

**MEDICALL**

**A Comprehensive Electronic Medical Record System using decision tree algorithm**

**PRIYANKA RAJBANSHI**

**TERM 2023/2024**

**FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY**

**LIMKOKWING UNIVERSITY OF CREATIVE TECHNOLOGY**

**APRIL 2023**







**BSC(HONS) IN INFORMATION TECHNOLOGY**

**BPROJ3501 Major Project 1**





Title : Medicall- A Comprehensive Electronic Medical Record System using decision tree algorithm

Submission Date : 2nd March 2024

Supervisor : Mr. Hikamt Saud

Student Name: Priyanka Rajbanshi

Student ID : 01202120

Term :2023/2024

THE PROJECT IS PREPARE FOR

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY

LIMKOKWING UNIVERSITY OF CREATIVE TECHNOLOGY

IN PARTIAL FULFILLMENT

FOR

BSC(HONS) IN INFORMATION TECHNOLOGY

FACULTY OF INFORMATION & COMMUNICATION TECHNOLOGY

LIMKOKWING UNIVERSITY OF CREATIVE TECHNOLOGY2023

# PERMISSION TO USE

In presenting this thesis in partial fulfillment of the requirements for undergraduate degree from Limkokwing University of Creative Technology, I agree that the University Library may make it freely available for inspection. I further agree that permission for copying of this thesis in any manner, in whole or in part, for scholarly purposes may be granted by my supervisor, in his absence, by the Dean of the faculty of Information; Communication Technology. It is understood that any copying or publication or use of this thesis or parts thereof for financial gain should not be allowed without my written permission. It is also understood that due recognition shall be given to me and to Limkokwing University of Creative Technology for any scholarly use which may be made of any material from my thesis.

Request for permission to copy or to make use of material in this thesis, in whole or in part should be addressed to:

Dean of the Faculty of Information &amp; Communication Technology

Limkokwing University of Creative Technology

Jalan Teknokrat 1/1

63000 Cyberjaya

Selangor Darul Ehsan

# ACKNOWLEDGEMENT

I would like to express my deepest gratitude to all those who have contributed to the completion of this project. Firstly, I would like to thank my supervisor, Hikmat Saud, for his guidance, support, and valuable insights throughout the development process. Their expertise and encouragement were instrumental in shaping the direction of this project.

I would also like to thank my teammates for their collaboration and dedication. Their hard work and commitment to excellence have been inspiring, and I am grateful for the opportunity to work alongside such talented individuals.

Last but not least, I would like to thank my family and friends for their unwavering support and understanding during this project. Their encouragement and belief in me kept me motivated even during the most challenging times.

Thank you to everyone who has been a part of this journey. Your contributions have been invaluable, and I am truly grateful for your support.

# 

# ABSTRACT

MediCall is a revolutionary electronic medical record (EMR) system tailored to the unique challenges of the Nepalese healthcare industry. In an era where ineffective paper-based records are common, MediCall stands out as an example of effectiveness, accessibility, and patient privacy. This cutting-edge system makes it simple for consumers to view their medical reports and makes it easy for doctors to enter vital data like prescriptions and symptoms. MediCall is unique because it incorporates state-of-the-art technology, specifically a dynamic decision tree algorithm. This algorithm rings in a new era of precision healthcare while also making disease prediction based on symptoms easier.

The document offers a thorough analysis of MediCall's architecture, features, and strict security protocols. It also highlights the implementation strategy and design of the system.

# TABLE OF CONTENTS

[**PERMISSION TO USE**](#_30j0zll) **i**

[**Abstract**](#_1fob9te) **ii**

[**Table of figures**](#_3znysh7) **v**

[**Chapter 1: Introduction 1**](#_2et92p0)

[1.1 Introduction 1](#_tyjcwt)

[1.2 Overview 2](#_3dy6vkm)

[1.3 Problem statement 3](#_1t3h5sf)

[1.4 Project objective 4](#_2s8eyo1)

[1.5 Project scope 5](#_17dp8vu)

[1.6 Project justification 6](#_3rdcrjn)

[1.7 Project constraint 6](#_26in1rg)

[1.8 Project timeline 7](#_lnxbz9)

[**Chapter 2: Literature review 9**](#_44sinio)

[2.1 Introduction 9](#_2jxsxqh)

[2.2 The Related System 9](#_z337ya)

[2.3 Research on Electronic Medical Recording Systems 11](#_3j2qqm3)

[2.4 Related technique and technology 12](#_1y810tw)

[**Chapter 3: Methodology 15**](#_4i7ojhp)

[3.1 Introduction 15](#_2xcytpi)

[3.2 Development Lifecycle Methodology Used in Project 16](#_1ci93xb)

[3.3 Justification of Methodology 18](#_2bn6wsx)

[3.4 Hardware and Software Requirements 19](#_qsh70q)

[**Chapter 4: Requirement analysis 21**](#_3as4poj)

[4.1 Introduction 21](#_1pxezwc)

[4.2 Gathering and Analysis of Requirements 21](#_2p2csry)

[4.2.1 Gathering of the requirements 21](#_147n2zr)

[4.2.2 Analysis of the data gathered 22](#_3o7alnk)

[4.3 Functional Requirements 22](#_ihv636)

[4.4 Non-Functional Requirements 23](#_32hioqz)

[4.5 Interface Requirements 24](#_1hmsyys)

[4.6 Conclusion 24](#_2grqrue)

[**Chapter 5: Design 25**](#_vx1227)

[5.1 Introduction 25](#_3fwokq0)

[5.2 Conceptual Architecture 25](#_4f1mdlm)

[5.3 Program Design 28](#_19c6y18)

[5.3.1 Conceptual Diagram: 29](#_28h4qwu)

[5.3.2 Sequence Diagram: 30](#_nmf14n)

[5.3.3 Use Case Diagram: 31](#_1mrcu09)

[5.4 Interface Design 32](#_2lwamvv)

[5.5 Database Design 36](#_3l18frh)

[**Chapter 6: Implementation 39**](#_4k668n3)

[**Chapter 7: Testing 39**](#_2zbgiuw)

[**Chapter 8: Conclusion 39**](#_1egqt2p)

[8.1 Limitations of Medicall 39](#_2dlolyb)

[8.2 Key Findings 40](#_sqyw64)

[8.3 Future Considerations 40](#_1rvwp1q)

[8.4: Summary 41](#_4bvk7pj)

[**Reference 43**](#_3q5sasy)

[**Appendix 44**](#_25b2l0r)

# LIST OF FIGURES

Figure 1: project timeline (Gantt Chart) 5

Figure 2: related system 11

Figure 3: analysis of "Which features do you think are essential in a medical history recording system? (Select all that apply)” 22

Figure 4: decision tree 25

Figure 5: conceptual diagram 26

Figure 6: sequence diagram 32

Figure 7: Interface design of Medicall 31

Figure 8: ER Diagram 32

**List of Abbreviation**

Decision Tree Algorithm

# Chapter 1: Introduction

## 1.1 Overview

MediCall is a comprehensive electronic medical record system that aims to address the

challenges faced by the Nepalese healthcare industry due to the lack of a

comprehensive electronic medical record system. Traditional paper-based records are

inefficient, often inaccessible during emergencies, and make tracking patient medical

history challenging. Moreover, paper-based records can compromise the confidentiality

and privacy of patient information.

To address these issues, MediCall provides a user-friendly, secure, and comprehensive electronic medical record system tailored to the Nepalese healthcare industry. The app includes patient, doctor, and admin panels and is equipped with various features, including recording patient demographics, medical history, allergies, medications, laboratory results, and imaging studies.

Additionally, MediCall uses advanced data encryption and security protocols to protect patient information from unauthorized access and data breaches. By improving accessibility, communication, and efficiency while enhancing patient safety and privacy, MediCall offers a cost-effective solution for healthcare facilities, allowing healthcare providers to focus on delivering high-quality care to their patients.

This report outlines the design and implementation specification deployed for Medicall, an advanced health records management system that uses decision tree algorithms to assist healthcare professionals in making accurate and efficient diagnosis and treatment decisions based on patient data. With its cross-platform design and user-friendly interface, Medicall aims to revolutionize the healthcare industry by providing a reliable and efficient tool for healthcare professionals to manage patient records and make informed decisions.

This report provides a comprehensive overview of the key features of Medicall, including its architecture, functionality, and security measures. It also outlines the development process, including the methodology, tools, and technologies used to build the system. To ensure that Medicall meets the needs and expectations of end-users, a survey was conducted to gather feedback from potential users on their preferences and needs regarding medical history recording systems. The survey results were analyzed and used to inform the design and development of Medicall.

Overall, this report presents a detailed description of the development procedure of Medicall, with the goal of comprehending development process of reliable, user-friendly, and efficient tool for healthcare professionals to manage patient records and provide high-quality care.

## 1.2 Problem statement

The healthcare industry is faced with the challenge of managing an ever-increasing amount of patient data, which is often stored in a variety of formats and locations. This makes it difficult for healthcare professionals to access and analyze patient data, leading to inefficiencies, errors, and delays in treatment.

Additionally, the lack of a standardized approach to managing patient data can lead to privacy and security issues. Current medical history recording systems often suffer from a lack of interoperability, making it difficult to exchange data between different systems. This results in data silos that are inaccessible to other healthcare providers, hindering collaboration and coordination of care. Furthermore, the user interface of many medical history recording systems is often complex and confusing, leading to user frustration and errors.

To address these challenges, there is a need for an advanced health records management system that utilizes advanced technologies such as decision tree algorithms to assist clinicians in making accurate and efficient treatment decisions based on patient data. Additionally, such a system should have a cross-platform design that allows for easy accessibility, while ensuring the security and privacy of patient data.

## 

## 1.3 Project objective

The scope of this project is to design and develop Medicall, an advanced health records management system that utilizes decision tree algorithms to assist healthcare professionals in making accurate and efficient treatment decisions based on patient data. The key components of the project include:

* User Interface: Development of a user-friendly interface for healthcare professionals to easily navigate and manage patient records. The interface should be accessible on mobile, desktop, and web platforms.
* Decision Tree Algorithm: Development of a decision tree algorithm that analyzes patient data and provides treatment recommendations based on the patient's medical history.
* Data Security: Ensure the security and privacy of patient data through the implementation of robust data protection measures and compliance with relevant regulations and standards.
* Survey: Conducting a survey to gather feedback from potential users on their preferences and needs regarding medical history recording systems, and incorporate this feedback into the design and development of Medicall.
* Agile Development Methodology: Use of an agile development methodology to ensure continuous improvement and adaptability to changing user needs and technological advancements.

The project involves collaboration with healthcare professionals and potential users to gather requirements, feedback, and testing of the application. The project also involves compliance with relevant regulations and standards to ensure the privacy and security of patient data. The project scope may be expanded or modified based on the findings of the survey and the needs of the healthcare industry.

## 1.4 Project scope

The project scope includes the following key components:

* User Interface: Development of a user-friendly interface for healthcare professionals to easily navigate and manage patient records. The interface should be accessible on mobile, desktop, and web platforms.
* Decision Tree Algorithm: Development of a decision tree algorithm that analyzes patient data using symptoms and helps to identify the disease.
* Data Security: Ensuring the security and privacy of patient data through the implementation of robust data protection measures and compliance with relevant regulations and standards.
* Survey: Conducting a survey to gather feedback from potential users on their preferences and needs regarding medical history recording systems and incorporating this feedback into the design and development of the system.
* Agile Development Methodology: Use of an agile development methodology to ensure continuous improvement and adaptability to changing user needs and technological advancements.

