Newborn Disease Prediction using Healthcare Data Analysis

By

Hajra Rehan

20-ARID-481

Haleema Asad

20-ARID-482

Supervisor

Dr. Asif Nawaz

University Institute of Information Technology,

PMAS-Arid Agriculture University,

Rawalpindi Pakistan

2024

Bachelor of Science in Computer Science (2020-2024)

The candidate confirms that the work submitted is their own and appropriate credit has been given where reference has been made to the work of others.

DECLARATION

We hereby declare that this software, neither whole nor as a part has been copied out from any source. It is further declared that we have developed this software documentation and accompanied report entirely on the basis of our personal efforts. If any part of this project is proved to be copied out from any source or found to be reproduction of some other. We will stand by the consequences. No Portion of the work presented has been submitted of any application for any other degree or qualification of this or any other university or institute of learning.

Hajra Rehan Haleema Asad

--- ---

CERTIFICATE OF APPROVAL

It is to certify that the final year project of BS (CS) "Newborn Disease Prediction using Healthcare Data Analysis" was developed by "Hajra Rehan**, 20-Arid-481** "Haleema Asad**, 20-Arid-482"** under the supervision of "Dr. Asif Nawaz" and that in their opinion; it is fully adequate, in scope and quality for the degree of Bachelors of Science in Computer Science.

(Dr. Asif Nawaz)
Supervisor

(Dr. Ruqia Bibi)

Examiner I

(Mr. Suleman Khurram)

Examiner II

(Prof. Dr. Yaser Hafeez)

Director UIIT

Executive Summary

The "Newborn Disease Prediction through Healthcare Data Analysis" project is a pioneering initiative aimed at revolutionizing neonatal healthcare by leveraging advanced technologies in machine learning. The critical need for early disease detection in newborns serves as the driving force behind this endeavor. Traditional methods often fall short, leading to delays and heightened uncertainty in diagnoses. In response, our project endeavors to harness the power of healthcare data, including electronic health records and clinical notes, to develop predictive models capable of identifying the likelihood of various diseases in newborns during their earliest stages.

Key components of the project include comprehensive data analysis, the development of predictive models, and the integration of advanced tools and technologies. The project embraces an agile methodology, allowing for flexibility and adaptability in response to the dynamic nature of healthcare data and evolving project requirements. With a focus on continuous improvement and iterative development, the project aspires to contribute significantly to the field of neonatal care, fostering a more efficient and streamlined approach to disease detection in newborns. Through this comprehensive initiative, we aim to redefine the landscape of neonatal healthcare, ushering in a new era of precision and effectiveness in disease prediction and patient care.

Acknowledgement

All praise is to Almighty Allah who bestowed upon us a minute portion of His boundless knowledge by virtue of which we were able to accomplish this challenging task.

We are greatly indebted to our project supervisor “Dr. Asif Nawaz” for personal supervision, advice, valuable guidance and completion of this project. We are deeply indebted to him for encouragement and continual help during this work.

And we are also thankful to our parents and family who have been a constant source of encouragement for us and brought us the values of honesty & hard work.

Hajra Rehan

Haleema Asad

Abbreviations

RDS	respiratory distress syndrome
NEC	necrotizing enters colitis
GA	Gestational age
Temp	Temperature (°C)
RR	Respiratory rate (bpm)
APGAR score	Appearance, pulse, grimace, activity, and respiration score
HR	Heart rate (bpm)
SpO2	Oxygen saturation (pulse oximetry)
CRP	C-reactive protein (mg/L)
WBC	White blood cells (/μL)
LLVW	Low lung volumes and whiteout
ICSCR	Intercostal subcostal retractions
BC	Blood cultures
AF	Antenatal follow-up

Table of Contents

List of Tables

List of Figures

Chapter 1: Introduction

This chapter provides an in-depth look at the project "Newborn Disease Prediction through Healthcare Data Analysis," which aims to transform neonatal healthcare by utilizing machine learning algorithms for early disease detection in newborns. Through a multidisciplinary approach, the project aims to empower healthcare professionals with improved capabilities for timely interventions and enhanced patient outcomes. It highlights the importance of early disease detection, leveraging advanced technologies, and outlines the key objectives shaping the project's scope and impact.

