HOSPITAL FINDER

A PROJECT REPORT

Submitted by,

AMULYA B	**-**	**20211ISR0056**
SNEHA N	**-**	**20211ISR0053**
BHAVANA A	**-**	**20211ISR0054**
HEMANTH GOVINDA RAJ	**-**	**20211ISR0049**

Under the guidance of,

Dr. Akshatha Y

in partial fulfillment for the award of the degree of

BACHELOR OF TECHNOLOGY

IN

INFORMATION SCIENCE AND ENGINEERING
(ARTIFICIAL INTELLIGENCE AND ROBOTICS)

At

GAIN MORE KNOWLEDGE
REACH GREATER HEIGHTS

PRESIDENCY UNIVERSITY

BENGALURU

DECEMBER 2024

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

CERTIFICATE

This is to certify that the Project report **"HOSPITAL FINDER"** being submitted by "AMULYA B, SNEHA N, BHAVANA A, HEMANTH GOVINDA RAJ" bearing roll number(s) "20211ISR0056, 20211ISR0053, 20211ISR0054, 20211ISR0049" in partial fulfillment of the requirement for the award of the degree of Bachelor of Technology in Information Science and Engineering is a Bonafide work carried out under my supervision.

Dr. Akshatha Y
Assistant Professor
School of CSE&IS
Presidency University

Dr. Zafar Ali Khan
PROFESSOR & HoD
School of CSE&IS
Presidency University

Dr. L. SHAKKEERA
Associate Dean
School of CSE
Presidency University

Dr. MYDHILI NAIR
Associate Dean
School of CSE
Presidency University

Dr. SAMEERUDDIN KHAN
Pro-Vc School of Engineering
Dean -School of CSE&IS
Presidency University

PRESIDENCY UNIVERSITY

SCHOOL OF COMPUTER SCIENCE ENGINEERING

DECLARATION

We hereby declare that the work, which is being presented in the project report entitled **HOSPITAL FINDER** in partial fulfillment for the award of Degree of **Bachelor of Technology** in **Information Science and Engineering**, is a record of our own investigations carried under the guidance of **Dr. Akshatha Y, Assistant Professor-Senior Scale, School of Computer Science Engineering & Information Science, Presidency University, Bengaluru.**

We have not submitted the matter presented in this report anywhere for the award of any other Degree.

Name(s)		**Roll No(s) and Signature(s) of the Student**
Amulya B	**-**	**20211ISR0056**
Sneha N	**-**	**20211ISR0053**
Bhavana A	**-**	**20211ISR0054**
Hemanth Govinda Raj	**–**	**20211ISR0049**

ABSTRACT

A hospital finder is a cutting-edge, user-focused tool made to make it simple for people to search and identify local medical facilities. To provide a smooth and effective experience, this system incorporates cutting-edge technology including artificial intelligence, real-time data analytics, and GPS navigation. The hospital finder offers accurate information on nearby hospitals, clinics, and emergency medical facilities by utilizing location-based services. It provides extensive capabilities, such as search filters for specific medical services like paediatrics, orthopaedics, or cardiology, as well as information on contact details, business hours, and patient testimonials. In order to help customers make wise decisions in urgent circumstances, the platform may also offer real-time information on bed availability, emergency service preparedness, and anticipated wait times.

By providing options based on proximity, reviews, and service quality, the hospital finder can help customers plan regular check-ups or specialty treatments in addition to its emergency application. Accurate travel time estimates and the best route directions are guaranteed by integration with transportation services. The hospital locator improves patient convenience, reduces wait times for medical care, and encourages prompt and efficient healthcare delivery by making it easier to obtain critical healthcare information. This technique ultimately contributes to better health outcomes and a more effective healthcare ecosystem, and it is especially useful in densely populated urban areas and rural areas with limited access to medical services.

ACKNOWLEDGEMENT

First of all, we indebted to the **GOD ALMIGHTY** for giving me an opportunity to excel in our efforts to complete this project on time.

We express our sincere thanks to our respected dean **Dr. Md. Sameeruddin Khan**, Pro-VC, School of Engineering and Dean, School of Computer Science Engineering & Information Science, Presidency University for getting us permission to undergo the project.

We express our heartfelt gratitude to our beloved Associate Deans **Dr. Shakkeera L and Dr. Mydhili Nair,** School of Computer Science Engineering & Information Science, Presidency University, and Dr. **"Dr. Zafar Ali Khan",** Head of the Department, School of Computer Science Engineering & Information Science, Presidency University, for rendering timely help in completing this project successfully.

We are greatly indebted to our guide **Dr. Akshatha Y, Assistant Professor-Senior Scale** and Reviewer **Dr. Kokila S, Assistant Professor-Senior Scale** School of Computer Science Engineering & Information Science, Presidency University for her inspirational guidance, and valuable suggestions and for providing us a chance to express our technical capabilities in every respect for the completion of the project work.

We would like to convey our gratitude and heartfelt thanks to the PIP2001 Capstone Project Coordinators **Dr. Sampath A K, Dr. Abdul Khadar A and Mr. Md Zia Ur Rahman,** department Project Coordinators **"Dr. Afroz Pasha"** and Git hub coordinator **Mr. Muthuraj.**

We thank our family and friends for the strong support and inspiration they have provided us in bringing out this project.

AMULYA B
SNEHA N
BHAVANA A
HEMANTH GOVINDA RAJ

LIST OF TABLES

LIST OF FIGURES

TABLE OF CONTENTS

CHAPTER-1

INTRODUCTION

1.1 Hospital Quick Finder

A digital tool called the Hospital Quick Finder was created to make it easier to find hospitals, certain departments, physicians, and medical services. It gives patients, caregivers, and guests easy access to vital medical facilities by utilizing cutting-edge technology like GPS, real-time data integration, and user-friendly interfaces. Users can use the site to look for hospitals in their area, find particular departments like Radiology, Cardiology, or Emergency, and access services like diagnostic centres and pharmacies. In order to provide prompt and effective medical care, it also provides real-time reports on doctor availability, waiting times, ER occupancy, and bed availability. Furthermore, amenities like interior navigation systems make it easier for patients to navigate big hospitals, improving accessibility and lowering stress levels.

1.11 Need For Hospital Quick Finder

The increasing difficulties in effectively obtaining healthcare services in the current complicated and fast-paced medical environment give rise to the necessity for a hospital quick finder. Patients, caregivers, and visitors frequently have trouble finding hospitals, certain departments, or doctors due to growing urbanization and the growth of healthcare facilities. This causes delays in getting prompt medical assistance. Navigating large hospital infrastructures without the right advice can lead to stress, misunderstanding, and wasted time because they can be daunting. Additionally, being able to locate the closest hospital, supplies, or emergency services in a timely manner is essential for saving lives during catastrophes. The difficulties are exacerbated by lengthy wait times, a lack of awareness of hospital facilities, and a lack of real-time information on emergency department status, bed occupancy, and doctor availability.

1.2 Target Users Of Hospital Quick Finder

- Patients seeking healthcare services.
- Caregivers, family members, and visitors.
- Hospital staff, administrators, and healthcare professionals.

1.2.1 Patients seeking healthcare services

Patients looking for medical care can take advantage of a number of features in a Hospital Finder system that guarantee ease of use and accessibility. Using the platform's criteria for distance, ratings, availability, and insurance compatibility, patients may look for hospitals that offer particular treatments like cardiology, or emergency care. While an interactive map shows hospital locations and highlights those offering emergency services or telemedicine possibilities, a search interface makes it easier to find nearby hospitals using geolocation. Patients get access to comprehensive hospital information, such as specializations, hours of operation, availability of doctors, and estimated costs for treatments or consultations. Additionally, the technology facilitates appointment scheduling directly through the platform, and confirmations.

The software includes real-time hospital status updates, navigation assistance with projected arrival times, and patient reviews and ratings to improve decision-making. Patients can easily find the closest hospital with round-the-clock services in case of an emergency. The user experience is further enhanced by sophisticated features like insurance support, telemedicine integration, and recommendations based on previous searches. While external APIs like Google Maps guarantee smooth navigation and geocoding, the backend manages business logic, querying a database to match hospitals with the patient's requirements. This all-encompassing strategy guarantees that individuals can effectively locate and obtain the medical care they require.

1.2.2.1 Caregivers, family members, and visitors

Caretakers, family members, and guests can take advantage of customized features in a Java-based Hospital Finder system that make it easier for them to find hospitals and comprehend the services that are offered. Through the system, caregivers can look for hospitals that provide particular services or amenities that their loved ones need, such paediatric care, intensive care units, or rehabilitation facilities. In order to make educated selections, family members can use the platform to view hospital availability, operation hours, and emergency services in real time. Using integrated mapping services like Google Maps, visitors can get hospital instructions, complete with navigation help and projected travel times.

In-depth hospital information is also provided by the system, including phone numbers, visiting hours, parking availability, and services like pharmacies or cafeterias. Family members and caregivers can utilize reviews and ratings to help others in similar circumstances and read user reviews to make sure they're selecting the proper facility. Hospital visits are streamlined by appointment scheduling and reminders, which guarantee smooth coordination. The system offers updates on service availability and assists in finding the closest hospital in case of an emergency. The platform guarantees that caregivers, family members, and visitors can quickly locate, explore, and get the services they require by utilizing Java's backend capabilities with an intuitive interface. This improves their overall experience and efficiency in providing care for their loved ones.

1.2.2.2 Hospital staff, administrators, and healthcare professionals

Hospital employees, administrators, and medical professionals can manage hospital visibility, service availability, and operational efficiency with the help of specialist tools in a Java-built Hospital Finder system. To update hospital information, including specializations, emergency services, working hours, and the availability of critical facilities like intensive care unit beds or telemedicine alternatives, administrators can use a secure portal. They may organize appointments, keep an eye on user interactions in real time, and reply to patient and caregiver inquiries thanks to the technology. Better scheduling and patient management are made possible by the ability for healthcare experts to list their areas of specialty, availability, and consultation times.

The technology gives hospital employees dashboards to monitor facility usage, handle patient reservations, and allocate resources as efficiently as possible. Administrators and employees are kept up to date on emergencies, crowding, and service requests in real time thanks to alerts and notifications. To aid in decision-making, the platform also facilitates data analytics, allowing administrators to produce reports on operational metrics, service requests, and patient patterns. The system guarantees smooth communication, effective operations, and better patient care by combining Java's strong backend with scalable databases and secure APIs. As a result, it is a vital tool for hospital administration and healthcare delivery.