## 1.5 Project justification

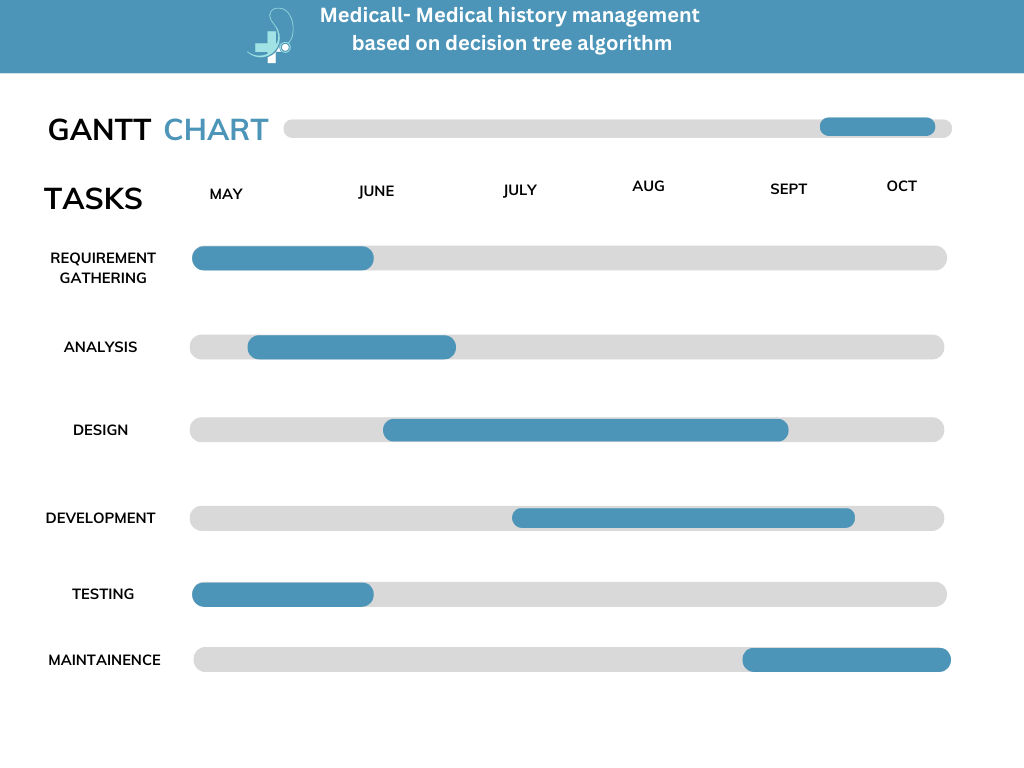
The developed system addresses the need for an advanced health records management system that can assist healthcare professionals in making accurate and efficient treatment decisions based on patient data. Current systems often rely on manual data entry, which can be time-consuming and prone to errors. The implementation of a decision tree algorithm can significantly reduce the time and effort required to input and analyze data, resulting in more accurate and efficient treatment decisions. The survey will ensure that the system meets the needs and preferences of potential users and can effectively improve patient care.

## 1.6 Project constraint

The project constraints include the following:

* Compliance with relevant regulations and standards regarding patient data privacy and security.
* Limited budget for development and implementation.
* Limited timeframe for completion of the project.
* Availability of healthcare professionals and potential users for collaboration and testing.

## 1.7 Project timeline



### Figure 1: project timeline (Gantt Chart)

The timeline for this project is divided into several phases, each with its own set of tasks and deliverables. The following is a tentative timeline for the project:

* Phase 1: Research and Planning (2 weeks)
* Conduct a literature review
* Develop project objectives and scope
* Identify project constraints
* Develop a project timeline
* Phase 2: Design and Development (12 weeks)
* Develop system architecture
* Implement decision tree algorithms
* Develop user interface and data storage
* Conduct testing and debugging
* Phase 3: Deployment and Testing (3 weeks)
* Deploy the system on chosen platforms
* Conduct testing and debugging
* Finalize system documentation
* Phase 4: Maintenance and Support (2 week)
* Provide ongoing maintenance and support to ensure that the application is up-to-date, secure, and bug-free
* Continuously improve the application based on user feedback and changing requirements

# Chapter 2: Literature Review

## 2.1 Introduction

The healthcare industry is one of the most data-intensive sectors, and the effective management of this data is critical for improving patient care and outcomes. The introduction of electronic health records (EHRs) has made it possible to store and manage patient information more efficiently, providing clinicians with real-time access to patient data. However, the implementation of EHRs is not without its challenges, such as interoperability, data privacy and security, and user adoption. In this chapter, we will examine the existing literature related to the development of health records management systems and the various approaches used to address these challenges.

## 2.2 The Related System

There are several health records management systems available in the market, including both desktop and cloud-based solutions. Some of the popular health records management systems are Epic, Cerner, Allscripts, and MEDITECH. These systems are designed to help healthcare providers manage patient information, streamline workflows, and improve patient outcomes.

Epic is a widely used electronic health records (EHR) system that helps healthcare providers manage patient data, including medical history, diagnoses, treatment plans, and test results. The system is designed to be user-friendly and customizable, allowing providers to tailor the system to their specific needs. However, the system can be complex and difficult to use, which can lead to errors and inefficiencies.

Cerner is another popular EHR system that helps healthcare providers manage patient data. The system includes features such as clinical decision support, order management, and medication management. Cerner is designed to be scalable and adaptable, making it a good choice for healthcare organizations of all sizes. However, like Epic, the system can be complex and difficult to use.

* Allscripts is another EHR system that is designed to help healthcare providers manage patient data. The system includes features such as medication management, clinical decision support, and patient engagement tools. Allscripts is designed to be easy to use and customizable, making it a good choice for healthcare providers who want a system that is tailored to their specific needs.
* MEDITECH is a cloud-based EHR system that helps healthcare providers manage patient data. The system includes features such as medication management, clinical decision support, and order management. MEDITECH is designed to be user-friendly and customizable, making it a good choice for healthcare providers who want a system that is easy to use and tailored to their specific needs.
* Despite the availability of these health records management systems, clinicians still face several challenges in using these systems. One of the biggest challenges is the lack of interoperability between systems, which makes it difficult for providers to access patient data from other providers. This can lead to errors and inefficiencies in patient care.
* Another challenge is the complexity of these systems, which can make it difficult for providers to find the information they need quickly. This can lead to delays in patient care and lower patient satisfaction. Additionally, these systems can be expensive to implement and maintain, which can be a barrier to adoption for smaller healthcare organizations.

In this project, we aim to address these challenges by developing an advanced health records management system that utilizes decision tree algorithms to assist clinicians in making accurate and efficient treatment decisions based on patient data. Our system will be designed to be user-friendly, customizable, and scalable, making it a good choice for healthcare providers of all sizes. Additionally, our system will be designed to be interoperable with other systems, making it easy for providers to access patient data from other providers.

## 2.3 Research on Electronic Medical Recording Systems

The importance of electronic medical record systems in improving healthcare quality, safety, and efficiency has been recognized in various studies. The Agency for Healthcare Research and Quality (AHRQ) defines electronic medical record (EMR) systems as electronic records of health-related information that can be created, gathered, managed, and consulted by authorized clinicians and staff within a healthcare organization. These systems have the potential to facilitate workflow, enhance patient care, and improve patient safety.

In the context of Nepal, there has been a growing awareness of the need for a national electronic medical record system. An article from Online Khabar highlights the necessity of such a system and discusses potential implementation strategies. The research report evaluating the electronic medical recording system in Trisuli District Hospital, Nuwakot in 2018 provides insights into the practical implementation and impact of an electronic medical record system in a specific healthcare setting in Nepal.

By leveraging the knowledge gained from these research studies, MediCall aims to provide a comprehensive electronic medical record system that addresses the specific challenges faced by the Nepalese healthcare industry. The system's features, including patient demographics, medical history, allergies, medications, laboratory results, and imaging studies, are designed to streamline the process of collecting and maintaining patient medical information, ultimately improving patient outcomes and enhancing the overall efficiency of healthcare delivery.

## 2.4 Related technique and technology

The development of health records management systems involves the use of various technologies and techniques. This section provides an overview of some of the most commonly used technologies and techniques in the development of these systems.

● Artificial Intelligence (AI): Artificial intelligence (AI) is a branch of computer science that focuses on the development of intelligent machines that can perform tasks that typically require human intelligence, such as visual perception, speech recognition, decision-making, and language translation. In the context of health records management systems, AI can be used to analyze large amounts of data to identify patterns and trends that can help clinicians make more informed decisions about patient care. Machine learning, a subset of AI, involves the development of algorithms that enable machines to learn from data without being explicitly programmed.

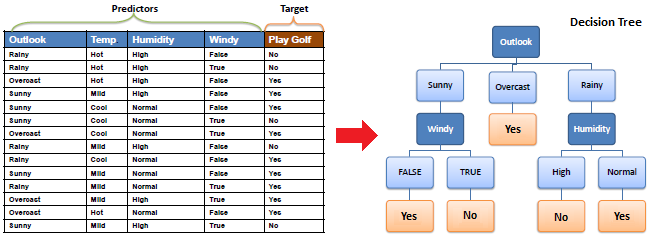
● Decision Trees: A decision tree is a tree-like model of decisions and their possible consequences, including chance event outcomes, resource costs, and utility. In the context of health records management systems, decision trees can be used to help clinicians make treatment decisions based on patient data.

● Agile Methodology: Agile methodology is an iterative approach to software development that emphasizes flexibility and collaboration between cross-functional teams. In the context of health records management systems, agile methodology can be used to quickly develop and improve software based on user feedback.

● Cloud Computing: Cloud computing refers to the delivery of computing services, including data storage, servers, and software, over the internet. In the context of health records management systems, cloud computing can be used to provide secure and scalable data storage solutions.

● Programming Languages and Frameworks: Various programming languages and frameworks are used in the development of health records management systems. Popular programming languages include Java, Python, and C++, while popular frameworks include React, Angular, and Vue.js. The choice of programming language and framework depends on the specific requirements of the project.

● Security: Security is a critical concern in the development of health records management systems. Patient data is highly sensitive and must be protected from unauthorized access. Various security measures, such as encryption, access controls, and firewalls, can be implemented to ensure the security of patient data. Additionally, compliance with regulations such as HIPAA is essential to ensure the privacy and security of patient data.



### fig 2: Related technology: Using decision tree for data mining

# Chapter 3: Methodology

## 3.1 Introduction

The object-oriented approach to software development serves as the methodology for the development of MediCall. This methodology emphasizes designing and implementing software systems based on objects, which are instances of classes that encapsulate data and behaviors. The object-oriented approach offers a modular and reusable structure, making it a widely adopted methodology in software development for its ability to model real-world entities and their interactions.

By adopting the object-oriented methodology for the development of MediCall, the project aims to reap several advantages. One of the key benefits is the ability to create modular and maintainable code. The system can be divided into smaller, self-contained objects that can be independently developed, tested, and modified. This promotes code reusability and simplifies the process of maintaining and updating the system.

Additionally, the object-oriented approach facilitates encapsulation, inheritance, and polymorphism within the system. Encapsulation ensures that the internal implementation details of objects are hidden, providing a clear and defined interface for interaction. Inheritance allows for the creation of hierarchical relationships between classes, enabling code reuse and enhancing the overall design flexibility. Polymorphism enables objects of different classes to be treated interchangeably, enhancing flexibility and extensibility within the system.

In summary, the object-oriented methodology provides a structured and modular approach to software development. By embracing this methodology for MediCall, the project aims to leverage the advantages of code reusability, maintainability, and flexibility. The system will be designed and implemented using objects, allowing for the modeling of real-world entities and promoting a clear and understandable system architecture.

## 3.2 Development Lifecycle Methodology Used in Project

The chosen development lifecycle methodology for MediCall is the Agile methodology. Agile is an iterative and incremental approach to software development that emphasizes flexibility, collaboration, and adaptability. It promotes adaptive planning and encourages frequent feedback and iterations throughout the development process.

The development lifecycle in MediCall will adhere to the following Agile phases:

Requirements Gathering: In this initial phase, the requirements for MediCall will be gathered from various stakeholders, including healthcare professionals, administrators, and patients. This involves understanding their needs, expectations, and desired functionalities for the system. The requirements will be documented and prioritized to guide the development process.

User Story Creation: Based on the gathered requirements, user stories will be created to capture the specific features and functionalities of MediCall from the user's perspective. User stories describe the interaction between the user and the system, focusing on the value delivered to the user.

Sprint Planning: The development work will be organized into time-boxed iterations called sprints. During sprint planning, the development team, including developers, designers, and testers, will collaborate to determine the scope of work for the upcoming sprint. User stories will be selected from the product backlog and broken down into smaller, manageable tasks.

Development and Testing: The development team will work on implementing the selected user stories and tasks during the sprint. Developers will write code, designers will create the user interface, and testers will conduct continuous testing throughout the development process. The team will aim to deliver a potentially shippable product increment at the end of each sprint.

Daily Stand-up Meetings: Daily stand-up meetings will be conducted to promote communication and collaboration within the development team. Team members will share their progress, discuss any challenges or roadblocks, and coordinate their work for the day. These short meetings ensure transparency and alignment among team members.

Sprint Review: At the end of each sprint, a sprint review meeting will be held to showcase the completed work to stakeholders and gather feedback. The development team will demonstrate the new features and functionalities implemented during the sprint, allowing stakeholders to provide input and suggest improvements.

Sprint Retrospective: Following the sprint review, a sprint retrospective meeting will be conducted to reflect on the sprint's successes and challenges. The development team will discuss what went well, identify areas for improvement, and make adjustments to their processes and practices for future sprints.

Continuous Integration and Deployment: Throughout the development process, continuous integration and deployment practices will be employed. This involves integrating new code changes regularly, running automated tests, and deploying the latest version of the software to a staging or production environment. This allows for early detection of issues and ensures a stable and reliable software build.

Ongoing Iterations: The Agile development process will continue with subsequent sprints, each focusing on delivering incremental value to the system. User feedback and stakeholder input will be continuously incorporated into the development process, allowing for flexibility and adaptability as the project progresses.

The Agile methodology provides a flexible and collaborative approach for the development of MediCall. It allows for frequent iterations, feedback, and adjustments, ensuring that the system meets the evolving needs of the healthcare industry and its stakeholders.