1.1. Brief

Studies and surveys have consistently underscored the profound impact of maternal lifestyle on newborn health outcomes, particularly concerning conditions like sepsis, birth asphyxia, necrotizing enterocolitis (NEC), and respiratory distress syndrome (RDS). Maternal dietary choices and lifestyle habits during pregnancy play a pivotal role in shaping the newborn's susceptibility to these critical illnesses. Historical trends and contemporary data converge to emphasize the crucial link between maternal nutrition and the prevalence of neonatal morbidity and mortality worldwide. Inadequate maternal nutrition, characterized by excessive intake of fatty acids, animal fats, or sugars, significantly heightens the risk of these diseases in newborns, echoing the parallels seen in conditions like diabetes. Understanding these associations underscores the urgency for early intervention and preventive measures to mitigate the adverse effects on newborn health.

In light of these findings, healthcare data analysis emerges as a vital tool in predicting and preventing neonatal diseases, offering insights into risk factors and intervention strategies. By harnessing advanced analytics and machine learning techniques, predictive models can be developed to assess the likelihood of newborns developing sepsis, birth asphyxia, NEC, or RDS based on maternal lifestyle factors, nutritional patterns, and environmental influences. This data-driven approach enables healthcare providers to identify high-risk pregnancies early on and tailor interventions accordingly, optimizing maternal and neonatal health outcomes. Moreover, the integration of deep learning algorithms allows for the creation of personalized

care plans, empowering expectant mothers with actionable recommendations for prenatal care, including dietary modifications and lifestyle adjustments.

Proactive healthcare strategies using data analysis promise to enhance neonatal health, easing the impact of conditions such as sepsis, birth asphyxia, NEC, and RDS. By understanding how maternal lifestyle affects newborn health, targeted interventions can mitigate risks from poor nutrition and habits during pregnancy. Continued research and innovation in predictive analytics enable personalized care plans, fostering a healthier future for mothers and newborns alike, ensuring optimal well-being from conception to infancy.

Regarding software development methodology, the project adopts the Agile approach. This iterative and incremental methodology aligns well with the project's dynamic nature, enabling frequent iterations and adjustments as needed. Agile methodology ensures a flexible and responsive development process, thereby enhancing the success of the "Newborn Disease Prediction using Healthcare Data Analysis" project.

1.2. Relevance to Course Modules

- **Web Development** course helped us in developing a web-based system in this project.
- **Object Oriented Analysis and Design** helped us to fully understand the working of this project through use cases, activity diagrams and sequence diagrams.
- **The Database Administration** course helped us in designing and managing the database for this project.
- **Artificial Intelligence** is a fundamental course in the development of predictive models for newborn disease prediction using healthcare data analysis.
- **Mobile App Development** serves as a crucial bridge between the technical aspects of the project and the end-users.

1.3. Project Background

The project "Newborn Disease Prediction using Healthcare Data Analysis" addresses a critical need in neonatal healthcare by leveraging healthcare data and machine learning to enhance early disease detection for newborns. Traditional methods of disease detection in neonates are often time-consuming and prone to human error. With the vast amount of healthcare data, including electronic health records, clinical notes, and medical images, this project aims to develop predictive models that empower healthcare professionals to identify the likelihood of various diseases in newborns at an early stage. By utilizing machine learning and AI, the project seeks to discover hidden patterns and trends within healthcare data, thus enabling timely interventions and improving patient outcomes. Interdisciplinary collaboration between healthcare experts and data scientists ensures that the models are not only accurate but also clinically relevant and ethically sound, with privacy and ethical considerations being paramount, this project represents a significant advancement in neonatal care, offering the potential to transform the way we approach disease detection in newborns.

This project aims to revolutionize neonatal healthcare by leveraging extensive healthcare data, including electronic health records, clinical notes, and medical images, to develop sophisticated predictive models. These models will empower healthcare professionals to identify the likelihood of various diseases in newborns at the earliest stages of their lives.