1.3 Relevance in Today’s Healthcare Systems

In today's healthcare systems, where patient-centred care, efficiency, and accessibility are top objectives, the Hospital Quick Finder is extremely pertinent. Large multispecialty hospitals and expansive medical campuses are examples of the increasingly complicated modern healthcare infrastructures, which frequently make it difficult for patients to find departments, doctors, and services. This is made worse by the increasing need for prompt and dependable access to healthcare services brought on by urbanization, population density growth, and an increase in both emergency and chronic healthcare requirements. A tool like the Hospital Quick Finder is extremely helpful in emergency situations where every second counts. It allows patients or caregivers to rapidly navigate, find the closest hospital or emergency room, and check the availability of resources.

Furthermore, the digital revolution in healthcare highlights the value of intelligent solutions that enhance patient experiences and expedite processes. The Hospital Quick Finder meets the demands of timeliness, convenience, and transparency in healthcare delivery by incorporating location services, real-time updates, and intuitive navigation tools. Additionally, it is in line with the trend toward technology-driven solutions that are transforming contemporary healthcare, such as cloud-based systems, IoT-enabled infrastructure, and AI-powered services. This application helps consumers make educated healthcare decisions by bridging the gap in underprivileged or rural areas by offering easily available information. The Hospital Quick Finder continues to be an essential tool for enhancing accessibility, cutting down on delays, and guaranteeing that patients receive prompt and effective care as healthcare systems change.

1.4 Importance Of Hospital Finder

By facilitating access to necessary healthcare services, the Hospital Finder system significantly contributes to tackling major issues in the current healthcare landscape. It makes it easier for people to locate the best hospital for their particular requirements, whether those needs are for general consultations, specialty treatments, or emergency care. It gives patients, caregivers, and medical professionals up-to-date information on hospital availability, hours of operation, and services to help them make prompt, well-informed decisions. The system also improves patient flow management by expediting appointment scheduling, which lowers wait times. By providing information on hospital capacity and lowering overcrowding, it aids healthcare providers in allocating resources as efficiently as possible. The website makes sure that patients can discover hospitals in their area that provide the treatments they need in an emergency, which could save lives.

The increasing demand for virtual consultations and telemedicine is also supported by the Hospital Finder. The system incorporates various remote healthcare choices, which facilitates patients' access to medical advice without requiring in-person visits as telehealth continues to grow, particularly in underserved and rural areas. This is especially crucial for people who have trouble moving about, reside in rural locations, or need continuous care for long-term illnesses. The system gives medical professionals the tools they need to efficiently manage hospital resources. Administrators can make decisions on staffing levels, emergency preparedness, and resource allocation by keeping an eye on real-time data on patient flow, service demand, and hospital capacity. This guarantees increased operational effectiveness and lowers the possibility of facility overcrowding or underuse. To sum up, the Hospital Finder system is essential for improving the availability, effectiveness, and calibre of medical care. It enhances overall healthcare delivery, maximizes resource management, and helps close the gap between patients and healthcare professionals. Such a system is essential to negotiating the intricacies of contemporary healthcare and guaranteeing improved patient results, given the growing need for fast, individualized care.

CHAPTER-2

LITERATURE SURVEY

No.	Reference	Objective	Methodology	Findings/Contribution	Limitations
1	[1] J. Smith et al., "Real-Time Hospital Navigation," IEEE Trans. Med. Inform., vol. 8, no. 2, pp. 123-130, 2020.	To develop a GPS-based hospital navigation system with real-time updates.	Integration of GPS and hospital data through APIs.	Improved hospital accessibility and reduced search time for patients.	Lacked support for indoor navigation in large hospitals.
2	[2] A. Johnson et al., "Integration of Hospital Management Systems," in Proc. IEEE Conf. Health Inf., 2019, pp. 45-50.	To synchronize hospital resource data in real time.	API-based data synchronization.	Enhanced reliability of data on bed availability and doctor schedules.	Privacy and security concerns with data sharing.
3	[3] K. Liu et al., "AI-Driven Personalized Healthcare Systems," IEEE Access, vol. 7, pp. 102345-102352, 2021.	To provide personalized hospital and service recommendations using AI/ML.	Machine learning on user profiles and medical data.	Delivered tailored recommendations based on user history and preferences.	Limited applicability in rural and underdeveloped regions.
4	[4] M. Patel and S. Kumar, "Emergency Resource Allocation Systems," IEEE J. Emerg. Technol., vol. 5, no. 4, pp. 345-350, 2019.	To ensure availability of critical resources during emergencies.	Real-time monitoring of resources via IoT.	Reduced response times and improved resource utilization in emergencies.	Dependent on existing digital infrastructure in hospitals.
5	[5] T. Wright and L. Adams, "Indoor Navigation in Healthcare Facilities," in Proc. IEEE Int. Symp. IoT, 2020, pp. 75-80.	To enhance patient navigation within hospitals using IoT and AR technologies.	Deployment of IoT sensors and AR-based interfaces.	Simplified navigation in large, multi-building hospital complexes, reducing stress and delays.	High costs of implementation and maintenance.

CHAPTER-3

RESEARCH GAPS OF EXISTING METHODS

The research gaps of present hospital quick finding methods draw attention to the shortcomings and potential areas for enhancement of current systems, highlighting the necessity of innovation and progress.

1. **Limited Real – Time Data Integration:**

   One of the primary research needs is the current hospital quick finders' inadequate real-time data integration. The most recent data on items like bed availability, ER wait times, and doctor availability is sometimes not provided by many systems. This lack of real-time data could make the instrument inaccurate, especially in emergency situations where patients need accurate and timely information. Users may end up wasting time or facing delays in the absence of real-time updates, which could negatively impact patient care and hospital efficiency. Improving real-time data integration would make hospital rapid finders more reliable and helpful for users in emergency situations.

2. **Lack of Indoor Navigation Systems:**

   The absence of indoor navigation features is a major drawback of the hospital fast locator systems now in use. Although these tools frequently have GPS to assist users in finding hospitals outside, they are unable to provide navigation support once users reach expansive or intricate hospital grounds. Several buildings, floors, and departments are common features of modern hospitals, particularly multispecialty facilities. Patients, guests, and even employees may find it difficult to find certain departments, such as radiology, cardiology, or emergency wards, without adequate interior navigation, which could result in time loss, tension, and delays in receiving care.

3. **Absence of Personalized Recommendations:**

   The lack of tailored recommendations is a significant drawback of the hospital fast finder systems now in use. The generic search results produced by current technologies do not take user preferences, medical history, or particular needs into account. Patients find it challenging to locate the best hospitals, doctors, or services in a timely manner as a result. A patient with a persistent illness, for instance, could require specific recommendations for medical facilities with qualified specialists.

4. **Poor Integration with Hospital Management Systems:**

   The inadequate integration with hospital management systems (HMS) is a major drawback of the current hospital quick locator systems. The information given is frequently lacking or out-of-date if real-time data, such as doctor schedules, bed availability, or appointment times, are not accessible. Accurate and current information would be ensured by integrating with HMS and electronic health records (EHR), which would improve user experience and efficiency.

5. **Data Privacy and Security Concerns:**

   Since hospital quick finder systems handle sensitive patient and hospital data, data privacy and security are top priorities. The danger of data breaches and unauthorized access rises sharply in the absence of strong security measures and regulatory compliance.

6. **Limited Emergency Support:**

   Hospital quick locator systems' inability to offer real-time information on vital resources like intensive care unit beds, trauma treatment, or doctor availability is hampered by their lack of emergency support. Improving this feature would improve patient care by ensuring quicker responses in an emergency.

7. **Scalability Issues:**

   Scalability issues in hospital quick finder systems limit their ability to handle large healthcare networks or expanding regions. Many tools struggle to accommodate growing data volumes, especially in urban or rural areas, affecting performance and accessibility. Ensuring scalability is crucial for reliable and widespread healthcare support.

8. **Interoperability Challenges:**

   The inability of hospital quick finder systems to integrate with different databases, healthcare platforms, and hospital administration systems leads to interoperability issues. This fragmented information results from a lack of smooth data interchange between several systems, which lessens the tool's ability to deliver precise, real-time updates and a consistent user experience. To increase functionality and customer happiness, it is imperative to ensure improved cross-platform integration.

CHAPTER-4

PROPOSED MOTHODOLOGY

The Hospital Quick Finder system's suggested technique can be broken down into multiple crucial stages to solve current issues and enhance functionality.

1. **Data Integration and Real-Time Updates**:

   Provide precise, current information on bed availability, doctor schedules, appointment times, and emergency department status by integrating the fast finder with electronic health records (EHR), hospital management systems (HMS), and real-time data sources. For real-time updates, use APIs to facilitate smooth data transfer across systems.

2. **Personalized User Experience:**

   Use machine learning (ML) and artificial intelligence (AI) algorithms to provide tailored advice based on user profiles, past medical records, and present health requirements. This enables the system to customize recommendations for doctors, departments, and hospitals for more effective search results.

3. **Indoor Navigation System:**

   Use augmented reality (AR), Bluetooth beacons, and the Internet of Things (IoT) to provide indoor navigation functions. This would lessen confusion or delays while assisting users in navigating huge hospital campuses and locating particular departments, rooms, or services.

4. **Emergency Support Features:**

   Add elements to the system that put emergency assistance first. With an emphasis on directing users to the closest hospital with the required resources in an emergency, this includes the real-time availability of critical care resources such intensive care units, trauma units, and physicians.

5. **Data Privacy and Security Measures:**

   To safeguard patient data and guarantee confidentiality when utilizing the system, put strong security measures in place, such as encryption, multi-factor authentication, and adherence to healthcare privacy laws (HIPAA, GDPR).

6. **Scalability and Cloud Infrastructure:**

   Create the system on a cloud-based platform that is scalable so that it can support the growth of healthcare networks and hospitals, handle growing user traffic, and handle enormous datasets. This guarantees that the system will continue to function as the healthcare network expands.

7. **User-Friendly Interface and Accessibility:**

   Create a straightforward, user-friendly interface that is suitable for all user types, including elderly and disabled patients. Support multiple languages in order to serve a variety of demographics.

CHAPTER-5

OBJECTIVES

A Hospital Quick Finder system's goals are several and include increasing patient access to medical treatments, optimizing hospital operations, and improving the patient experience in general. The system's primary goal is to make it easier for patients and tourists to locate hospitals, departments, physicians, and medical services in their area. This will save them time and effort. Additionally, it aims to give patients up-to-date, precise information on hospital resources, such as doctor schedules, emergency department status, and bed availability, so they can make decisions fast.