## 

## 3.3 Justification of Methodology

The Agile methodology has been chosen for the development of MediCall due to its inherent flexibility, collaborative nature, iterative approach, stakeholder involvement, and ability to adapt to changing requirements.

Agile allows for continuous feedback and iterations, enabling the development team to respond to evolving needs and priorities in the healthcare industry. By involving stakeholders throughout the development process, MediCall can ensure that their requirements and expectations are addressed effectively. The iterative nature of Agile allows for frequent inspection and adaptation, reducing the risk of potential issues going unnoticed until later stages of development.

Moreover, Agile promotes a collaborative environment, fostering effective communication and teamwork among developers, designers, testers, and stakeholders. By embracing an Agile mindset, MediCall can encourage regular interaction and collaboration, leading to a better understanding of the system's goals and functionalities.

Additionally, the Agile methodology provides a framework for managing risks and uncertainties inherent in software development projects. Through the continuous integration of feedback and ongoing testing, potential risks and issues can be identified and addressed early on, minimizing their impact on the final product.

Overall, the Agile methodology justifies its selection for MediCall by offering a flexible, collaborative, and adaptive approach that aligns with the dynamic nature of the healthcare industry and its evolving requirements.

## 3.4 Hardware and Software Requirements

To develop MediCall, the following hardware and software requirements are needed:

**Hardware:**

A computer with a minimum of 8 GB RAM

100 GB of available hard disk space

A multicore processor

**Software:**

Operating System: Windows 10 or macOS Big Sur

Integrated Development Environment (IDE): Visual Studio CodeBackend Development: Frontend Development: HTML5, CSS3, and JavaScript

These hardware and software requirements provide a foundation for the development of MediCall and support the necessary tools and technologies for building a robust and user-friendly healthcare software solution.

# Chapter 4: Requirement Analysis

## 4.1 Introduction

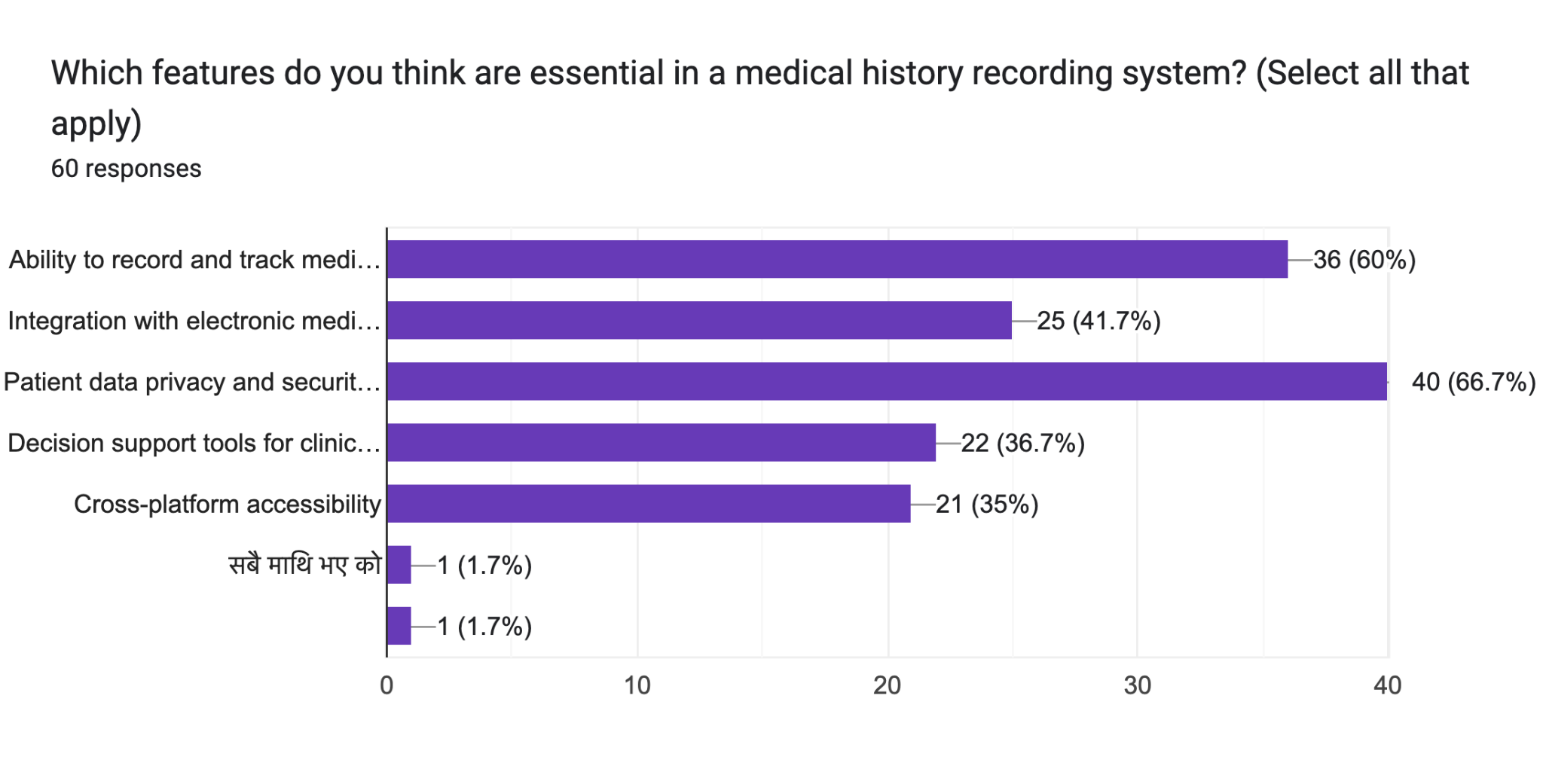
The requirement analysis is a crucial stage in the development of any software project, including MediCall. It plays a vital role in identifying and analyzing the requirements of the project stakeholders, ensuring that the developed software meets their expectations and addresses their needs. This document outlines the requirements analysis for the MediCall project, a healthcare management platform.

## 

## 4.2 Gathering and Analysis of Requirements

To gather the requirements for the MediCall project, various methods were employed, including surveys, interviews, and consultations with healthcare professionals and stakeholders. The stakeholders for this project include healthcare providers, administrators, doctors, nurses, and patients. The gathered requirements were then analyzed to identify functional, non-functional, and interface requirements. The following were some of questions asked in the survey:

1. Chose your role while using this system
2. How often do you interact with medical history recording systems?
3. What challenges have you faced with current medical history recording systems?
4. How comfortable are you with the use of technology in medical settings?
5. Which features do you think are essential in a medical history recording system?
6. How user-friendly do you want the system to be, and what features do you think would make it easier to use?
7. How important is data security in a medical history recording system, and what steps can be taken to ensure the security of patient data?
8. How would you like 'patient data' to input into the system ?



### fig 3 for: Which features do you think are essential in a medical history recording system? (Select all that apply)

## 4.3 Functional Requirements

1. User registration and login system to access the MediCall platform.
2. Patient management system to store and manage patient records, including personal information, medical history.
3. Electronic health record (EHR) management to securely store and retrieve patient health records.
4. Prescription and medication management system for healthcare providers to prescribe and manage medications for patients.
5. Billing and invoicing system to handle financial transactions and generate invoices for services rendered.
6. Reporting and analytics functionality to generate insights and reports on patient data, appointment statistics, and financial performance.

## 4.4 Non-Functional Requirements

1. Security and Privacy: The MediCall platform should ensure the confidentiality, integrity, and availability of patient data, complying with relevant data protection regulations.
2. Performance: The platform should have fast response times, minimal latency, and be able to handle a large number of concurrent users.
3. Scalability: The platform should be scalable to accommodate the growing user base and increasing data volume.
4. Usability: The platform should have an intuitive and user-friendly interface, making it easy for healthcare providers and patients to navigate and perform tasks.
5. Reliability: The platform should have a high level of reliability, minimizing downtime and ensuring continuous access to critical features.

## 4.5 Interface Requirements

1. User-friendly Interface: The MediCall platform should have a clean and intuitive interface, with easy navigation and clear presentation of information.
2. Responsive Design: The platform should be compatible with different devices (desktop, tablets, mobile devices) and screen sizes, providing a consistent user experience.
3. Integration with Existing Systems: The platform should have the capability to integrate with existing healthcare systems (e.g., laboratory systems, imaging systems) for seamless data exchange and interoperability.

## 

## 4.6 Conclusion

In conclusion, the requirements analysis for the MediCall project has identified functional, non-functional, and interface requirements for the healthcare management platform. These requirements will serve as a foundation for the development process, ensuring that the final product meets the needs of healthcare providers and patients while maintaining high standards of security, performance, and usability.

# Chapter 5: Design

## 5.1 Introduction

The design phase of the MediCall project is crucial for transforming system requirements into a well-structured and efficient healthcare management solution. This chapter focuses on the design aspects of the project, including the conceptual architecture and program design. The goal is to create a design that is modular, scalable, and user-friendly, ensuring the successful implementation of the system.

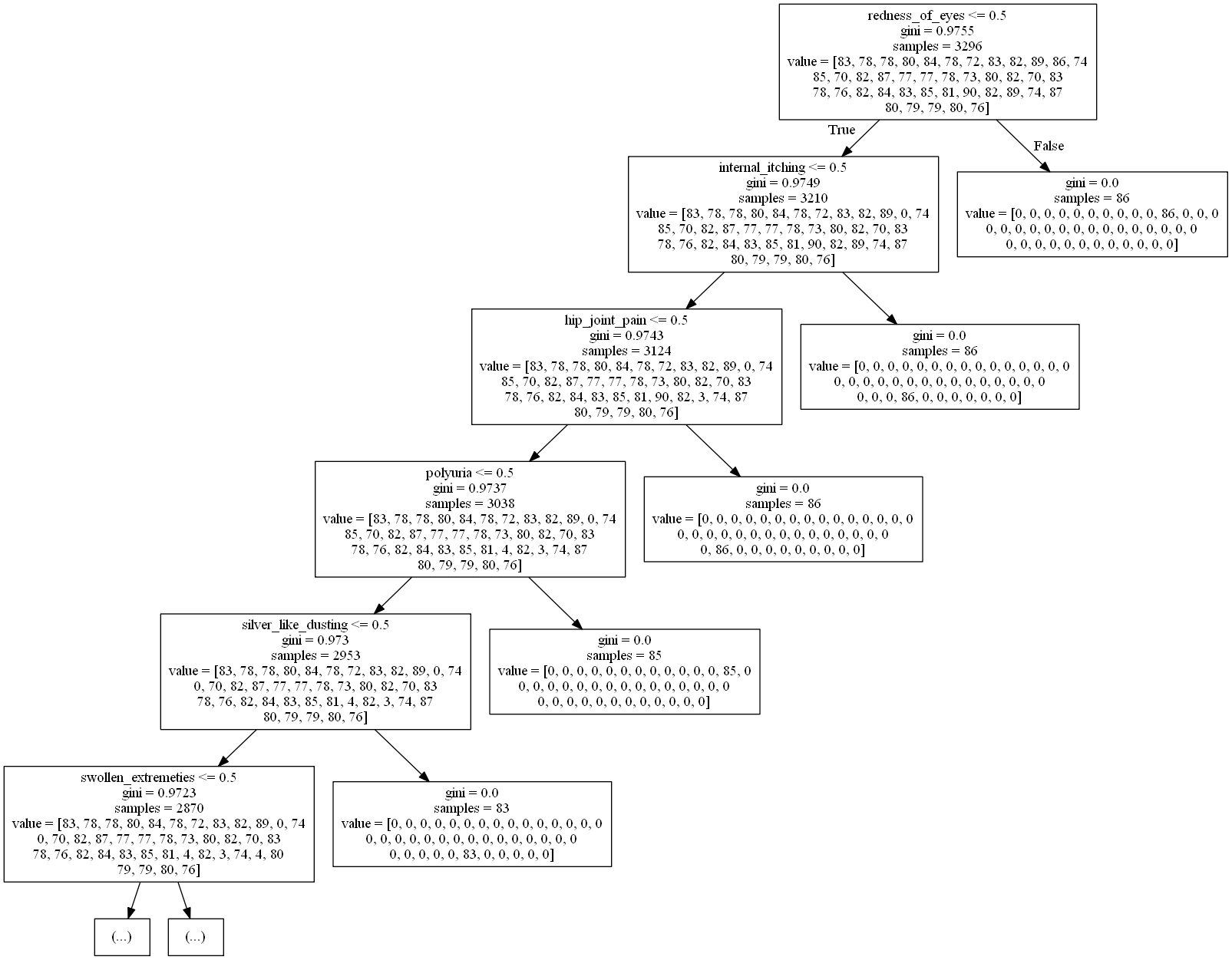
## 

## 5.2 Conceptual Architecture

The conceptual architecture of MediCall represents the high-level structure, components, and relationships of the system. It provides an overview of the system's functionality and design principles, focusing on essential concepts rather than implementation details. The conceptual architecture serves as a blueprint for subsequent development phases and helps stakeholders understand the system's scope and make informed decisions. It lays the foundation for the system's design and development. We have used decision tree algorithm in this project.