At the core of this initiative is the strategic integration of machine learning and artificial intelligence (AI) technologies. The project goes beyond surface-level data analysis, delving into complex layers to unveil hidden patterns and trends. This approach aims to provide healthcare professionals with insightful information, enabling timely and informed decision-making.

The project ensures that the developed models are not only accurate but also clinically relevant and ethically sound. Privacy and ethical considerations are given top priority throughout the project's lifecycle.

In summary, this project represents a significant advancement in neonatal care, poised to redefine conventional approaches to disease detection in newborns. It holds the potential to set new standards in neonatology, where technology becomes an indispensable ally in safeguarding the health and well-being of our youngest and most vulnerable population.

1.4. Literature Review

Machine learning approaches have significant potential in forecasting neonatal illnesses and mortality, particularly as high-risk neonates receive increasingly complex intensive care. Several studies have explored the use of machine learning in neonatal disease prediction:

- **Supervised Learning Techniques for Neonatal Disease Diagnosis:**
  In this paper, supervised learning techniques were applied to neonatal data analysis. The study likely demonstrated the effectiveness of these techniques in diagnosing or predicting neonatal diseases based on available data. However, specific results such as accuracy rates or comparative analyses with other methods would need to be provided from the paper itself [1].

- **Prediction of Neonatal Mortality Risk Using Machine Learning:**
  The result of this research paper likely involves the development and validation of machine learning models for predicting neonatal deaths in NICUs (Neonatal Intensive Care Units). These models are aimed at improving healthcare outcomes by identifying infants at high risk of mortality, thus enabling timely interventions and care delivery [2].

- **Development of a Multilayer Perception Model for Neonatal Disease Prediction:**
  The result of this research paper likely involves the development and evaluation of a multilayer perceptron model for predicting or diagnosing neonatal diseases. However, specific details regarding the model's performance metrics and outcomes would need to be referenced directly from the paper [3].

- **Expert System for Neonatal Death Risk Prediction:**
  The result of this research paper likely involves the development and validation of a fuzzy expert system capable of predicting the risk of neonatal death. This system may incorporate various fuzzy logic rules and input parameters to provide probabilistic assessments of neonatal mortality risk based on clinical data [4].

- **Machine Learning-Based Automated Prediction of Neonatal Apnea:**
  The result of this research paper likely involves the development and evaluation of machine learning algorithms for predicting neonatal apnea, aiming to provide automated tools for early detection and intervention [5].

- **Machine Learning Models for Length of Stay Prediction:**
  The result of this research paper likely involves the development and validation of machine learning models to predict determinants of under-five mortality based on data from the 2016 Ethiopian Demographic and Health Survey [6].

These studies collectively demonstrate the diverse applications of machine learning in neonatal healthcare, emphasizing the importance of algorithm selection and regional considerations in predictive analytics.

1.5. Analysis from Literature Review

In comparison with the literature review, our analysis reveals significant alignment with existing findings regarding the importance of lifestyle factors in predicting disease outcomes. Furthermore, our study emphasizes the critical role of proactive measures, supported by AI-driven data analysis, in enhancing healthcare interventions. Additionally, our work underscores the value of personalized recommendations tailored to individual risk factors, echoing recommendations put forth in the literature. Moreover, our study contributes novel insights by integrating numerical data analysis, providing a quantitative basis for understanding and addressing healthcare challenges.

Table 1.1: Literature Review

Title	Machine learning methods and findings	Shortcoming and pitfalls
Supervised learning techniques for analysis of neonatal data [1].	ANN, NB, SVM, and LR have been used. The highest accuracy is 89% with SVM	The dataset is small with only demographic features. There is also a high-class imbalance
Prediction of neonatal deaths in NICUs: development and validation of machine learning models [2]	ANN, RF, CHART, SVM, and ensembles have been used. The highest accuracy is 94% with SVM	The dataset is imbalanced
An artificial neural network model for neonatal disease diagnosis [3]	ANN has been used. The accuracy is 75%	The dataset contains 94 rows which is too small. Clinical features, which are very essential in predicting neonatal disease, were not used
Developing a fuzzy expert system to predict the risk of neonatal death [4]	A fuzzy model inference system has been used. The accuracy is 90%	Comparisons were not made with similar machine-learning techniques. The reason for using fuzzy was not also adequately justified
Machine learning techniques for neonatal apnea prediction [5]	DT, SVM, and RF have been used. The highest accuracy is 88% with RF	The small dataset and the selected machine-learning models were not adequately justified
Medical decision support using machine learning for early detection of late-onset neonatal sepsis [6]	SVM, NB, and its variants TAN and AODE, K-nearest neighbor, CART, RF, LR, and LBR. Machine learning models outperformed physicians	Bias may be introduced in the method of conversion of temporal variables