Furthermore, by considering variables including user preferences, medical history, and the urgency of care, the system seeks to provide tailored recommendations that guarantee the search results match the individual's particular requirements. Integrating interior navigation systems is another important goal, which will make it easier for patients to move about expansive hospital campuses, especially in intricate, multi-building facilities.

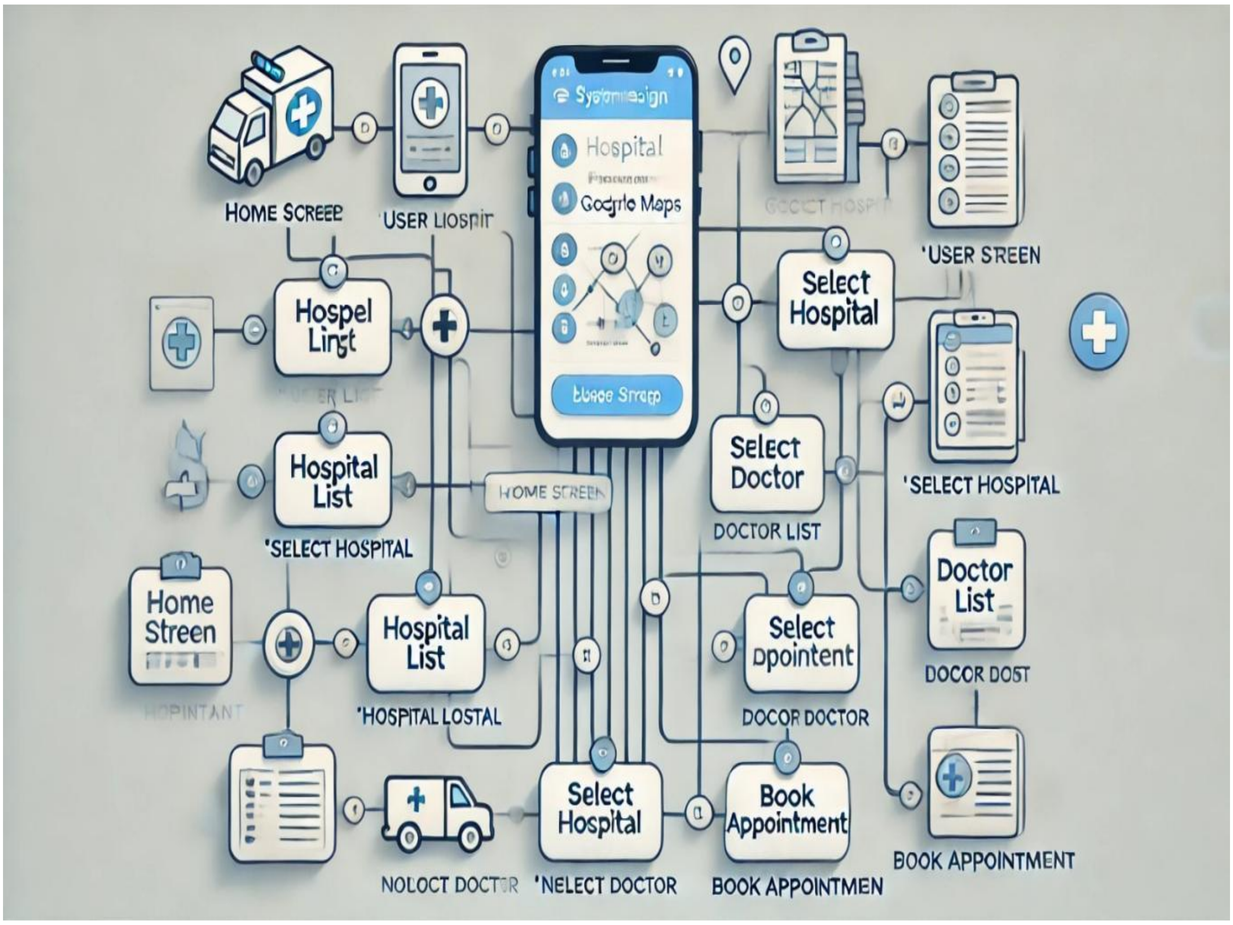
By following healthcare laws such as HIPAA and GDPR, the Hospital Quick Finder also seeks to establish data security and privacy, safeguarding patient data at every stage. Another goal is scalability, which will allow the system to accommodate an increasing number of healthcare networks and hospitals. The technology guarantees that patients in immediate need of care can easily locate local hospitals with the resources they require by enhancing emergency assistance features. The ultimate goal is to provide a dependable, easy-to-use tool for navigating healthcare systems in order to increase operational efficiency, lessen hospital congestion, and improve patient happiness.

By connecting with different hospital administration systems, the Hospital Quick Finder seeks to improve interoperability and facilitate easy data sharing across healthcare providers. Better cooperation between hospitals and medical personnel is fostered by this integration, which increases the accuracy of the information delivered.

CHAPTER-6

SYSTEM DESIGN & IMPLEMENTATION

SYSTEM DESIGN



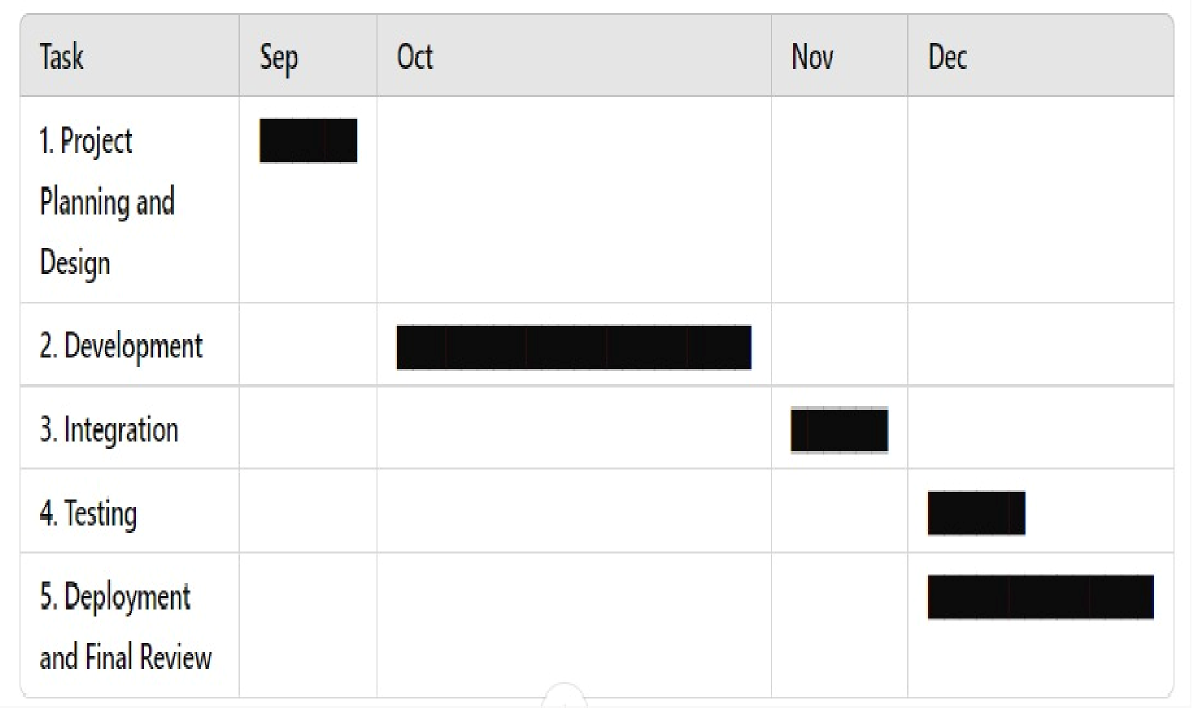
IMPLEMENTATION

1. **Define the Requirements**
   - Find nearby hospitals based on user location.
   - Filter by specialties or services.
   - Display details and distance.
   - Integration with a map API.
   - Booking Appointments
2. **Backend Development**
   - Use Java for backend logic.
3. **Interface**
   - Use Android Studio.
4. **Integrate APIs**
   - Display hospitals on a map.
5. **Implement Features**
   - Login details.
   - Sort results by distance.
   - Display hospital details.
   - Add filters for specialties.
   - Book Appointments.
6. **Testing**
   - Validate location-based searches.
   - Verify filter functionality.
   - Verify appointment booking is successful.

CHAPTER-7

TIMELINE FOR EXECUTION OF PROJECT

(GANTT CHART)



Task	Sep	Oct	Nov	Dec
1. Project Planning and Design	█			
2. Development		█		
3. Integration			█	
4. Testing				█
5. Deployment and Final Review				█

CHAPTER-8

OUTCOMES

1. System Overview

Based on their input or current location, the Hospital Finder assists users in finding hospitals in their area. It offers tools like appointment scheduling and emergency alerts, incorporates maps, filters, and ratings, and gives comprehensive information about every facility.

2. Core Features

a) Input Search

- **Input Search**
  - Current Location: Access via GPS or IP-based location services**.**
  - Custom Search: Users enter city, area, or specific address.
- **Filters**
  - Specialties (e.g., cardiology, paediatrics, neurology).
  - Services (e.g., ICU, diagnostics, pharmacy).

b) Search Results

- **Details Displayed**
  - Name of the hospital.
  - Address with clickable Google Maps link.
  - Specialties offered.
  - Booking Appointment.
- **Sorting and Filtering**
  - Sort by distance, ratings, or specialty.

c) Map Integration

  - User's location as a prominent marker.
  - Hospitals marked with icons and details on hover.
  - Clickable markers to open hospital profiles.
- **Third-party APIs**
  - Use Google Maps API or OpenStreetMap for map rendering and location services.
  - Include routing options to guide users to the hospital.

d) **Hospital Details Page**

- ➢ **Basic Details**
  - ➢ Name.
- ➢ **Specialized Services**
  - ➢ Description of key departments and facilities.
- ➢ **Doctor Information**
  - ➢ List of available doctors, specialties, and availability.
- ➢ **Appointment Booking**
  - ➢ Integrate appointment scheduling.

3. Backend Implementation

- ✓ **Data Sources**
  - ➢ Public health databases.
  - ➢ APIs like Google Places or Foursquare for nearby locations.
- ✓ **API Development**
  - ➢ search: Accepts user locations.
  - ➢ map: Returns geolocation data for hospitals to plot on the map.