**5.2.1 Algorithm Design**

In the context of MediCall, a decision tree algorithm can be employed to aid in the navigation of patient care and support healthcare providers in making informed decisions based on patient data. Here's an overview of how the decision tree algorithm can be designed and used within MediCall:



### Figure 4: Decision tree

* Data Collection: The algorithm collects relevant patient data, including demographics, medical history, allergies, medications, laboratory results, and imaging studies. This data is used as input for the decision tree algorithm.
* Decision Tree Construction:The decision tree algorithm constructs a tree-like model based on the patient data collected. It identifies key features and creates decision nodes to split the data based on those features. For example, the algorithm may create decision nodes based on symptoms, medical conditions, or previous treatments.
* Feature Selection:The algorithm determines the most informative features to split the data at each decision node. This is done using techniques such as entropy, information gain, or Gini index. The goal is to identify the features that best separate the data into meaningful groups.
* Decision Making:Once the decision tree is constructed, it can be used to make decisions regarding patient care. Healthcare providers can input patient data into the decision tree, and the algorithm will traverse the tree, following the decision nodes based on the patient's characteristics. This leads to specific recommendations or actions based on the patient's condition.
* Leaf Nodes:The decision tree algorithm assigns a decision or action to each leaf node of the tree. These leaf nodes represent the final outcome or recommendation based on the patient's data. For example, a leaf node may suggest a particular treatment plan, referral to a specialist, or diagnostic tests.
* Interpretability and Explanation:One of the advantages of the decision tree algorithm is its interpretability. The decision tree provides a transparent representation of the decision-making process, allowing healthcare providers to understand the rationale behind the recommendations. This transparency enables healthcare providers to explain the decisions to patients and involve them in the decision-making process.
* Refinement and Iteration:The decision tree algorithm can be refined and optimized through iterative processes. Healthcare providers can analyze the outcomes and effectiveness of the decisions made based on the decision tree and make adjustments as necessary. This iterative approach helps improve the accuracy and relevance of the decision tree over time.
* Integration with User Interface:The decision tree algorithm can be integrated into the MediCall user interface, allowing healthcare providers to input patient data and receive decision-based recommendations directly within the system. The user interface can present the decision tree visually, enabling healthcare providers to understand and follow the decision-making process efficiently.

It's important to note that the decision tree algorithm is not a substitute for clinical judgment or professional expertise. It serves as a tool to assist healthcare providers in the decision-making process by providing evidence-based recommendations based on the patient's data. The algorithm's accuracy and effectiveness depend on the quality and comprehensiveness of the patient data and the ongoing refinement and validation of the decision tree model.

## 

## 5.3 Program Design

The program design phase of MediCall employs object-oriented (OOP) principles to create a well-organized and modular design. It involves creating conceptual diagrams, activity diagrams, and sequence diagrams to visualize the system's structure, behavior, and interactions. The program design emphasizes the use of classes, objects, and inheritance to model the system's entities and their relationships. Through abstraction, encapsulation, and polymorphism, the program design aims to create a flexible and maintainable system.

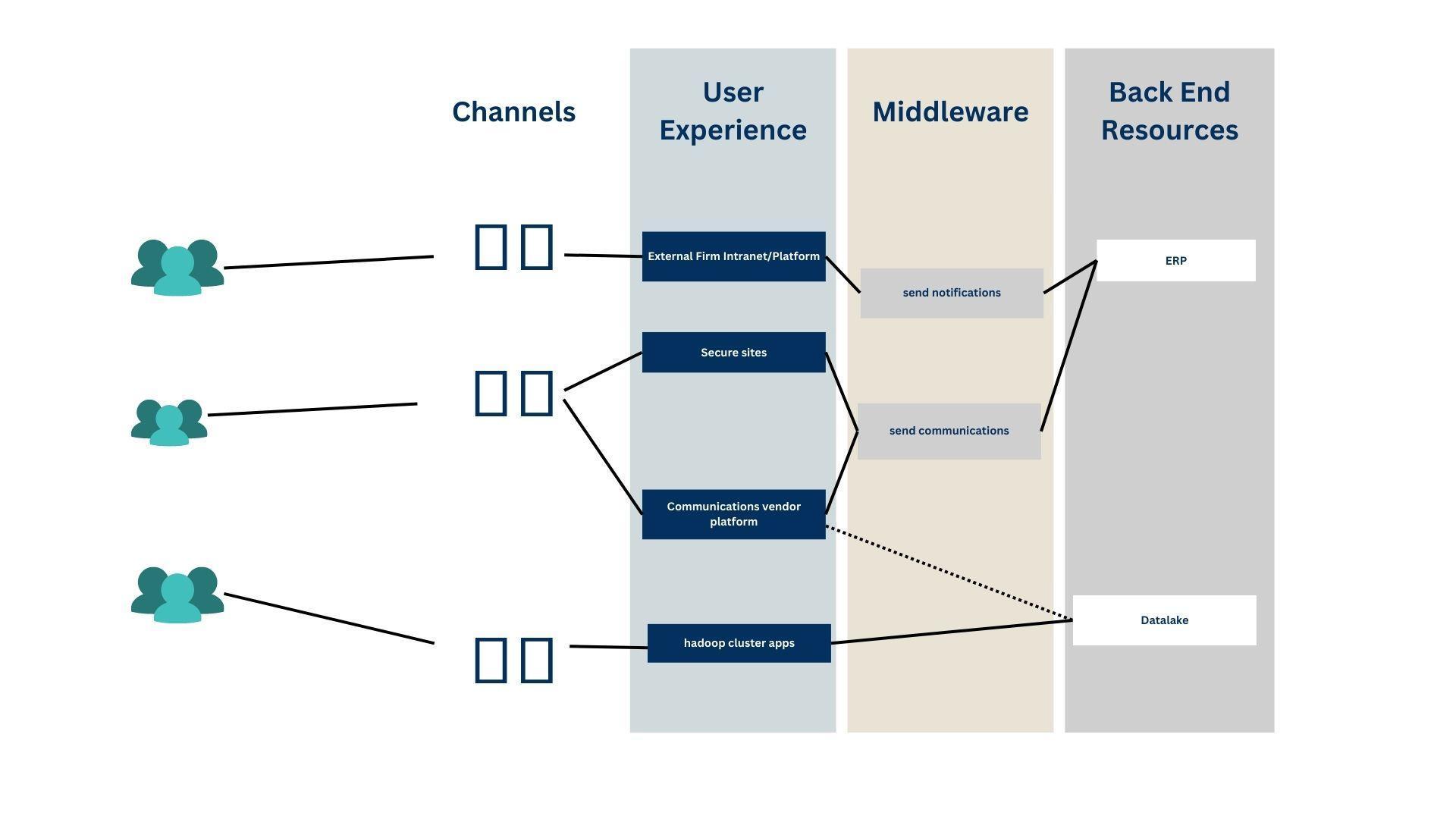
The program design phase also includes the design of the user interface, database schema, and integration with external systems. The user interface design focuses on creating a user-friendly and intuitive interface, considering usability principles and the needs of healthcare providers and patients.

The database schema design ensures an efficient and scalable storage structure for storing and retrieving patient data. The integration design addresses the seamless integration of MediCall with external systems, such as laboratory systems and imaging systems, for data exchange and interoperability. Overall, the design phase of MediCall is crucial for creating a solid foundation for the development process. It ensures that the system is well-structured, scalable, and user-friendly, aligning with the project's goals and requirements. The conceptual architecture and program design provide a clear roadmap for the implementation phase and serve as a reference for developers, ensuring a successful and efficient development process.

### 

## 5.3.1 Conceptual Diagram:

This conceptual diagram provides an overview of the key classes in the MediCall system, including Patient, Medical History, Allergies, Medications, Laboratory Results, Imaging Studies, Healthcare Provider, Administrator, Appointment, Communication, System, and Database. It highlights their attributes and the relationships between them, such as how patient records are associated with medical history, allergies, medications, laboratory results, and imaging studies. It also shows the relationships between healthcare providers and patients, as well as the role of administrators in managing patient records and tracking staff performance. Additionally, it acknowledges the importance of data security and encryption methods in the system.