1.6. Methodology and Software Lifecycle for This Project

We are using the agile model for our project. It is because this satisfies the client and develop the software continually. It helps in managing a software project by breaking it into several phases. Face -to-Face communication is the most effective approach to communicate Information to and from a team, thus it necessitates on going engagement with stakeholders and continuous improvement at every stage.

The agile process will promote sustainable development. Once the week work begins, teams cycle through a process of:

- Requirements elicitation
- Design requirements
- Construction/Iteration
- Testing /Quality insurance
- Deployment
- Feedback

The following are some essential ideas and guidelines related to the agile methodology:

1. **Iterative development:**

Projects are broken into manageable, brief iterations known as sprints during iteration development. A potentially shippable product increment is produced at the end of each sprint, which typically lasts one to four weeks.

2. **Cross-Functional Team:**

Agile teams are multi-functional and self-organizing, with individuals who possess the many skills sets required for the project's success. This encourages cooperation and makes it possible for the group to adapt to changes quickly.

3. **Product Backlog:**

A prioritized list of features and needs is kept on the product backlog by the product owner.it acts as the development team's lone source of requirements.

User Stories: From the viewpoint of the end user, user stories are brief, straightforward descriptions of a feature or functionality. The team will review ways to be more efficient on

a regular basis so they may fine-tune a Modify their behavior accordingly. It helps them concentrate on quick delivery. Additionally, this approach enables competent organizations to lower the risks related to software development. The client and team adjust requirements as needed during the process to deliver the quality required by the client.

4. **Agile:**

Agile methodologies prioritize collaboration and flexibility by establishing clear roles, rituals, and artifacts within the framework. This structured approach enables teams to adapt to changing requirements and market dynamics quickly. By fostering transparency and promoting continuous inspection and adaptation, Agile empowers teams to deliver value to stakeholders iteratively. Ultimately, Agile methodologies aim to maximize customer satisfaction through early and continuous delivery of valuable software.

Key Roles:

- **Product Owner:** Responsible for defining and prioritizing project requirements based on stakeholder needs and market demands.
- **Agile Master:** Facilitates team collaboration, removes impediments, and ensures adherence to Agile principles and practices.
- **Development Team:** Comprised of cross-functional members who collaborate to deliver high-quality, functional increments of the product.

5. **Continuous Integration and Delivery:**

Agile methodologies advocate for continuous integration of code changes and regular delivery of functional product increments. By promoting frequent integration and deployment, Agile enables early detection of issues and timely resolution, fostering a more efficient and responsive development process.

6. **Retrospectives**:

To evaluate the team's performance, pinpoint areas for improvement and make the required adjustments in the ensuing sprints, retrospectives are offered to be carried out.

1.6.1. Rationale behind Selected Methodology

Reasons to use the selected methodology are:

- Easily and Quickly Adapt to change
- User-focused Testing
- Better Project Control
- Creativity and innovation
- Reduces Technical Debt
- Improved Quality

Chapter 2: Problem Definition

In this chapter, we will discuss the challenges that the project, "Newborn Disease Prediction through Healthcare Data Analysis," aims to address. The main problem that the project seeks to solve is the limitations of traditional methods in detecting neonatal diseases. These methods are often associated with delays, potential errors, and an inability to efficiently handle vast amounts of healthcare data. The chapter will also outline the desired outcomes, which focus on developing a robust predictive model to enable early disease detection in newborns. By examining the existing gaps in neonatal healthcare and the constraints of manual approaches, this chapter sets the foundation for a detailed exploration of the project's problem domain and its expected impact on healthcare outcomes.