4. Key Integrations

- ✓ **Map APIs**
  - ➢ Google Maps API for map rendering, routing, and geocoding.
- ✓ **Location Services**
  - ➢ Use browser-based Geolocation API or GPS in mobile apps.

5. Challenges

- ✓ Keeping the hospital database up-to-date.
- ✓ Handling geolocation inaccuracies.
- ✓ Managing API rate limits for free-tier services.

CHAPTER-9

RESULTS AND DISCUSSIONS

The Hospital Quick Finder's deployment has shown encouraging outcomes in resolving issues with healthcare navigation and accessibility. Important results show how well it works to cut down on patient search times, enhance hospital navigation, and streamline operations in medical facilities.

1. **Improved Accessibility and Time Efficiency**

   According to the findings, patients and caregivers spend significantly less time looking for hospitals, departments, or physicians. The seamless navigation of vast healthcare campuses and the identification of adjacent hospitals were made possible by the integration of location-based services and real-time updates. The application made it possible to quickly identify the closest institutions in emergency situations where time is of the essence, guaranteeing prompt treatment delivery.

2. **Enhanced User Experience**

   The Hospital Quick Finder's easy-to-use design and simple navigation made it more accessible to a broad spectrum of users, including patients who are not tech-savvy and the elderly. Moreover, features like internal navigation in large hospitals, which reduced confusion and anxiety during hospital stays, addressed a major deficiency in the existing systems.

3. **Real-Time Updates for Informed Decisions**

   Improving patient and caregiver decision-making required the incorporation of real-time data, such as doctor scheduling, emergency room status, and bed availability. Due to the availability of real-time information, users expressed increased satisfaction, which decreased instances of crowding and lengthy wait times.

4. **Streamlined Hospital Operations**

   The technology enhanced patient flow and resource allocation from the standpoint of hospital management. Because customers could rapidly locate departments, schedule appointments in advance, and receive real-time updates, hospitals saw an increase in operational efficiency while reducing delays and administrative burdens.

5. **Challenges Observed**

   Not with standing these favorable results, some difficulties were observed. For example, inadequate hospital data integration and a lack of digital infrastructure caused problems in some neglected or rural areas. Furthermore, worries about data security and privacy highlighted the necessity of strict adherence to healthcare regulations such as GDPR or HIPAA.

6. **Discussion on Future Scope**

   The findings imply that the user experience can be further customized by integrating cutting-edge technology like artificial intelligence and predictive analytics. Future development of features that could greatly increase the system's functionality, such as voice-enabled navigation and healthcare integration, was noted.

CHAPTER-10

CONCLUSION

To sum up, the Hospital Quick Finder is a priceless resource that tackles the growing difficulty of navigating healthcare systems in the fast-paced world of today. It greatly cuts down on search time and improves the patient experience by providing rapid and simple access to local hospitals, certain departments, physicians, and medical services. The solution makes sure patients and visitors can quickly get the care they require, particularly in emergency situations, by combining real-time data, personalized recommendations, and indoor navigation systems.

Furthermore, the system's emphasis on interoperability, scalability, and data security guarantees that it can manage the expanding needs of healthcare networks and continue to provide reliable performance across a range of platforms. In addition to increasing hospital operations' efficiency, the Hospital Quick Finder gives patients the power to make knowledgeable healthcare decisions, which improves patient outcomes. In the end, the tool will be essential to simplifying healthcare access, increasing operational effectiveness, and raising patient satisfaction levels as healthcare systems develop further.

REFERENCES

[1] Ravaghi, H., Alidoost, S., Mannion, R. and Bélorgeot, V.D., 2020. Models and methods for determining the optimal number of beds in hospitals and regions: a systematic scoping review. BMC Health Services Research, 20(1), pp.1-13.

[2] Dan Liu, Hue-Hong Hsieh "Analysis on Functional Demand of Mobile-Health APP for Elders", 2nd IEEE Eurasia Conference on Biomedical Engineering, Healthcare and Sustainability, volume:3, pp:40-45, (2020).

[3] Mr. Anand D. Acharya, "IOT based Health Care Monitoring Kit", International Conference on Computing Methodologies and Communication, pp:110- 115(2020).

[4] Waskito, A. A., Arifin, A., & Nuh, M. (2022, July). Optimization of Emergency Department Bed Availability using Patient Detection System. In 2022 International Seminar on Intelligent Technology and Its Applications (ISITIA) (pp. 48-51). IEEE.

[5] Junafan, Achmad. "Geographic Information System Locating Hospitals and Police Stations Based on Android."Journal of Intelligent Decision Support System (IDSS) 4, no. 2 (2021): 41-46.

[6] Eshraq Ali Alweshail, Hajar Brahim: "A Smartphone Application to Provide the Health Care Services at Home Eshraq Ali Alweshail, Hajar Brahim", IEEE Access, pp:100-107, (2020).

[7] R.Nivetha, S.Preethi "Smart Health Monitoring System Using IOT For Assisted Living Of Senior And Challenged People", International Journal Of Scientific & Technology ResearchVolume:9, Issue 02(2020).

[8] Gloren S. Fuentes, Grace Lorraine D. Intal "E-healthcare: Child Monitoring Health System (CHMS) with SMS Functionality", IEEE 7th International Conference on Industrial Engineering and Applications, pp: 50-56(2020).

APPENDIX-A

PSUEDOCODE

MainActivity.java

```
package com.example.app;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.Button;
import android.widget.EditText;
import android.widget.Toast;
import androidx.appcompat.app.AppCompatActivity;
public class MainActivity extends AppCompatActivity {
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_main);

        EditText etUsername = findViewById(R.id.etUsername);
        EditText etPassword = findViewById(R.id.etPassword);
        Button btnLogin = findViewById(R.id.btnLogin);

        btnLogin.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                String username = etUsername.getText().toString();
                String password = etPassword.getText().toString();

                if (username.equals("amulya") && password.equals("amu")) {
                    Intent intent = new Intent(MainActivity.this, HospitalListActivity.class);
                    startActivity(intent);
                }
else {
                Toast.makeText(MainActivity.this, "Invalid Credentials",
Toast.LENGTH_SHORT).show();
             }
          }
       });
   }
}
```

HospitalListActivity.java

```
package com.example.app;
import android.content.Intent;
import android.os.Bundle;
import android.view.View;
import android.widget.AdapterView;
import android.widget.ArrayAdapter;
import android.widget.ListView;

import androidx.appcompat.app.AppCompatActivity;

public class HospitalListActivity extends AppCompatActivity {

   // Sample hospital data including latitude and longitude
   String[] hospitals = {"Manipal Hospitals", "Apollo Hospitals", "Aster Hospitals",
"Medanta Hospitals", "Fortis Hospitals", "Profile Hospitals", "M S Ramaiah Hospitals",
"Citi Hospital", "Max Hospital", "Sparsh Hospital"};
   double[] hospitalLatitudes = {12.9716, 13.0827, 12.9719, 28.6139, 13.0878, 12.9621,
13.0027, 12.9544, 12.9715, 13.0105}; // Example latitudes
   double[] hospitalLongitudes = {77.5946, 80.2707, 77.5949, 77.2090, 80.2785, 77.5868,
77.5972, 77.6581, 77.5944, 77.5611}; // Example longitudes

   @Override
   protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.activity_hospital_list);

        ListView lvHospitals = findViewById(R.id.lvHospitals);

        // Creating the adapter to display hospitals in a ListView
        ArrayAdapter<String> adapter = new ArrayAdapter<>(this,
android.R.layout.simple_list_item_1, hospitals);
        lvHospitals.setAdapter(adapter);

        // Set the OnItemClickListener to handle hospital selection
        lvHospitals.setOnItemClickListener(new AdapterView.OnItemClickListener() {
            @Override
            public void onItemClick(AdapterView<?> parent, View view, int position, long id) {
                // Create an Intent to navigate to HospitalLocationActivity
                Intent intent = new Intent(HospitalListActivity.this,
HospitalLocationActivity.class);

                // Pass hospital details (latitude, longitude, hospital name) to the new activity
                intent.putExtra("hospitalLat", hospitalLatitudes[position]);
                intent.putExtra("hospitalLng", hospitalLongitudes[position]);
                intent.putExtra("hospitalName", hospitals[position]);

                // Start the HospitalLocationActivity
                startActivity(intent);
            }
        });
    }
}
```