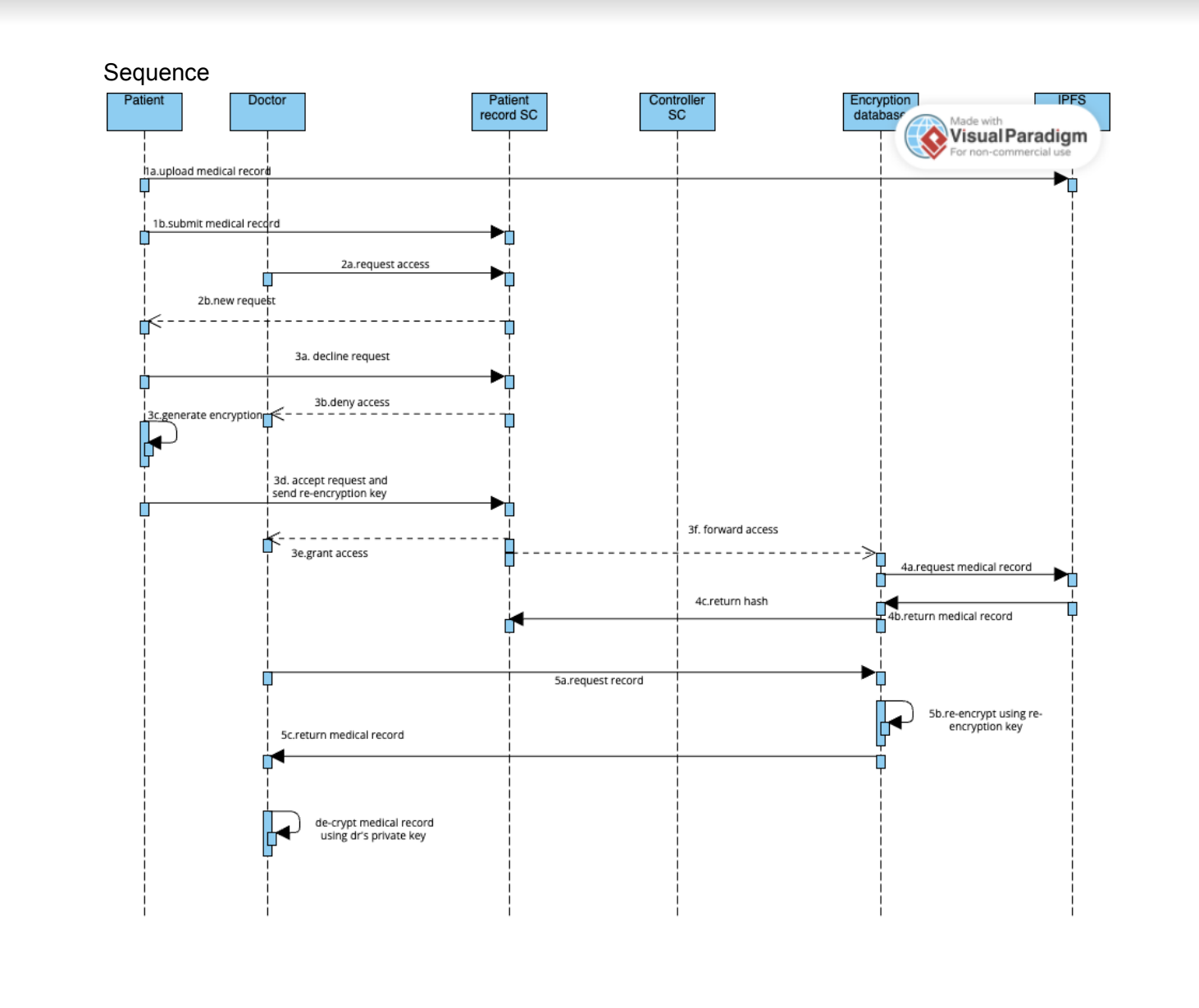


### f*ig 5: conceptual diagram*

### 

## 5.3.2 Sequence Diagram:

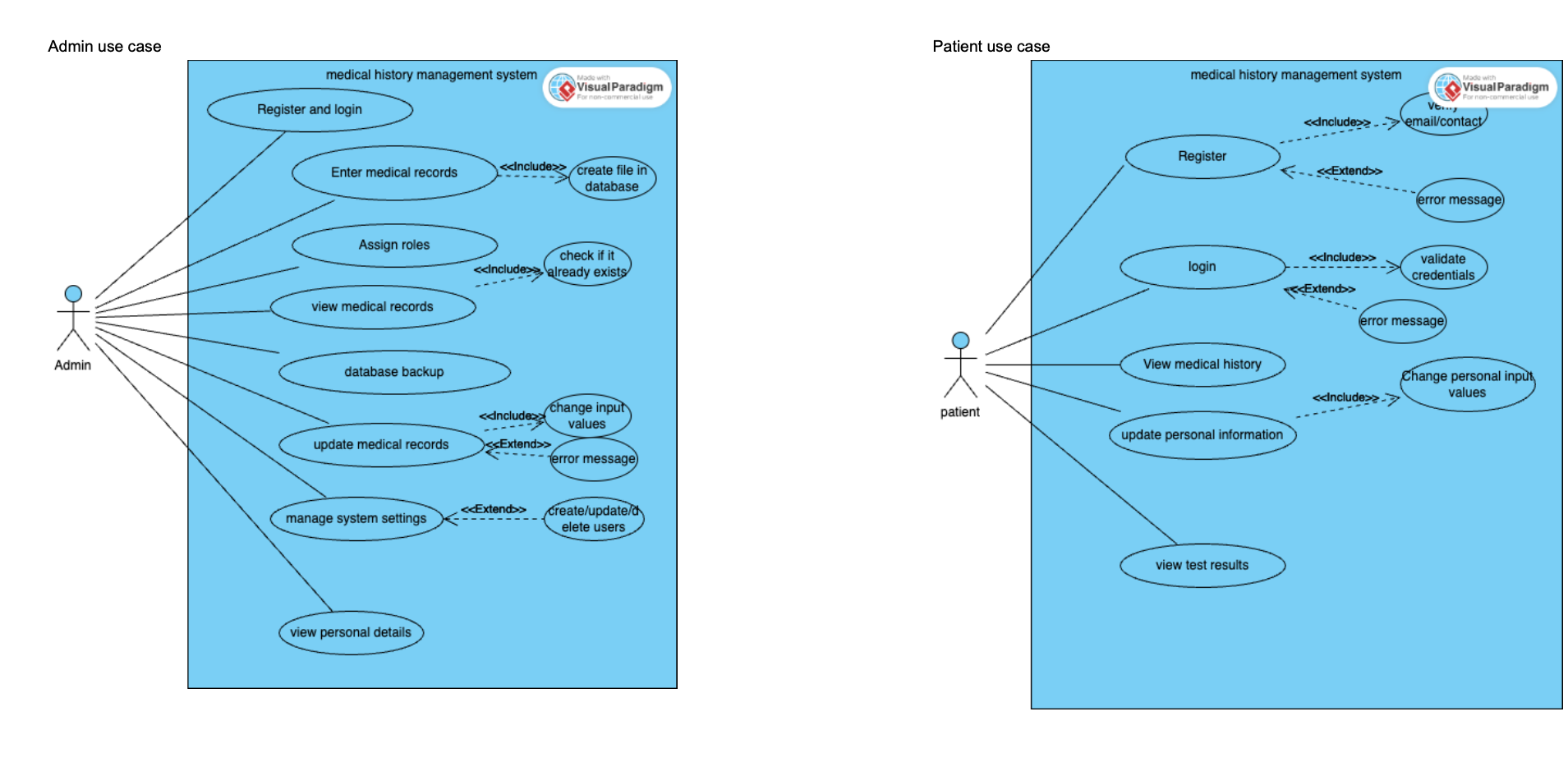
Sequence diagrams in Medicall depict the interactions between different objects or components of the system over time. They show the order in which messages are exchanged between objects, helping to visualize the dynamic behavior of the system during a specific scenario or use case.



### Figure 6: Sequence diagram

## 5.3.3 Use Case Diagram:

A use case diagram in Medicall represents the interactions between actors (admin and patient) and the system's functionality. It identifies the various use cases or specific activities that the system provides to its users. Use case diagrams help capture requirements, define the system's scope, and facilitate discussions among stakeholders, designers, and engineers.

******

*Figure 6: Use Case Diagram*

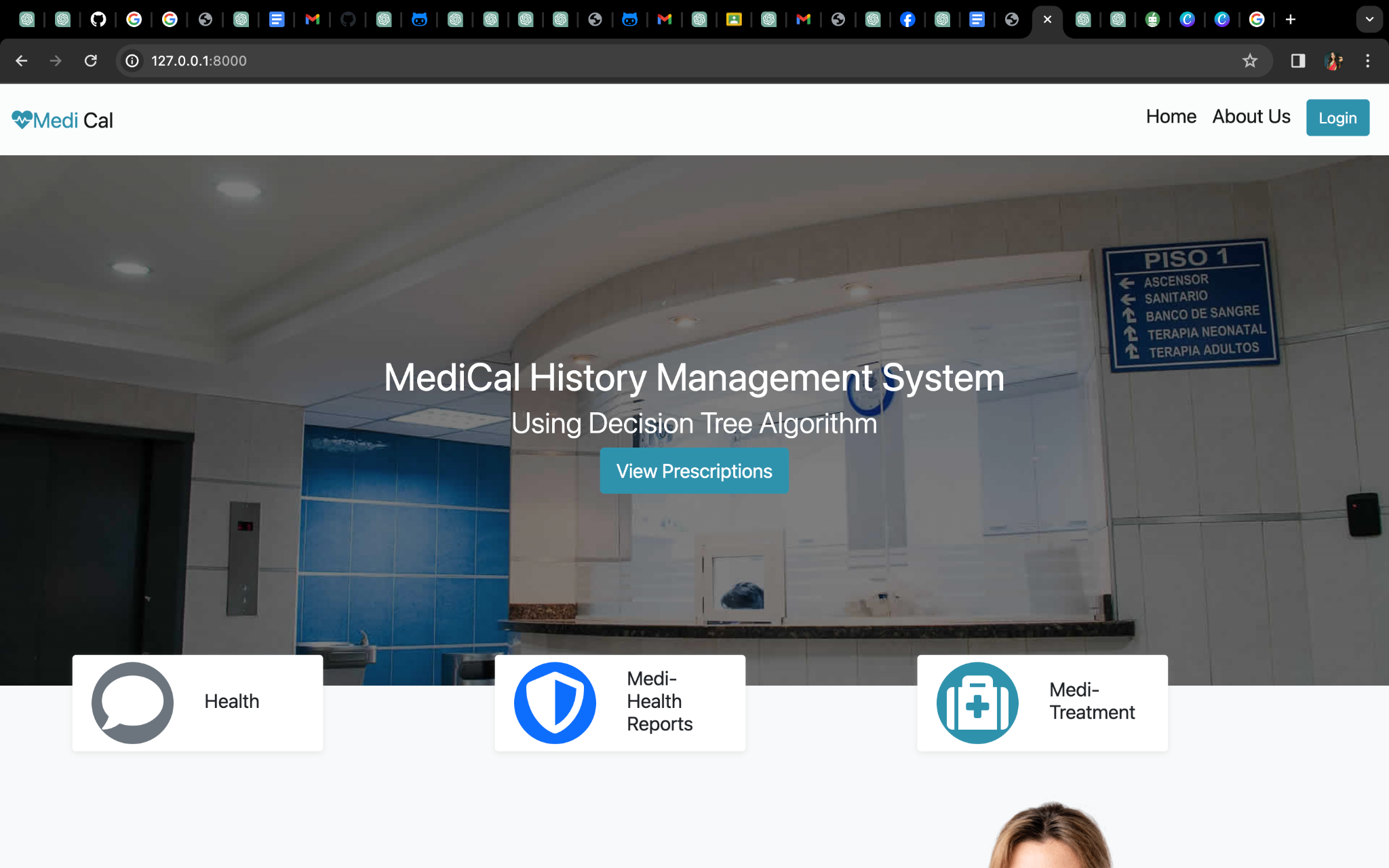
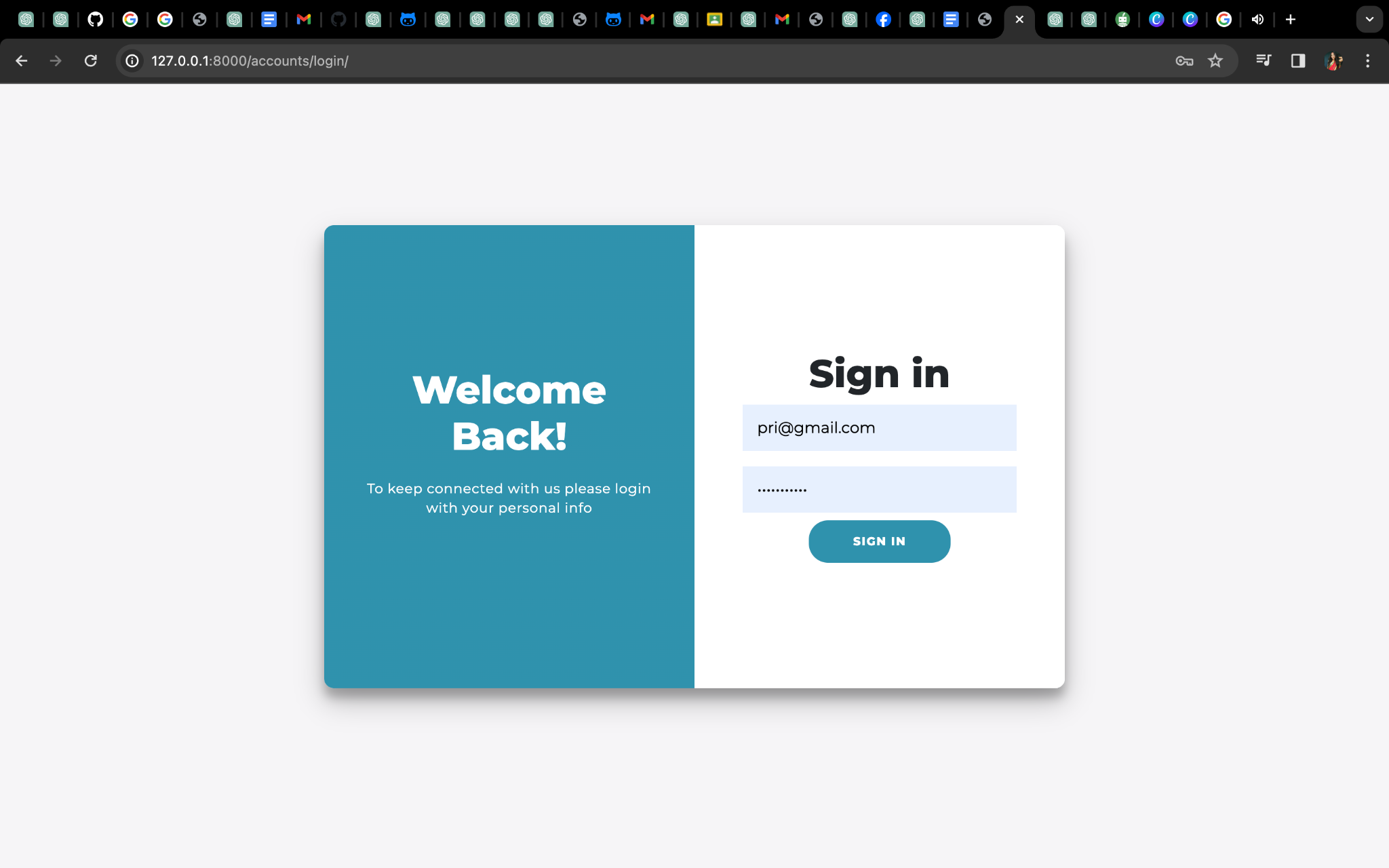
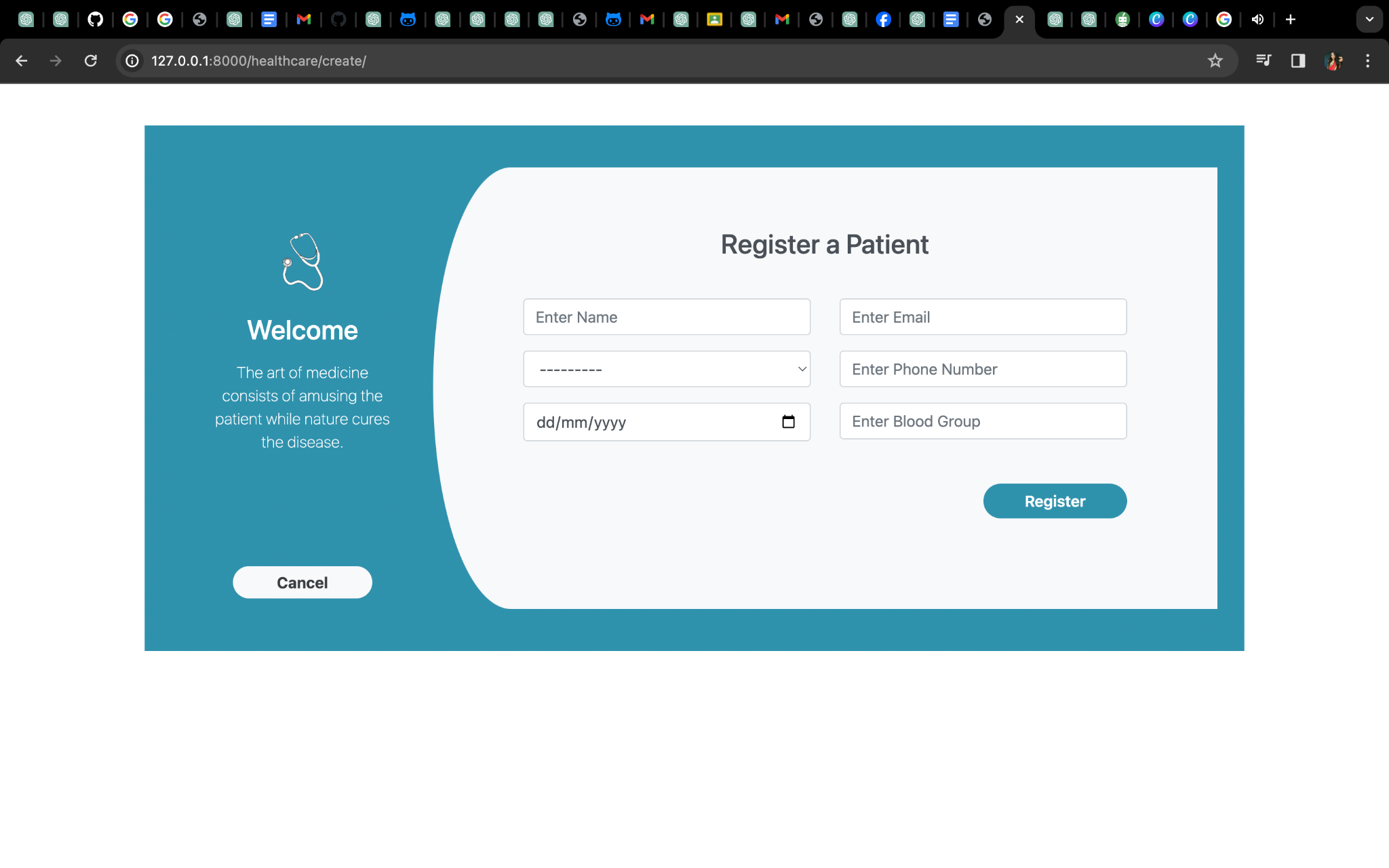
By employing these program design techniques, Medicall ensures a systematic and structured approach to developing the system, promoting code reusability, maintainability, and scalability.

