED EVAC continues to evolve and expand, its impact on emergency response operations will become increasingly profound, shaping a safer and more responsive healthcare infrastructure for communities worldwide.

## 4.2 Future Enhancement

Looking ahead, there are several exciting avenues for future enhancements to the MED EVAC that can further elevate its capabilities and impact on emergency response operations. One potential enhancement could involve leveraging emerging technologies such as artificial intelligence (AI) and machine learning (ML) algorithms to improve the predictive capabilities of the system. By analysing historical data on traffic patterns, emergency incidents, and ambulance movements, AI-powered algorithms could anticipate potential traffic congestion hotspots and dynamically adjust ambulance routes in real-time to minimize delays. Additionally, integrating the ATSDTS with smart city infrastructure and vehicle-to-infrastructure (V2I) communication systems could enable even greater coordination between ambulances and traffic signals. This could include the implementation of vehicle-to-everything (V2X) communication protocols, allowing ambulances to communicate directly with traffic signals to request priority passage when approaching intersections. Furthermore, advancements in sensor technology and connectivity could enable the development of more sophisticated detection systems capable of identifying not only ambulances but also other emergency vehicles such as fire trucks and police cars. By expanding the scope of the system to accommodate various types of emergency responders, it could enhance overall emergency management and coordination efforts. Overall, by embracing these future enhancements, the ATSDTS has the potential to further revolutionize emergency

response operations, making them even more efficient, responsive, and ultimately, life-saving.

## 4.3 Appendix

### 4.3.1 Source code

#### Login page

```
<head>
  <title> Login Form </title>

  <meta name="viewport" content="width=device-width, initial-scale=1">
  <meta charset="utf-8">
  <meta name="keywords" content="Key Login Form a Responsive Web Template, Bootstrap
Web Templates, Flat Web Templates, Android Compatible Web Template, Smartphone
Compatible Web Template, Free Webdesigns for Nokia, Samsung, LG, Sony Ericsson, Motorola
Web Design">
  <script>
    addEventListener("load", function () {
      setTimeout(hideURLbar, 0);
    }, false);

    function hideURLbar() {
      window.scrollTo(0, 1);
    }
  </script>

  <link rel="stylesheet" href="../static/css/style.css" type="text/css" media="all">

  <link rel="stylesheet" href="../static/css/font-awesome.min.css" type="text/css"
media="all">
  <link
href="../static//fonts.googleapis.com/css?family=Quattrocento+Sans:400,400i,700,700i"
rel="stylesheet">
  <link
href="../static//fonts.googleapis.com/css?family=Mukta:200,300,400,500,600,700,800"
rel="stylesheet">

</head>

<body>

<section class="main">
  <div class="layer">

    <div class="bottom-grid">
      <div class="logo">
      </div>
      <div class="links">
      </div>
    </div>
    <div class="content-w3ls">
      <div class="text-center icon">
        <span class="fa fa-ambulance"></span>
      </div>
      <div class="content-bottom">
        <form action="/loginn" method="post">
          <div class="field-group">
            <span class="fa fa-user" aria-hidden="true"></span>
            <div class="wthree-field">
              <input name="textfield1" id="text1" type="text" value=""
```

```

placeholder="Username" required>
    </div>
</div>
<div class="field-group">
    <span class="fa fa-lock" aria-hidden="true"></span>
    <div class="wthree-field">
        <input name="textfield2" id="myInput" type="Password"
placeholder="Password">
    </div>
</div>
<div class="wthree-field">
    <button type="submit" class="btn">Get Started</button>
</div>
<ul class="list-login">
    <li>

    </li>
    <li class="clearfix"></li>
</ul>
<ul class="list-login-bottom">
    <li class="">

    </li>
    <li class="">

    </li>
    <li class="clearfix"></li>
</ul>
</form>
</div>
</div>
<div class="bottom-grid1">
    <div class="links">

    </div>
    <div class="copyright">
    </div>
</div>
</div>
</section>
</body>