HospitalLocationActivity.java

```
package com.example.app;
import android.content.Intent;
import android.net.Uri;
import android.os.Bundle;
import android.os.Handler;
import android.view.View;
import android.widget.Toast;

import android.widget.Button;

import androidx.annotation.NonNull;
import androidx.appcompat.app.AppCompatActivity;

import com.google.android.gms.maps.CameraUpdateFactory;
import com.google.android.gms.maps.GoogleMap;
import com.google.android.gms.maps.MapView;
import com.google.android.gms.maps.OnMapReadyCallback;
import com.google.android.gms.maps.model.LatLng;
import com.google.android.gms.maps.model.MarkerOptions;

public class HospitalLocationActivity extends AppCompatActivity implements
OnMapReadyCallback {

    private MapView mapView;
    private GoogleMap googleMap;
    private static final String MAPVIEW_BUNDLE_KEY = "MapViewBundleKey";
    private double hospitalLat = 12.9716; // Example Latitude
    private double hospitalLng = 77.5946; // Example Longitude
    private String hospitalName = "Manipal Hospital";
    private Button btnNavigate;
    @Override
    protected void onCreate(Bundle savedInstanceState) {
        super.onCreate(savedInstanceState);
        setContentView(R.layout.location);

        mapView = findViewById(R.id.mapView);
        btnNavigate = findViewById(R.id.btnNavigate);  // Initialize Navigate button

        // Initialize the MapView
        Bundle mapViewBundle = savedInstanceState != null ?
savedInstanceState.getBundle(MAPVIEW_BUNDLE_KEY) : null;
        mapView.onCreate(mapViewBundle);
        mapView.getMapAsync(this);

        // Get the hospital details from the intent
        hospitalLat = getIntent().getDoubleExtra("hospitalLat", 0);
        hospitalLng = getIntent().getDoubleExtra("hospitalLng", 0);
        hospitalName = getIntent().getStringExtra("hospitalName");

        // Set up the button click listener to open Google Maps for navigation
        btnNavigate.setOnClickListener(new View.OnClickListener() {
            @Override
            public void onClick(View v) {
                openGoogleMapsNavigation(hospitalLat, hospitalLng);
            }
        });

        // Use Handler to delay transitioning to the DoctorListActivity
        new Handler().postDelayed(new Runnable() {
            @Override
            public void run() {
                // After the map is displayed for a short time, navigate to DoctorListActivity
                Intent intent = new Intent(HospitalLocationActivity.this,
DoctorListActivity.class);
                intent.putExtra("hospital", hospitalName); // Pass the selected hospital name
                startActivity(intent);
                finish(); // Finish the current activity to prevent the user from coming back
            }
        }, 30000); // 3 seconds delay
    }

    @Override
    public void onMapReady(@NonNull GoogleMap map) {
        googleMap = map;
        LatLng hospitalLocation = new LatLng(hospitalLat, hospitalLng);
        googleMap.addMarker(new
MarkerOptions().position(hospitalLocation).title(hospitalName));
        googleMap.moveCamera(CameraUpdateFactory.newLatLngZoom(hospitalLocation,
15));
    }

    private void openGoogleMapsNavigation(double latitude, double longitude) {
        // Create the URI to open Google Maps with the given coordinates
        String uri = "google.navigation:q=" + latitude + "," + longitude;
        Intent intent = new Intent(Intent.ACTION_VIEW, Uri.parse(uri));
        intent.setPackage("com.google.android.apps.maps");
        if (intent.resolveActivity(getPackageManager()) != null) {
            startActivity(intent);
        } else {
            // Handle the case where Google Maps is not installed
            Toast.makeText(this, "Google Maps is not installed.",
Toast.LENGTH_SHORT).show();
        }
    }

    @Override
    protected void onResume() {
        super.onResume();
        mapView.onResume();
    }

    @Override
    protected void onPause() {
        super.onPause();
        mapView.onPause();
    }

    @Override
    protected void onDestroy() {
        super.onDestroy();
        mapView.onDestroy();
    }

    @Override
    protected void onSaveInstanceState(@NonNull Bundle outState) {
        super.onSaveInstanceState(outState);
        Bundle mapViewBundle = outState.getBundle(MAPVIEW_BUNDLE_KEY);
        if (mapViewBundle == null) {
            mapViewBundle = new Bundle();
            outState.putBundle(MAPVIEW_BUNDLE_KEY, mapViewBundle);
        }
        mapView.onSaveInstanceState(mapViewBundle);
    }
}
```

BookAppointmentActivity.java

```
package com.example.app;

import android.content.Intent;

import android.os.Bundle;

import android.view.View;

import android.widget.Button;

import android.widget.EditText;

import android.widget.TextView;

import android.widget.Toast;



import androidx.appcompat.app.AppCompatActivity;



import java.text.SimpleDateFormat;

import java.util.Date;

import java.util.Locale;



public class BookAppointmentActivity extends AppCompatActivity {



    private String doctorName;

    private String specialization;
private String hospitalName;



@Override

protected void onCreate(Bundle savedInstanceState) {

    super.onCreate(savedInstanceState);

    setContentView(R.layout.activity_book_appointment);



    // Retrieve doctor's name, specialization, and hospital name from the intent

    doctorName = getIntent().getStringExtra("doctor");

    specialization = getIntent().getStringExtra("specialization");

    hospitalName = getIntent().getStringExtra("hospital");



    // Initialize views

    TextView tvDoctorDetails = findViewById(R.id.tvDoctorDetails);

    EditText etPatientName = findViewById(R.id.etPatientName);

    Button btnBookAppointment = findViewById(R.id.btnBookAppointment);



    // Display doctor's name and specialization

    tvDoctorDetails.setText("Doctor: " + doctorName + " (" + specialization + ")");



    // Handle button click for booking appointment
    btnBookAppointment.setOnClickListener(new View.OnClickListener() {
        @Override
        public void onClick(View v) {
            String patientName = etPatientName.getText().toString();


            if (patientName.isEmpty()) {
                Toast.makeText(BookAppointmentActivity.this, "Please enter patient name",
Toast.LENGTH_SHORT).show();
                return;
            }
            // Get current date and time
            String currentDateTime = new SimpleDateFormat("yyyy-MM-dd HH:mm:ss",
Locale.getDefault()).format(new Date());


            // Show booking confirmation message
            Toast.makeText(
                    BookAppointmentActivity.this,
                    "Appointment booked for " + patientName + "\nDoctor: " + doctorName +
"\nSpecialization: " + specialization + "\nHospital: " + hospitalName + "\nDate & Time: " +
currentDateTime,
                    Toast.LENGTH_LONG
            ).show();
            // Create an Intent to navigate to AppointmentSuccessActivity

            Intent intent = new Intent(BookAppointmentActivity.this, AppointmentSuccessActivity.class);

            intent.putExtra("doctor", doctorName);

            intent.putExtra("specialization", specialization);

            intent.putExtra("hospital", hospitalName);

            intent.putExtra("patientName", patientName);

            intent.putExtra("appointmentTime", currentDateTime);



            // Start the AppointmentSuccessActivity

            startActivity(intent);

            finish();  // Close the current activity

          }

        });

    }

}
```

AppointmentSuccessActivity.java

```
package com.example.app;

import android.os.Bundle;

import android.widget.TextView;



import androidx.appcompat.app.AppCompatActivity;
```

```
public class AppointmentSuccessActivity extends AppCompatActivity {


    @Override

    protected void onCreate(Bundle savedInstanceState) {

        super.onCreate(savedInstanceState);

        setContentView(R.layout.activity_appointment_success);


        // Get appointment details passed from the previous activity

        String doctorName = getIntent().getStringExtra("doctor");

        String specialization = getIntent().getStringExtra("specialization");

        String hospitalName = getIntent().getStringExtra("hospital");

        String patientName = getIntent().getStringExtra("patientName");

        String appointmentTime = getIntent().getStringExtra("appointmentTime");


        // Initialize views

        TextView tvSuccessMessage = findViewById(R.id.tvSuccessMessage);


        // Display the successful appointment details

        String successMessage = "Appointment Booked Successfully!\n\n" +

                "Patient: " + patientName + "\n" +
            "Doctor: " + doctorName + " (" + specialization + ")\n" +

            "Hospital: " + hospitalName + "\n" +

            "Date & Time: " + appointmentTime;



        tvSuccessMessage.setText(successMessage);

    }

}
```

APPENDIX-B

SCREENSHOTS

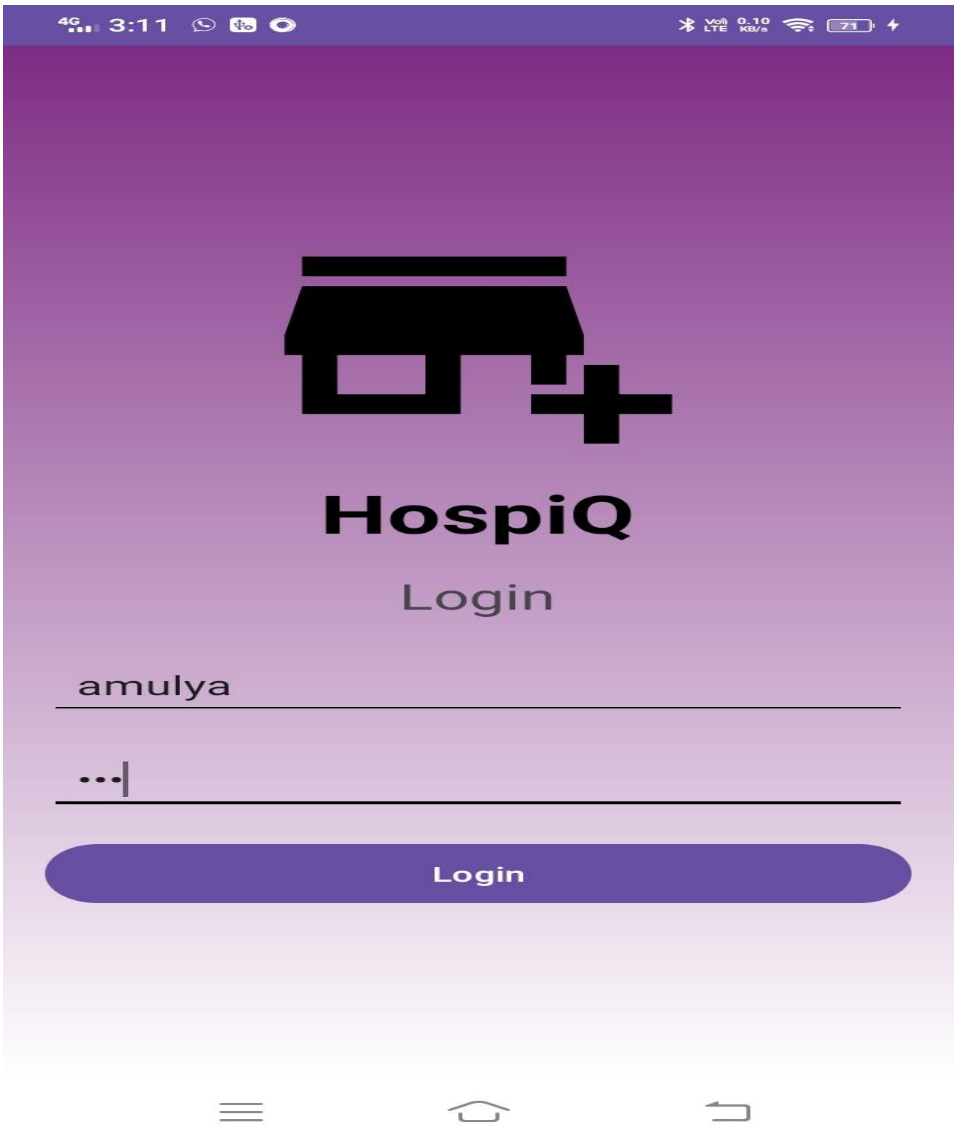
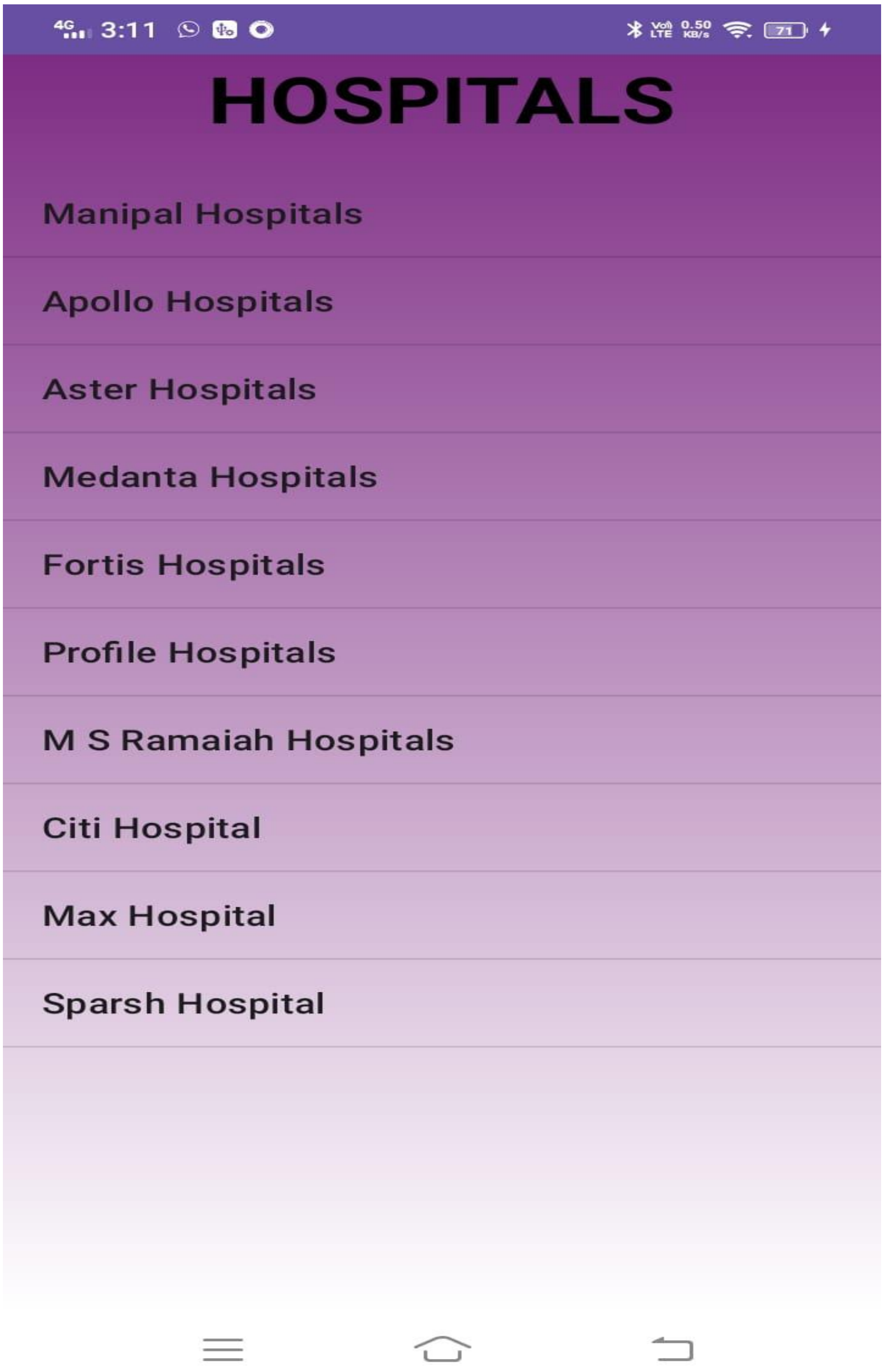