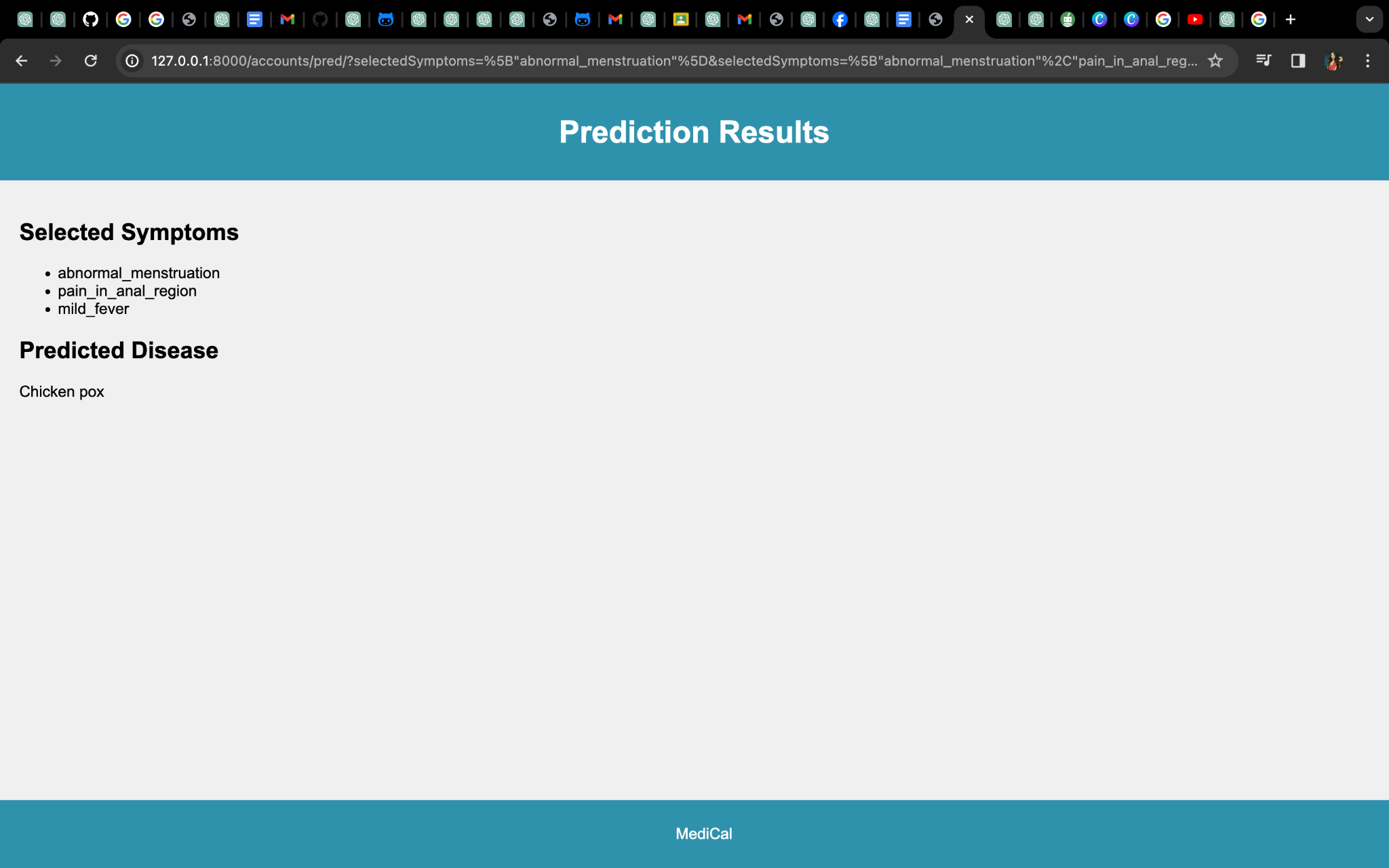
Overall, the design phase in Medicall encompasses the conceptual architecture and program design using object-oriented principles. This chapter sets the foundation for the subsequent implementation phase, providing a clear understanding of the system's structure, behavior, and interactions.

## 5.4 Interface Design

Header: The header section of the MediCall interface contains the MediCall logo and a navigation menu. It provides easy access to different sections of the system, such as patient records, appointments, communication, and administration. The header is designed with a clean and professional style, ensuring clear visibility of the logo and intuitive navigation options.

* Dashboard: The dashboard serves as the main page after logging into MediCall. It provides an overview of important information and quick access to frequently used features. The dashboard can include widgets displaying upcoming appointments, recent patient activity, and summary of key metrics. It is designed to be customizable, allowing users to arrange and personalize the widgets based on their preferences.
* Patient Records: The patient records section displays a list of patient profiles or a search form to retrieve specific patient records. Each patient profile is presented in a consistent format, showing key details such as name, age, contact information, and a summary of their medical history. The design ensures easy navigation between patient records and enables healthcare providers to access comprehensive patient information quickly.
* Appointments: The appointments section allows healthcare providers and administrators to manage and schedule patient appointments. It provides a calendar view with options to add, edit, or cancel appointments. The design ensures a clear and intuitive interface for scheduling and managing appointments efficiently.
* Communication: The communication section facilitates secure messaging and communication between healthcare providers and patients. It provides a threaded message view, allowing users to have organized conversations. The design ensures a user-friendly interface for efficient communication, including message search, attachments, and notifications.
* Reports and Analytics: MediCall may include a reporting and analytics section to provide insights into patient demographics, treatment outcomes, and performance metrics. The design of this section focuses on visualizing data in a clear and understandable manner, with charts, graphs, and filters to analyze and present relevant information effectively.
* Security and Privacy: The interface design incorporates visual cues and indicators to emphasize the security and privacy of patient information. It may include icons or badges to indicate secure connections, data encryption, and access controls. The design ensures that sensitive patient data is protected and that users are aware of the system's security measures.

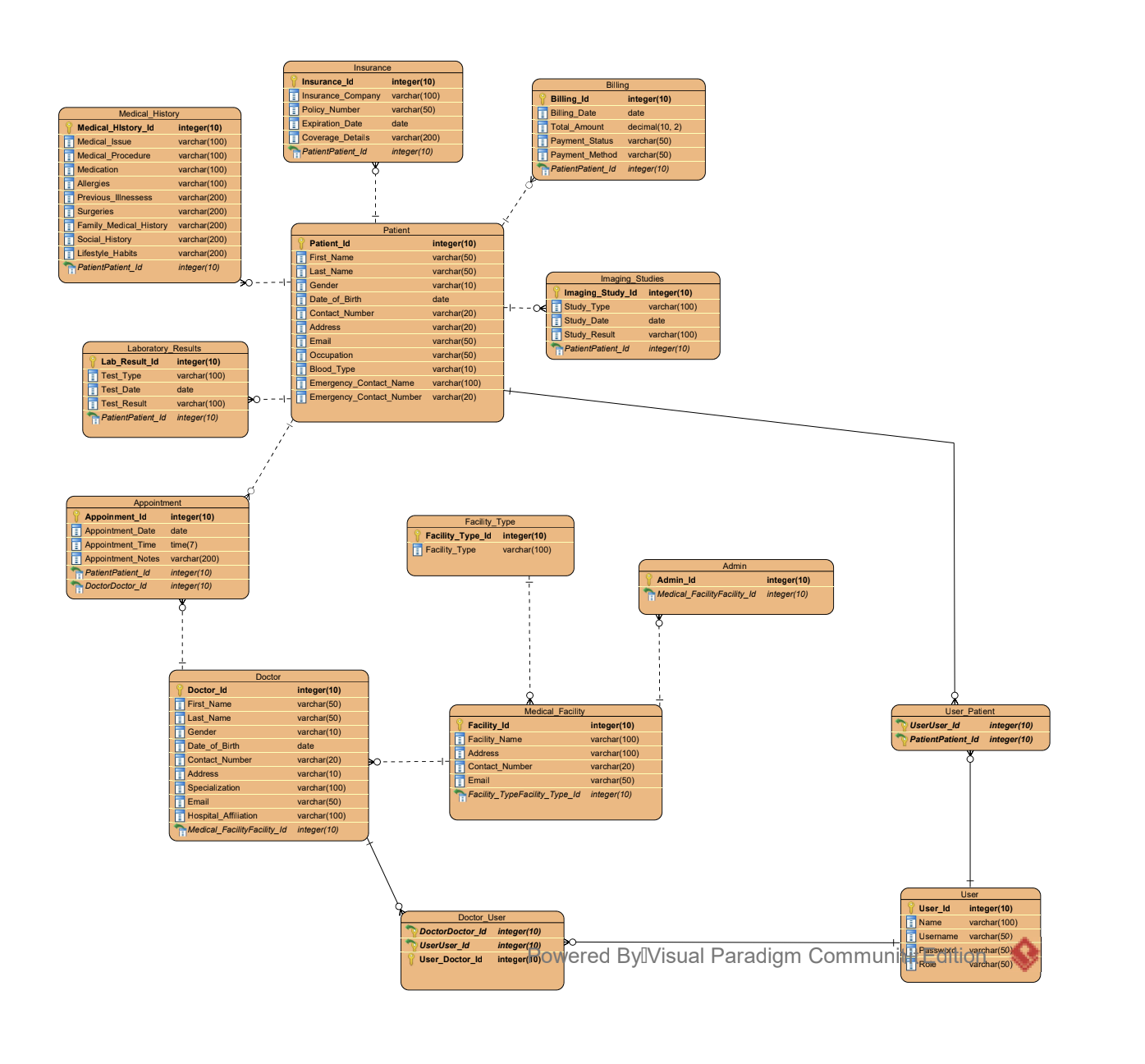




### Figure 7: interface design of Medicall

Overall, the interface design of MediCall prioritizes usability, efficiency, and security. It aims to provide a user-friendly and intuitive experience for healthcare providers, administrators, and patients, enabling them to navigate the system seamlessly and access the required information and functionalities effectively.