```

## Admin home

```

{% load static %}
<head>
    <meta charset="utf-8">
    <meta content="width=device-width, initial-scale=1.0" name="viewport">

    <title>MedEvac</title>
    <meta content="" name="description">
    <meta content="" name="keywords">

    <link href="{% static 'mainassets/img/favicon.png' %}" rel="icon">
    <link href="{% static 'mainassets/img/apple-touch-icon.png' %}" rel="apple-touch-
icon">

    <link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,7
00i|Nunito:300,300i,400,400i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,60
0i,700,700i" %}" rel="stylesheet">

    <!-- Vendor CSS Files -->
    <link href="{% static 'mainassets/vendor/bootstrap/css/bootstrap.min.css' %}"

```

```

rel="stylesheet">
<link href="{% static 'mainassets/vendor/bootstrap-icons/bootstrap-icons.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/boxicons/css/boxicons.min.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/glightbox/css/glightbox.min.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/remixicon/remixicon.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/swiper/swiper-bundle.min.css' %}"
rel="stylesheet">

<link href="{% static 'mainassets/css/style.css' %}" rel="stylesheet">

</head>

<body>

<header id="header" class="fixed-top ">
<div class="container d-flex align-items-center justify-content-between">

    <!-- <a href="index.html" class="logo"></a-->

    <nav id="navbar" class="navbar">
        <ul>
            <li><a class="nav-link scrollto" href="#home">Home</a></li>
            <li><a class="nav-link " href="#mngghsp">Manage Hospital</a></li>
            <li><a class="nav-link " href="#mngpamb">Manage Ambulance</a></li>
            <li><a class="nav-link " href="#add_manage_officer">Manage
Officer</a></li>
            <li><a class="nav-link " href="#trackloc">Location & Status </a></li>
            <li><a class="nav-link " href="#viewfeedback">Feedbacks</a></li>
            <li><a class="nav-link " href="#logout">LOGOUT</a></li>
        </ul>
        <i class="bi bi-list mobile-nav-toggle"></i>
    </nav><!-- .navbar -->

</div>
</header><!-- End Header -->

<!-- ===== Hero Section ===== -->
<section id="hero">
<div class="hero-container">
    <h3><strong>MedEvac</strong></h3>

</div>
</section><!-- End Hero -->

<main id="main">

    <!-- ===== About Section ===== -->
    <section id="about" class="about">
        <div class="container">

            {% block body %}
            {% endblock %}

        </div>

    </div>

</section><!-- End About Section -->
<!-- Vendor JS Files -->
<script src="{% static
'mainassets/vendor/bootstrap/js/bootstrap.bundle.min.js' %}"></script>

```

```

<script src="{% static 'mainassets/vendor/glightbox/js/glightbox.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/isotope-
layout/isotope.pkgd.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/swiper/swiper-bundle.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/php-email-form/validate.js' %}"></script>

<!-- Template Main JS File -->
<script src="{% static 'mainassets/js/main.js' %}"></script>

</body>

```

## Appraiser Home

```

{% load static %}
<head>
<meta charset="utf-8">
<meta content="width=device-width, initial-scale=1.0" name="viewport">

<title>MedEvac</title>
<meta content="" name="description">.

<meta content="" name="keywords">

<link href="{% static 'mainassets/img/favicon.png' %}" rel="icon">
<link href="{% static 'mainassets/img/apple-touch-icon.png' %}" rel="apple-touch-
icon">

<link
href="https://fonts.googleapis.com/css?family=Open+Sans:300,300i,400,400i,600,600i,700,7
00i|Nunito:300,300i,400,400i,600,600i,700,700i|Poppins:300,300i,400,400i,500,500i,600,60
0i,700,700i" rel="stylesheet">

<link href="{% static 'mainassets/vendor/bootstrap/css/bootstrap.min.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/bootstrap-icons/bootstrap-icons.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/boxicons/css/boxicons.min.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/glightbox/css/glightbox.min.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/remixicon/remixicon.css' %}"
rel="stylesheet">
<link href="{% static 'mainassets/vendor/swiper/swiper-bundle.min.css' %}"
rel="stylesheet">

<link href="{% static 'mainassets/css/style.css' %}" rel="stylesheet">

</head>

<body>

<header id="header" class="fixed-top">
<div class="container d-flex align-items-center justify-content-between">

<!-- <a href="index.html" class="logo"></a-->

<nav id="navbar" class="navbar">
<ul>
<li><a class="nav-link scrollto" href="#home">Home</a></li>
<li><a class="nav-link scrollto" href="#mnghsp#about">Manage Hospital</a></li>
<li><a class="nav-link scrollto" href="#add_manage_officer#about">Manage
Officer</a></li>
<li><a class="nav-link scrollto" href="#viewfeedback#about">Feedbacks</a></li>
<li><a class="nav-link scrollto" href="#logout">LOGOUT</a></li>
</ul>
<i class="bi bi-list mobile-nav-toggle"></i>
</nav><!-- .navbar -->