Fig 1.1 : User Login page



4G 3:11

HOSPITALS

Manipal Hospitals

Apollo Hospitals

Aster Hospitals

Medanta Hospitals

Fortis Hospitals

Profile Hospitals

M S Ramaiah Hospitals

Citi Hospital

Max Hospital

Sparsh Hospital

Fig 1.2 : List of Hospitals

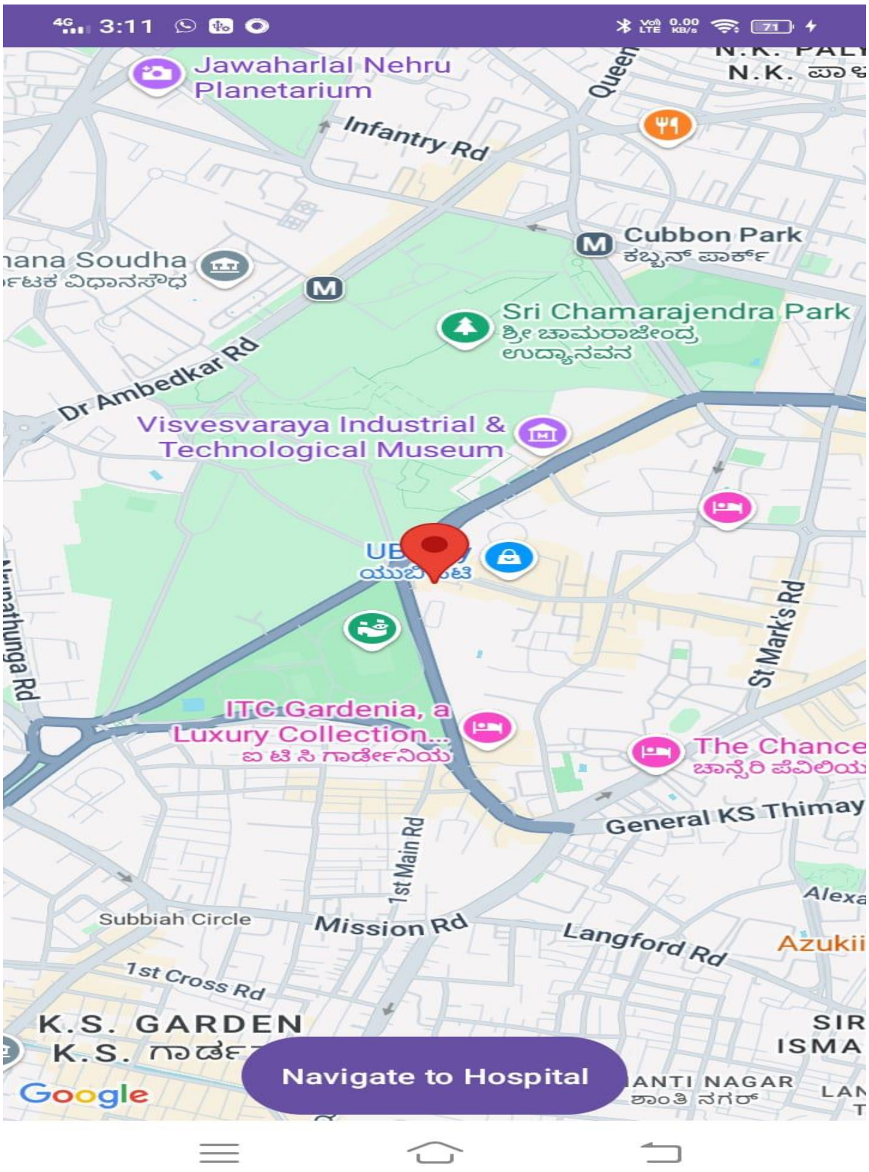
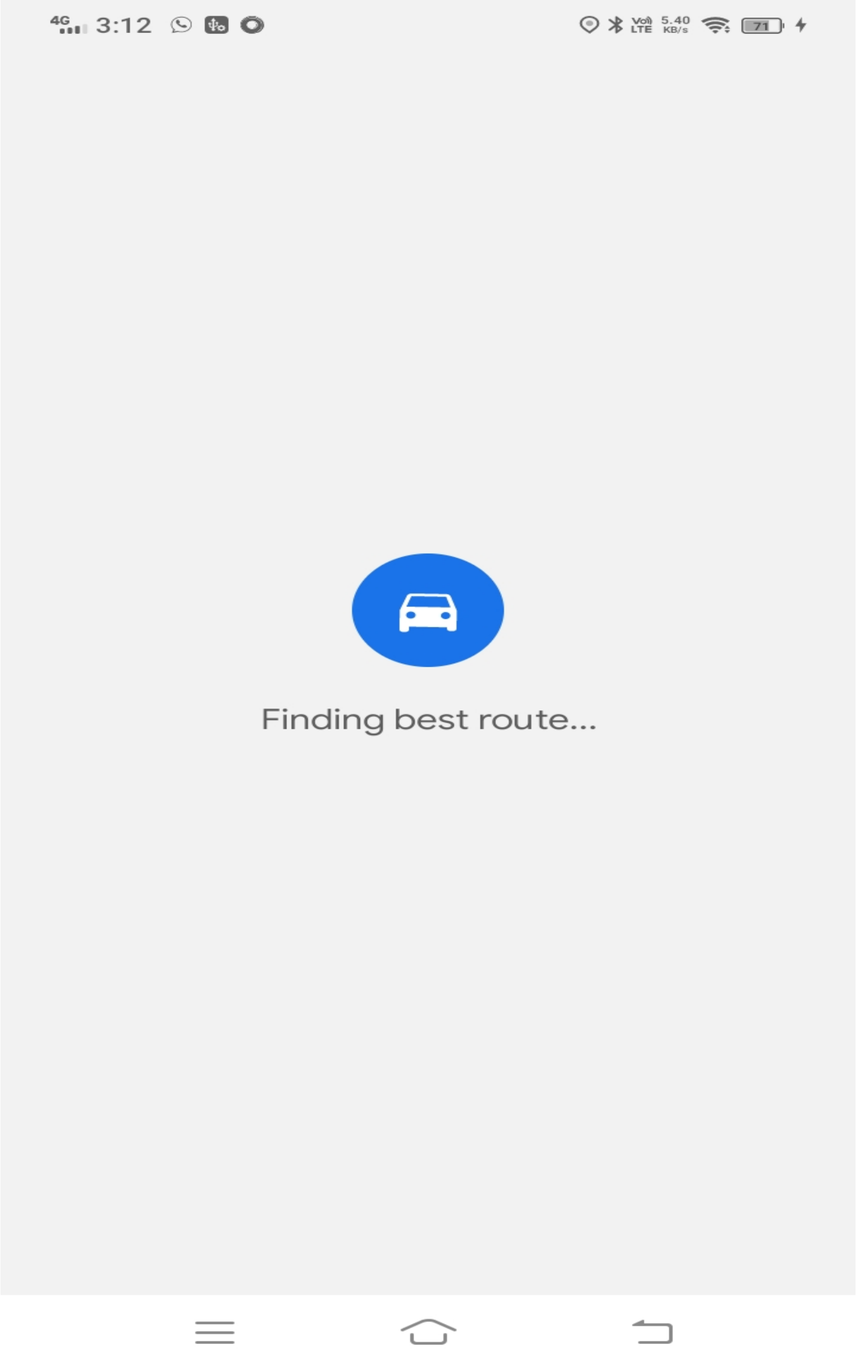
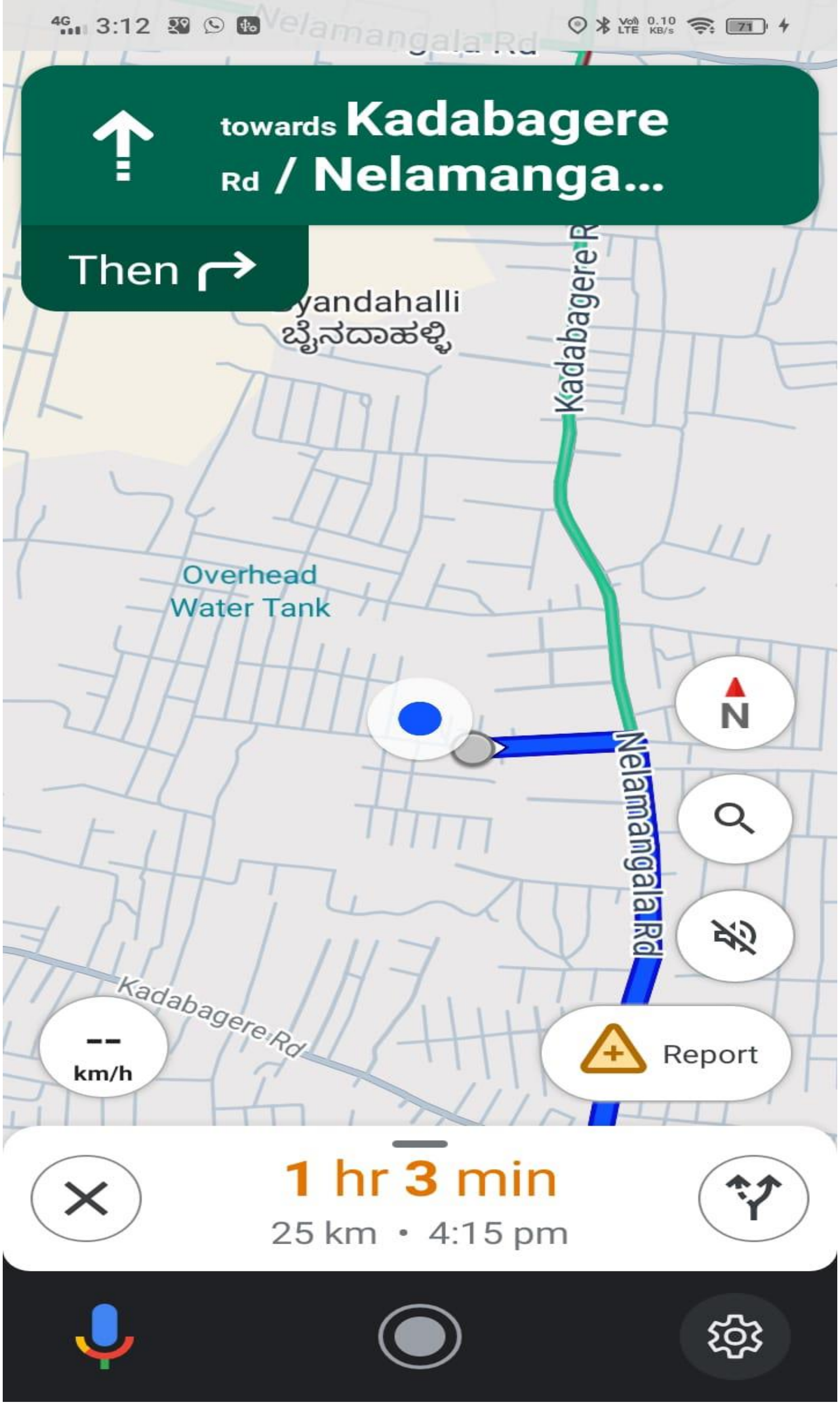