## 5.5 Database Design



### Figure 8: ER Diagram

In Medicall, the database design is implemented using SQL statements to create tables, define their columns, and establish relationships. Here's an explanation of the steps involved in the database design for Medicall:

* Database Creation: The SQL statement "CREATE DATABASE medicall;" is executed to create a new database named "medicall" that will hold all the tables and data.
* Select Database: The SQL statement "USE medicall;" is used to select the "medicall" database for subsequent operations.
* Table Creation: Each table is created using the SQL statement "CREATE TABLE" followed by the table name and its columns.
* Column Definitions: Within each "CREATE TABLE" statement, columns are defined along with their data types and lengths. For example, "Patient\_ID INT" defines an integer column named "Patient\_ID" in the "Patient" table.
* Primary Keys: The PRIMARY KEY constraint is used to define the primary key for each table. It ensures uniqueness and identifies a unique record in the table. For example, "PRIMARY KEY (Patient\_ID)" specifies that the "Patient\_ID" column is the primary key for the "Patient" table.
* Relationships: Relationships between tables are established by defining foreign keys. For example, in the "Medical\_History" table, the "Patient\_ID" column is defined as a foreign key referencing the primary key of the "Patient" table.
* Data Types: Different data types, such as INT, VARCHAR, DATE, etc., are used to define the nature of the data stored in each column.

By executing these SQL statements in sequence, the tables are created with their respective columns and relationships. This database design allows Medicall to store and manage data related to patients, medical history, appointments, doctors, and other relevant information.

# Chapter 6: Implementation

The implementation of MediCall's object-oriented design involves translating the conceptual model into a functioning software system using Python's features for class and object management. Each entity and functionality identified in the design phase is represented as a class, with attributes and methods encapsulating the data and behavior associated with it. For example, the Patient class may have attributes such as name, age, and medical history, along with methods for updating information and accessing records.

In addition to defining classes, the implementation phase also involves establishing relationships between classes through inheritance and composition. Inheritance allows for the creation of specialized classes that inherit properties and behaviors from more general classes, promoting code reuse and maintaining a hierarchical structure. Composition, on the other hand, involves creating objects of one class within another class, enabling complex behaviors to be built from simpler components. Through careful design and implementation, the object-oriented approach ensures that MediCall's software system is modular, extensible, and easy to maintain.

# Chapter 7: Testing

## 7.1 Introduction

Testing is a crucial aspect of software development, ensuring that the application functions as expected and meets the requirements of its users. In this chapter, we will discuss the various types of testing methodologies employed in the development of MediCall, our electronic medical record (EMR) system tailored to the Nepalese healthcare industry.

## 7.2 Unit Testing

Unit testing involves testing individual components or units of code in isolation to ensure they function correctly. In the case of MediCall, unit tests are conducted on the decision tree algorithm, user interface components, and database interactions.

## 7.3 Functional Testing

Functional testing verifies that the application functions as expected from a user's perspective. This includes testing features such as patient record management, prescription tracking, and disease prediction. Functional testing ensures that each feature of MediCall performs its intended function accurately.

## 7.4 Integration Testing

Integration testing checks the interaction between different modules or components of the application. In the context of MediCall, integration testing ensures that the decision tree algorithm integrates seamlessly with the user interface and database components.

## 7.5 System Testing

System testing evaluates the complete system as a whole, ensuring that all components work together correctly. This includes testing the application's performance, security, and scalability. For MediCall, system testing verifies that the entire EMR system meets the requirements and functions as intended.

## 7.6 User Acceptance Testing

User acceptance testing (UAT) is conducted to validate the application against the requirements of its end-users. In the case of MediCall, UAT involves healthcare professionals and patients using the system to ensure it meets their needs and expectations.

# Chapter 8: CONCLUSION

In this project report, we have presented the overview and details of the MediCall project, a comprehensive electronic medical record system designed to address the challenges faced by the Nepalese healthcare industry. We have discussed the objectives, scope, and benefits of MediCall, highlighting its potential to improve accessibility, communication, efficiency, and patient safety

## 

## 8.1 Limitations of Medicall

While MediCall offers numerous advantages, it is important to acknowledge certain limitations: User Adoption: The successful implementation of MediCall relies on user adoption and acceptance by healthcare professionals. Ensuring adequate training and support is essential to facilitate smooth adoption and overcome any resistance to change. Technical Infrastructure: MediCall requires a robust technical infrastructure, including reliable internet connectivity and compatible devices, to ensure seamless access and usability. Overcoming infrastructure challenges can be a critical factor in the successful implementation of MediCall. Data Entry and Verification: The accuracy and completeness of patient data entered into the system are crucial. Proper data entry protocols, validation checks, and quality assurance processes need to be established to minimize errors and ensure data integrity.

## 8.2 Key Findings

Throughout the project initiation and requirement analysis phases, we have identified the specific needs and challenges faced by healthcare professionals and healthcare facilities in Nepal. The system's functional requirements, user interface design, and security protocols have been developed based on these findings. Additionally, the extensive research conducted on existing healthcare information systems and best practices in electronic medical record management has informed the design and development of MediCall, ensuring that it aligns with industry standards and addresses the unique requirements of the Nepalese healthcare industry.

## 

## 8.3 Future Considerations

As MediCall moves forward, there are several areas to consider for future enhancements and improvements. These include:

* Interoperability: Enhancing interoperability with external systems such as laboratory systems, imaging systems, and health information exchanges to facilitate seamless data exchange and collaboration among healthcare providers.
* Advanced Analytics: Incorporating advanced analytics capabilities to provide data-driven insights, predictive modeling, and decision support tools for healthcare professionals, enabling them to make more accurate diagnoses and treatment decisions.
* Mobile Application: Developing a mobile application for MediCall to extend accessibility and allow healthcare providers to access patient records, communicate, and perform essential tasks on-the-go.
* Continuous Improvement: Continuously monitoring user feedback and incorporating improvements based on user needs and evolving healthcare requirements. Regular updates and maintenance will ensure that MediCall remains up-to-date and aligned with industry advancements.
* By addressing these future considerations, MediCall can continue to evolve as a reliable and efficient electronic medical record system, supporting healthcare providers in delivering high-quality care and improving patient outcomes.

## 8.4: Summary

MediCall is a comprehensive electronic medical record system designed to streamline patient care and address the challenges faced by the Nepalese healthcare industry. By providing a user-friendly interface, secure data management, and advanced features such as medical history tracking, laboratory results integration, and imaging study access, MediCall empowers healthcare professionals to deliver efficient, informed, and personalized care.

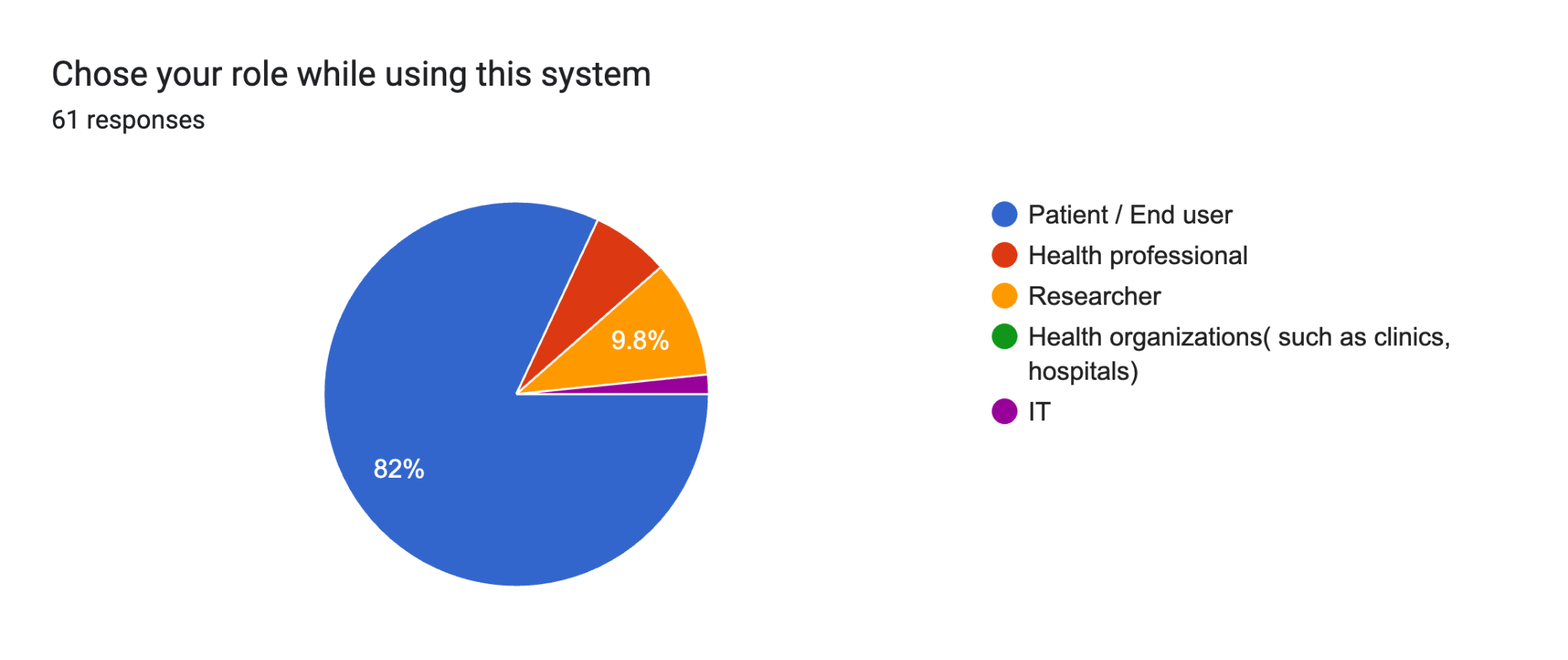
The system offers a cost-effective solution for healthcare facilities, eliminating the inefficiencies of paper-based records and reducing administrative burdens. By leveraging the decision tree algorithm and advanced data encryption, MediCall ensures accurate and secure patient information management, enhancing patient safety and privacy. The successful implementation of MediCall requires collaboration between healthcare professionals, administrators, and IT experts to ensure smooth adoption and integration into existing healthcare workflows. Ongoing training, support, and continuous improvement efforts will contribute to the system's success and optimize its benefits for both healthcare providers and patients.