```

```

    </div>
</header><!-- End Header -->

<!-- ===== Hero Section ===== -->
<section id="hero">
    <div class="hero-container">
        <h3><strong>MedEvac</strong></h3>

    </div>
</section><!-- End Hero -->

<main id="main">

    <!-- ===== About Section ===== -->
    <section id="about" class="about">
        <div class="container">

            {% block body %}
            {% endblock %}

        </div>

    </div>
</section><!-- End About Section -->
<!-- Vendor JS Files -->
<script src="{% static
'mainassets/vendor/bootstrap/js/bootstrap.bundle.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/glightbox/js/glightbox.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/isotope-
layout/isotope.pkgd.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/swiper/swiper-bundle.min.js' %}"></script>
<script src="{% static 'mainassets/vendor/php-email-form/validate.js' %}"></script>

<!-- Template Main JS File -->
<script src="{% static 'mainassets/js/main.js' %}"></script>
</body>

```

## Flutter login

```
import 'dart:convert';

import 'package:aambulance_tracking/home.dart';
import 'package:aambulance_tracking/location.dart';
import 'package:aambulance_tracking/main.dart';
import 'package:aambulance_tracking/officer%20home.dart';
import 'package:aambulance_tracking/registration.dart';
import 'package:aambulance_tracking/user%20home.dart';
import 'package:flutter/material.dart';
import 'package:flutter/services.dart';
import 'package:shared_preferences/shared_preferences.dart';
import 'package:http/http.dart' as http;

class login extends StatefulWidget {
  const login({super.key});

  @override
  State<login> createState() => _loginState();
}

class _loginState extends State<login> {
  final TextEditingController usernameController = TextEditingController();
  final TextEditingController passwordController = TextEditingController();
  final _formKey = GlobalKey<FormState>(); // Add a global key for the form
```

```

@override
Widget build(BuildContext context) {
  return Scaffold(
    appBar: AppBar(
      title: const Text("Login"),
    ),
    body: WillPopScope(
      child: SafeArea(
        child: Form(
          key: _formKey,
          child: Stack(
            children: <Widget>[
              // Background image
              Image.asset(
                'assets/login.jpg',
                fit: BoxFit.cover,
                width: double.infinity,
                height: double.infinity,
              ),

              Center(
                child: Column(
                  mainAxisAlignment: MainAxisAlignment.center,
                  crossAxisAlignment: CrossAxisAlignment.center,
                  children: [
                    Padding(
                      padding: const EdgeInsets.all(8.0),
                      child: TextFormField(
                        controller: usernameController,
                        decoration: const InputDecoration(
                          filled: true,

```



```

    fillColor: Colors.transparent,
    border: OutlineInputBorder(
      borderSide: BorderSide(
        color: Colors.blue), // Set the border color
    ),
    hintText: "Username",
    hintStyle: TextStyle(color: Colors.white),
  ),
  style: TextStyle(
    color: Colors.white,
  ), validator: (value) {
    if (value!.isEmpty) {
      return 'Please enter username';
    }
    return null; // Return null if the input is valid
  },
),
),
),
Padding(
  padding: const EdgeInsets.all(8.0),
  child: TextFormField(
    obscureText: true,
    controller: passwordController,
    decoration: const InputDecoration(
      filled: true,
      fillColor: Colors.transparent,
      border: OutlineInputBorder(
        borderSide: BorderSide(
          color: Colors.blue), // Set the border color
        ),
      hintText: "Password",
      hintStyle: TextStyle(color: Colors.white),