Fig 1.3 : Hospital Navigation



Finding best route...

Fig 1.4 : Finding the hospital route



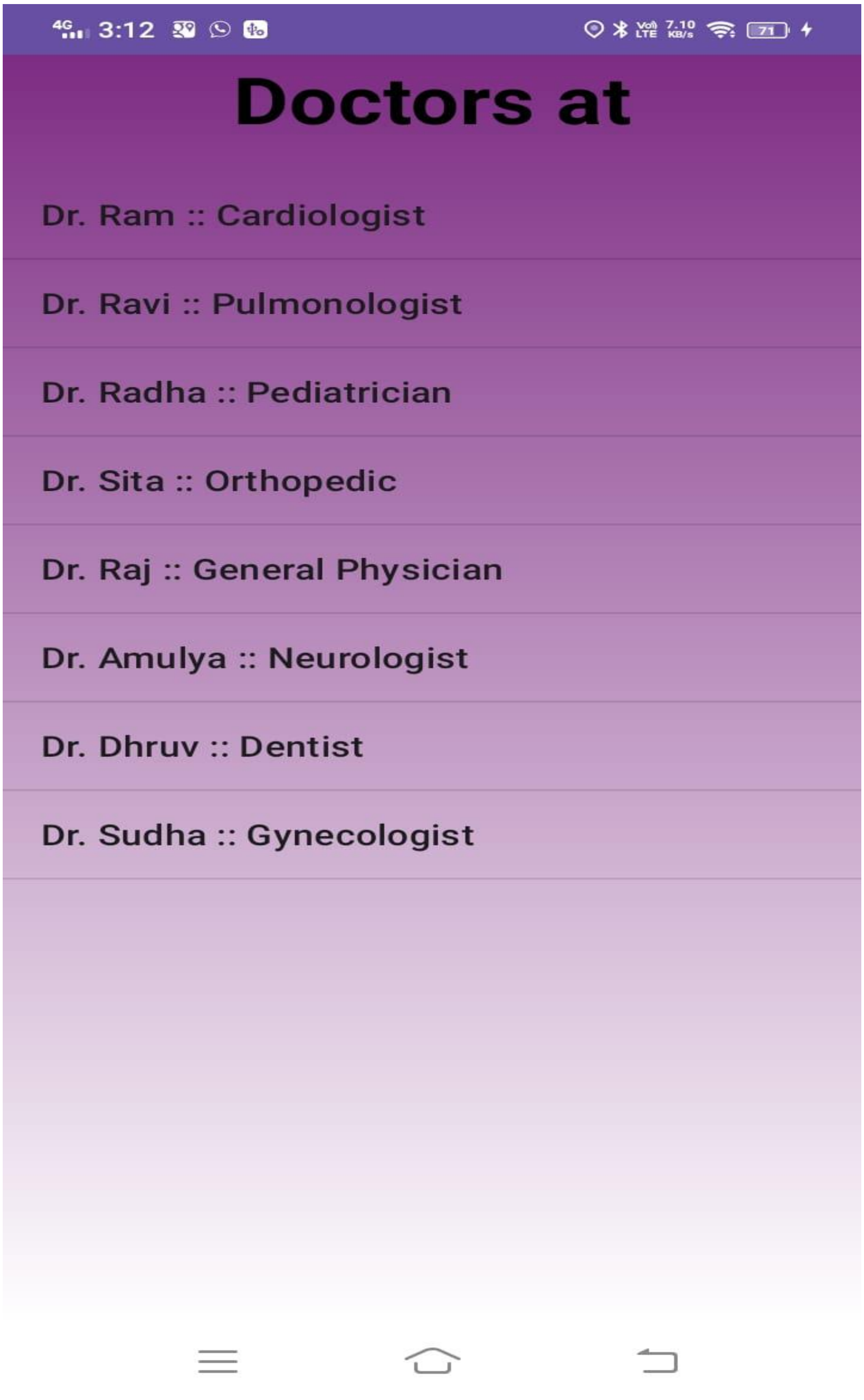
towards Kadabagere Rd / Nelamanga...