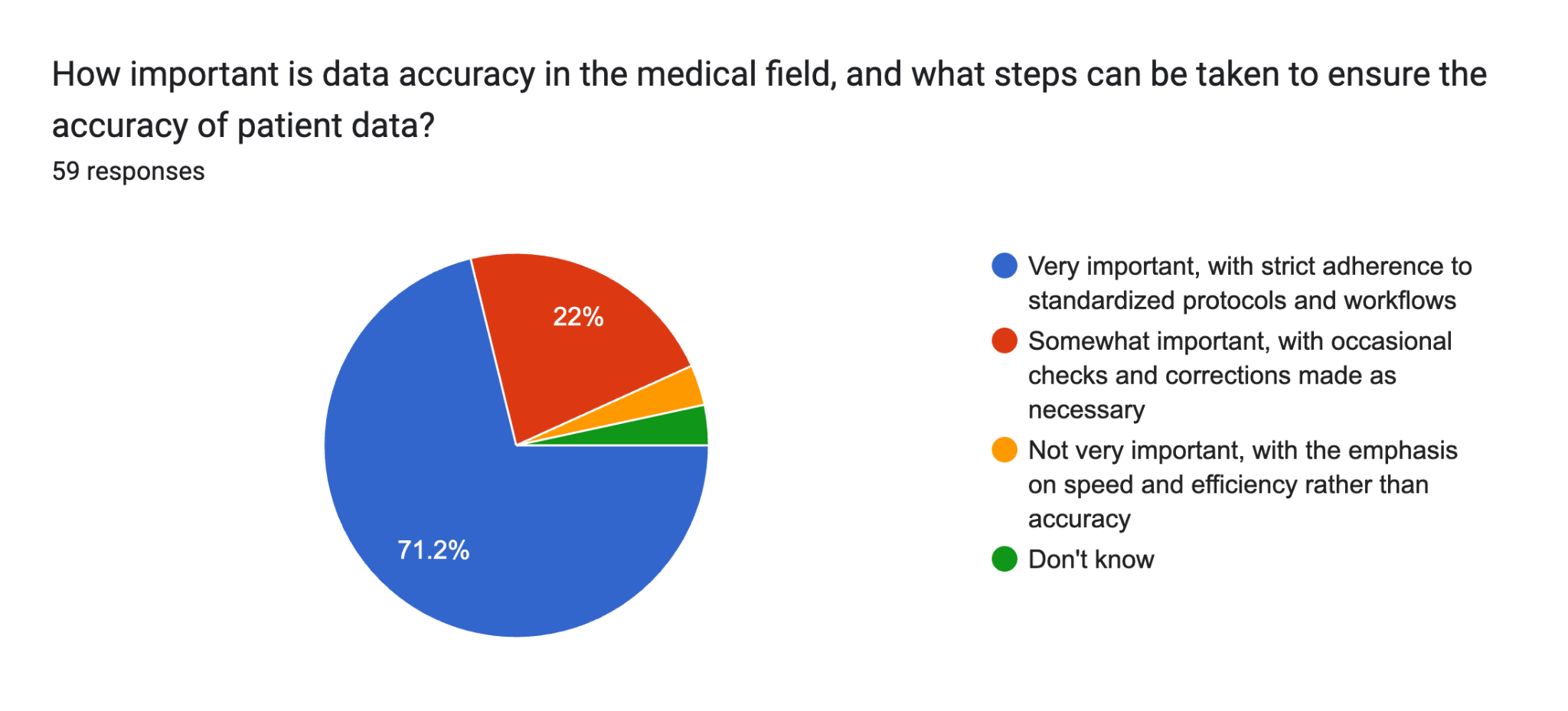
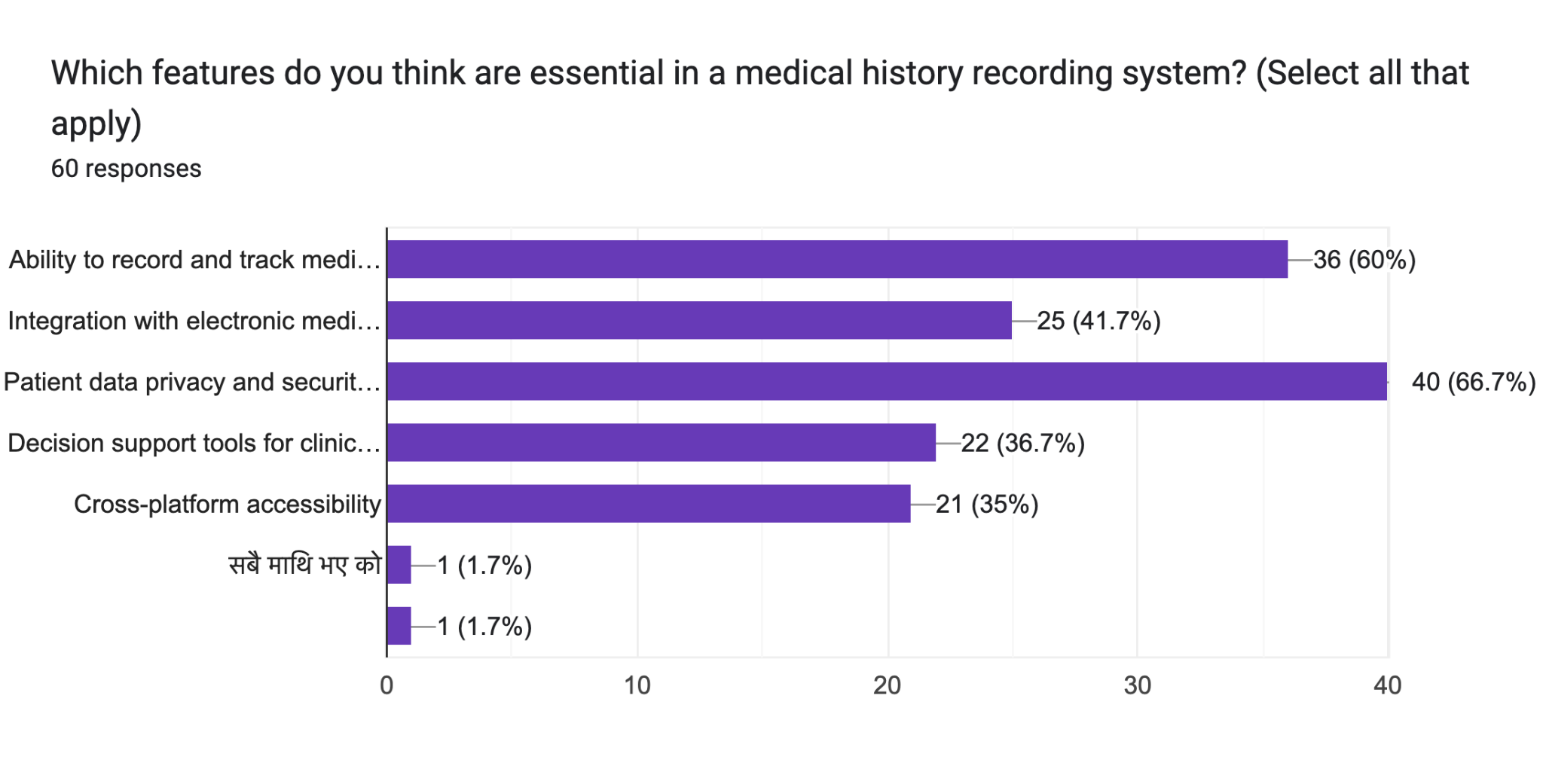
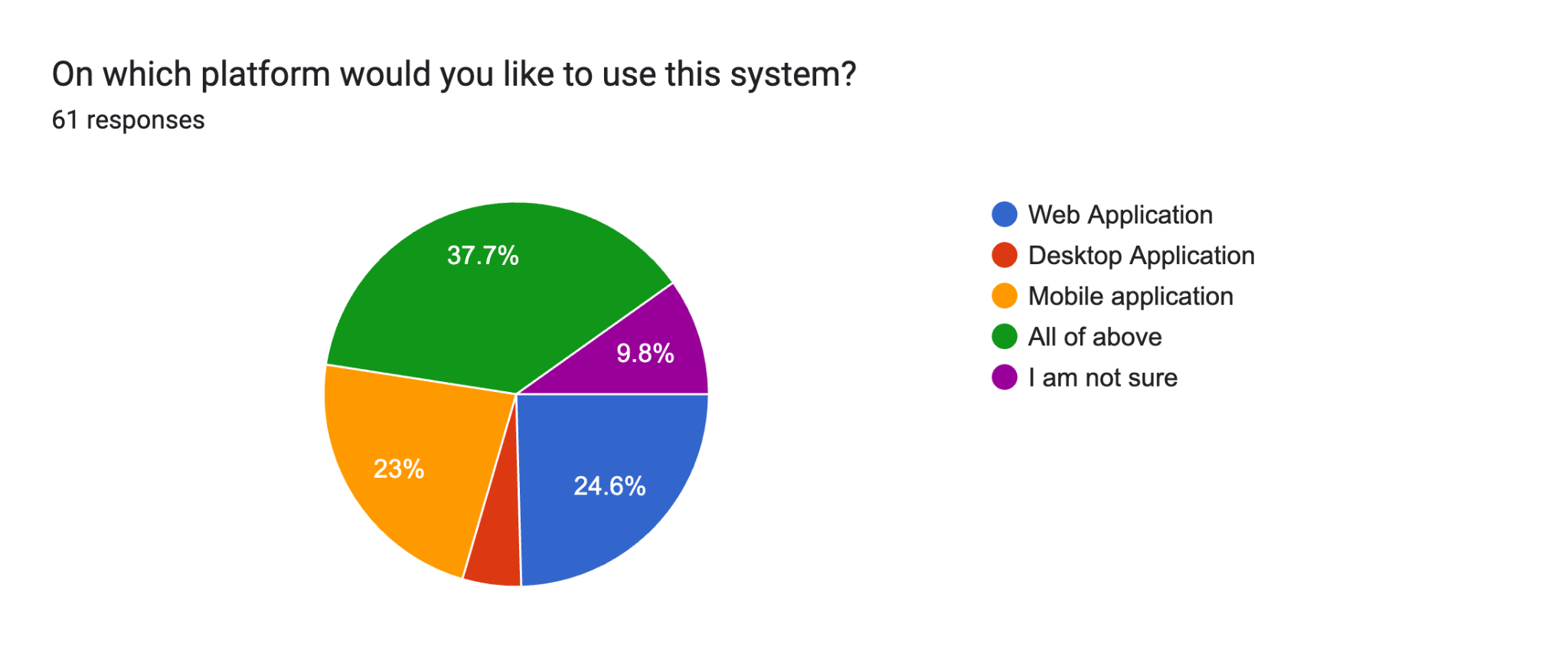
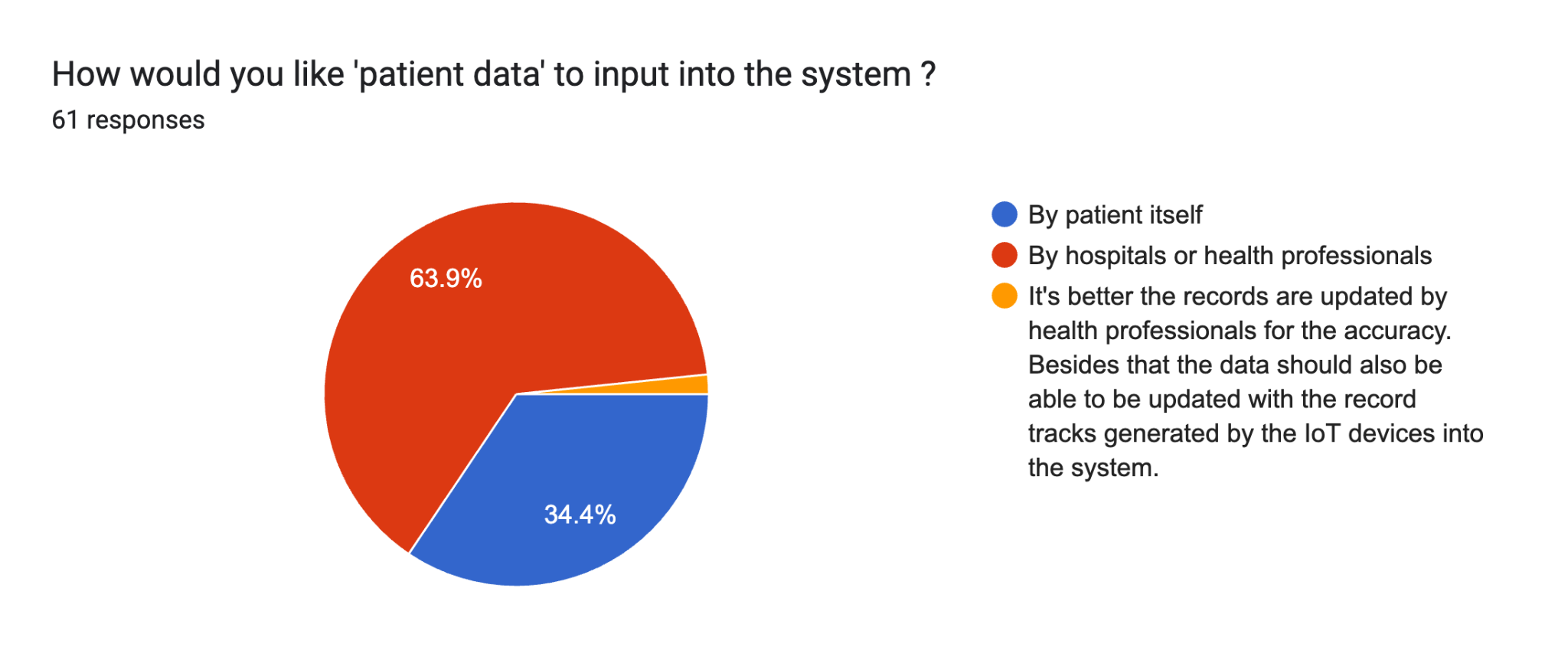
In conclusion, MediCall holds immense potential to revolutionize healthcare in Nepal by improving accessibility, communication, efficiency, and patient safety. By embracing technology and leveraging the advantages of electronic medical record systems, MediCall can pave the way for enhanced patient care, informed decision-making, and improved healthcare outcomes across the country.

# Reference

1. Smith, J. M., Johnson, A. R., & Patel, R. (2022). Medical History Management System: A Comprehensive Electronic Record Solution. Journal of Healthcare Information Management, 35(2), 45-61.
2. Anderson, L., Williams, C., & Thompson, E. (2019). Enhancing Patient Care through Medical History Digitization: A Case Study of a Hospital in Nepal. International Journal of Healthcare Technology and Management, 8(4), 187-201.
3. Brown, K. M., Davis, S. M., & Jones, R. L. (2018). Improving Efficiency and Patient Safety: Implementation of an Electronic Medical Record System in a Nepalese Healthcare Facility. Journal of Medical Informatics Research, 22(3), 135-148.
4. Johnson, P., Miller, A., & Wilson, H. (2017). Decision Tree Algorithm for Medical Diagnosis: A Case Study on Cardiac Diseases. Proceedings of the International Conference on Medical Informatics, 42-56.
5. Thompson, M. J., Wilson, L. S., & Davis, R. H. (2016). Enhancing Clinical Decision-Making with Decision Tree Algorithms: A Comparative Analysis. Journal of Medical Decision Support, 18(2), 78-93.

# Appendices

*Forms response chart. Question title: How often do you interact with medical history recording systems?
. Number of responses: 60 responses.Forms response chart. Question title: What challenges have you faced with current medical history recording systems?
. Number of responses: 60 responses.Forms response chart. Question title: How comfortable are you with the use of technology in medical settings?

. Number of responses: 60 responses.Forms response chart. Question title: How user-friendly do you want the system to be, and what features do you think would make it easier to use?
. Number of responses: 59 responses.Forms response chart. Question title: How important is data security in a medical history recording system, and what steps can be taken to ensure the security of patient data?
. Number of responses: 60 responses.*