```

```

    ),
    style: TextStyle(
      color: Colors.white,
    ), validator: (value) {
    if (value!.isEmpty) {
      return 'Please enter password';
    }
    return null; // Return null if the input is valid
  },
),
),
Padding(
  padding: const EdgeInsets.all(16.0),
  child: Row(
    mainAxisAlignment: MainAxisAlignment
      .center, // Center align the buttons
    children: [
      Expanded(
        child: ElevatedButton(
          onPressed: () async {
            if (!_formKey.currentState!.validate()) {
              print("Not validated");
            } else {
              final sh =
                await SharedPreferences.getInstance();
              print("=====");
              String Uname =
                usernameController.text.toString();
              String Paswd =
                passwordController.text.toString();
              String url = sh.getString("url").toString();
              print("=====+++++++" + url);
            }
          },
        ),
      ),
    ],
  ),
),

```

```

print(Uname);
print(Paswd);

var data = await http.post(
  Uri.parse(url + "and_login"),
  body: {
    'username': Uname,
    "password": Paswd,
  },
);

var jasondata = json.decode(data.body);
String status =
  jasondata['status'].toString();
print(
  "===== " + status);

if (status == "ambulance") {
  String lid = jasondata['lid'].toString();
  sh.setString("lid", lid);
  sh.setString("type", "a");
  print(lid);
  Navigator.push(
    context,
    MaterialPageRoute(
      builder: (context) => sapp()));
  Navigator.push(
    context,
    MaterialPageRoute(
      builder: (context) => Home()));

  print("+++++++Ambulance+++++++");

```

```

} else if (status == "user") {
    String lid = jasondata['lid'].toString();
    sh.setString("lid", lid);
    print(lid);
    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => UserHome()));
    print("=====User=====");
}
else if (status == "officer") {
    String lid = jasondata['lid'].toString();
    sh.setString("lid", lid);
    sh.setString("type", "o");
    print(lid);
    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => sapp()));
    Navigator.push(
        context,
        MaterialPageRoute(
            builder: (context) => officerhome()));
    print("=====Officer=====");
}
else {
    // Invalid username or password
    ScaffoldMessenger.of(context).showSnackBar(
        SnackBar(
            content: Text(
                'Invalid username or password.'),
            duration: Duration(seconds: 3),

```

```

        ),
      );
    }
  }},
  child: Text(
    "Login",
    style: TextStyle(fontSize: 16.0),
  ),
),
),
),
  SizedBox(
    width: 16.0), // Add space between the buttons
  Expanded(
    child: ElevatedButton(
      onPressed: () {
        // Navigate to your registration or sign-up screen here
        Navigator.push(
          context,
          MaterialPageRoute(
            builder: (context) =>
              Registration()));
      },
      child: Text(
        "Sign Up",
        style: TextStyle(fontSize: 16.0),
      ),
    ),
  ),
),
],
),
],
],

```

```

    ),
    ),
  ],
),
),
),
onWillPop: () async {
  // Handle the back button press or pop gesture here
  // Navigator.push(
  //   context, MaterialPageRoute(builder: (context) => Ambulance());
  showExitConfirmationDialog(context);
  return true;
},
),
);
}
}

Future<bool> showExitConfirmationDialog(BuildContext context) async {
  return await showDialog(
    context: context,
    builder: (context) {
      return AlertDialog(
        title: Text('Confirm Exit'),
        content: Text('Do you want to exit the app?'),
        actions: <Widget>[
          TextButton(
            onPressed: () {
              SystemNavigator.pop();
              // Navigator.push(context, MaterialPageRoute(builder: (context) => login())); // Allow exit
            },
            child: Text('Yes'),
          ),

```

```
    TextButton(  
      onPressed: () {  
        Navigator.of(context).pop(false); // Don't exit  
      },  
      child: Text('No'),  
    ),  
  ],  
);  
,  
  
);  
}
```

**BIBLIOGRAPHY****10.BIBLIOGRAPHY**

- <https://www.w3schools.com>
- <https://github.com/>
- <https://openai.com/>
- <https://www.djangoproject.com/>
- <https://react-bootstrap.netlify.app/>