Then

1 hr 3 min

25 km • 4:15 pm

Fig 1.5 : Hospital Route Map



Doctors at

Dr. Ram :: Cardiologist

Dr. Ravi :: Pulmonologist

Dr. Radha :: Pediatrician

Dr. Sita :: Orthopedic

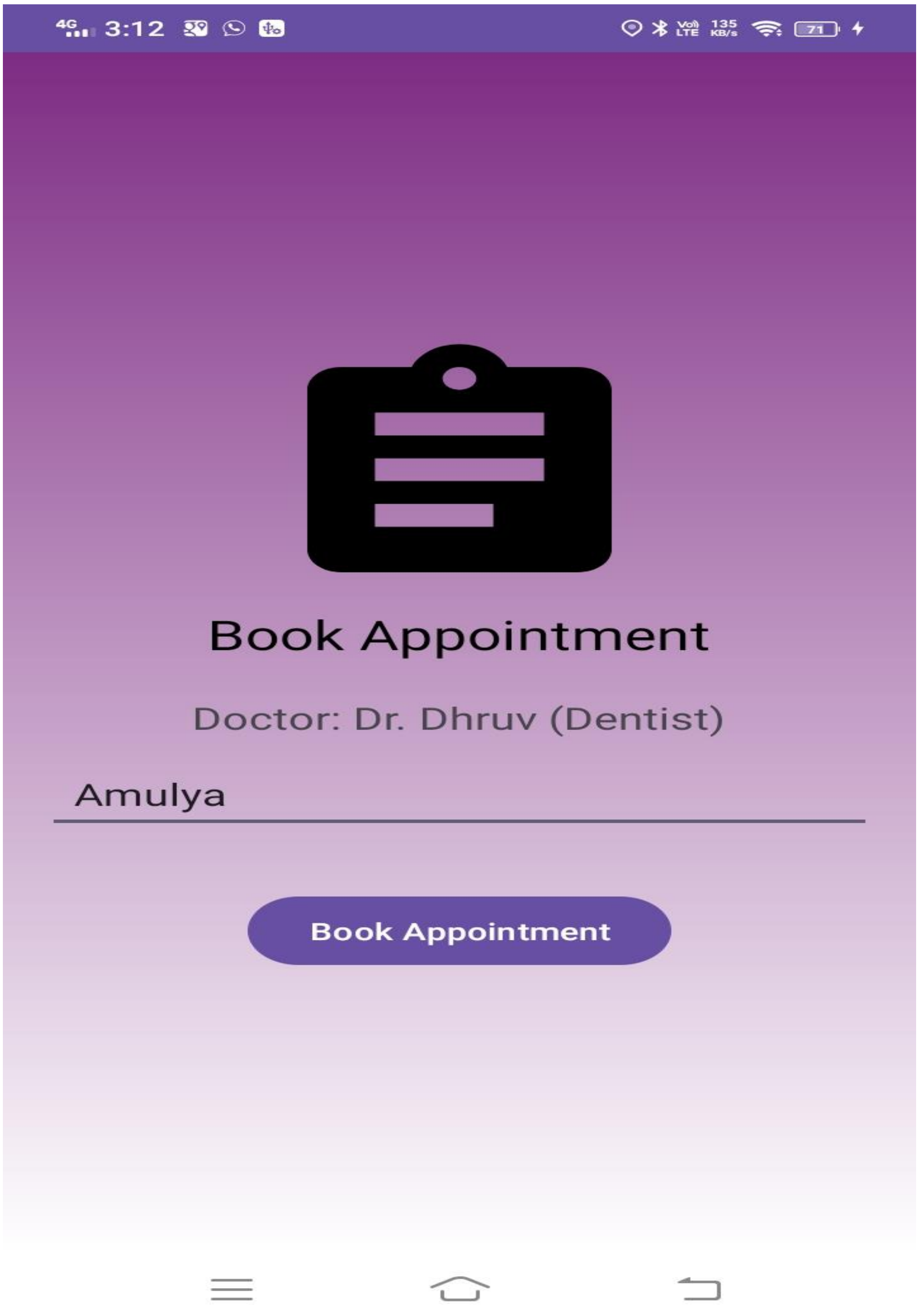
Dr. Raj :: General Physician

Dr. Amulya :: Neurologist

Dr. Dhruv :: Dentist

Dr. Sudha :: Gynecologist

Fig 1.6 : Doctors List



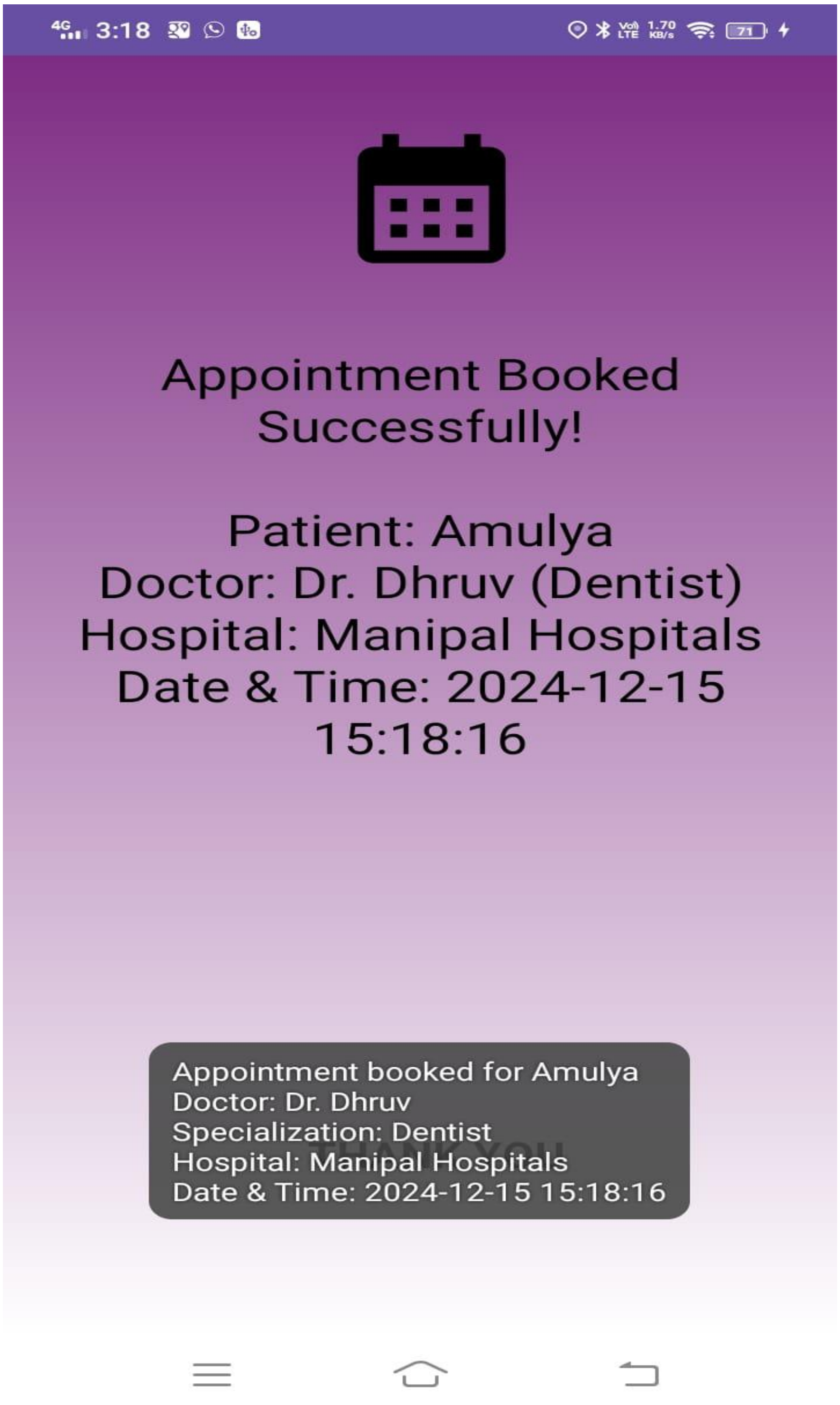
3:12

Book Appointment

Doctor: Dr. Dhruv (Dentist)

Amulya

Fig 1.7 : Appointment Booking

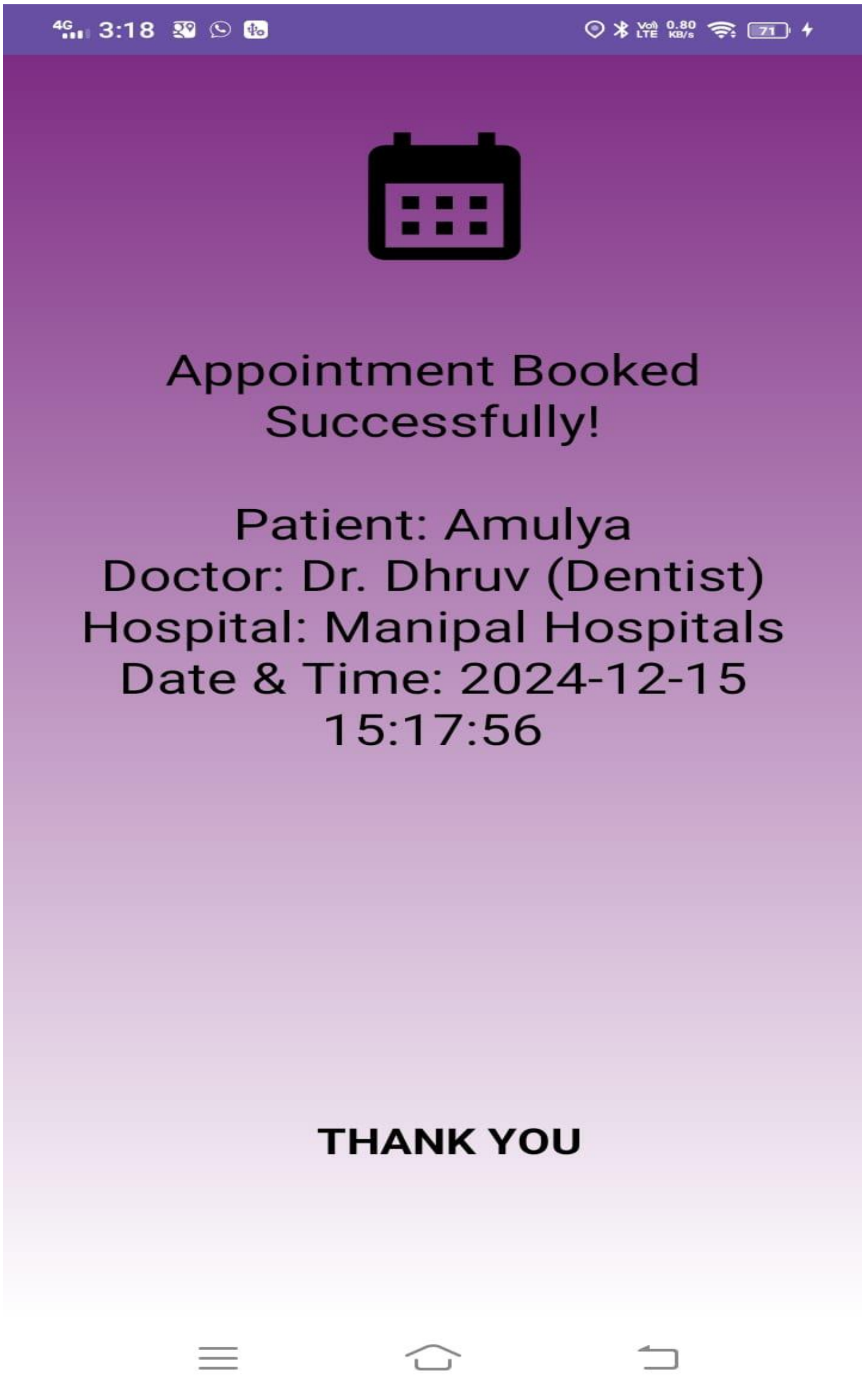


Appointment Booked Successfully!

Patient: Amulya
Doctor: Dr. Dhruv (Dentist)
Hospital: Manipal Hospitals
Date & Time: 2024-12-15 15:18:16

Appointment booked for Amulya
Doctor: Dr. Dhruv
Specialization: Dentist
Hospital: Manipal Hospitals
Date & Time: 2024-12-15 15:18:16

Fig 1.8 :Pop up message for Successful Booking



Appointment Booked Successfully!

Patient: Amulya
Doctor: Dr. Dhruv (Dentist)
Hospital: Manipal Hospitals
Date & Time: 2024-12-15 15:17:56

THANK YOU

Fig 1.9 : Confirmation page for Successful Appointment Booking

APPENDIX-C

ENCLOSURES

1. Journal publication/Conference Paper Presented Certificates of all students.

2. Include certificate(s) of any Achievement/Award won in any project-related event.

3. Similarity Index / Plagiarism Check report clearly showing the Percentage (%). No need for a page-wise explanation.

4. Details of mapping the project with the Sustainable Development Goals (SDGs).