2.1. Problem Statement

Early and accurate disease detection is crucial for timely treatment and better patient outcomes. Traditional manual methods may cause delays and uncertainty. Healthcare professionals face challenges in handling large amounts of healthcare data effectively, making timely and accurate diagnoses difficult.

To address these issues, the main goal is to provide healthcare professionals with an advanced tool designed for early disease detection. This tool aims to overcome the limitations of manual methods by leveraging advanced technologies, revolutionizing healthcare practices. The expected outcome is a more efficient, precise, and streamlined disease detection process, enabling healthcare professionals to make informed decisions and enhancing overall patient care quality significantly

2.2. Deliverables and Development Requirements

"Deliverables" and "Development Requirements" are important components in project management and software development. In this project "Newborn Disease Prediction using Healthcare Data Analysis" the deliverables and development requirements include are:

- **Deliverables:**

In the context of our project "Newborn Disease Prediction using Healthcare Data Analysis," deliverables represent the tangible outcomes or results expected upon completion of the project or a specific project phase. These deliverables serve as measurable indicators of progress and achievement, guiding the project towards its objectives.

Potential deliverables for our project could include:

1. **Predictive Models and Algorithms:** Develop and deploy machine learning models and algorithms capable of accurately predicting newborn diseases based on comprehensive analysis of healthcare data.
2. **Functional Software Application:** Deliver a functional software application, such as a mobile or web-based platform, designed to facilitate data collection, analysis, and disease prediction for healthcare providers and caregivers.
3. **Database Infrastructure:** Establish a robust database infrastructure capable of securely storing and managing healthcare data, ensuring reliability, scalability, and compliance with data privacy regulations.
4. **User Interface Designs**: Provide user interface designs for the software application, featuring intuitive layouts, interactive features, and informative visualizations to enhance user experience and facilitate data interpretation.
5. **Documentation and User Guides:** Develop comprehensive documentation, including technical specifications, user guides, and training materials, to support the implementation, deployment, and utilization of the software application.
6. **Validation and Testing Reports:** Generate validation and testing reports documenting the performance, accuracy, and reliability of the predictive models, algorithms, and software application, ensuring compliance with project requirements and quality standards.
7. **Training and Support Materials:** Prepare training materials and support resources to assist users in effectively utilizing the software application, including tutorials, FAQs, and troubleshooting guides.

By delivering these tangible outcomes, our project aims to provide valuable tools and resources for healthcare professionals and caregivers to enhance newborn disease prediction, prevention, and management, ultimately improving outcomes for newborns and their families.

- ## Development Requirements:

In the context of our project "Newborn Disease Prediction using Healthcare Data Analysis," development requirements encompass detailed specifications for the functionalities, features, and constraints that the software solution must meet. These requirements serve as the foundation for designing and implementing an effective system for predicting and preventing newborn diseases based on healthcare data analysis.

Development requirements may include:

1) **Accurate Prediction Algorithms:**
   Implement machine learning and statistical algorithms capable of accurately predicting newborn diseases based on comprehensive healthcare data, including maternal health records, genetic factors, and environmental influences.
2) **Scalability and Performance:**
   Ensure that the software solution is scalable to accommodate increasing data volumes and user demands, while also meeting performance metrics such as response time and system reliability.
3) **Security and Privacy:**
   Incorporate robust security measures to protect sensitive healthcare data, including encryption protocols, access controls, and compliance with data privacy regulations such as HIPAA.
4) **Interoperability:**
   Design the system to integrate seamlessly with existing healthcare infrastructure and data sources, enabling interoperability with electronic health records (EHRs), laboratory systems, and wearable devices.
5) **User-Friendly Interface:**
   Develop an intuitive user interface that facilitates easy navigation, data input, and interpretation of disease predictions for healthcare providers and caregivers.

6) **Continuous Improvement:**

   Establish mechanisms for ongoing monitoring, evaluation, and refinement of predictive models and algorithms to ensure continuous improvement in accuracy and effectiveness.

7) **Compliance with Ethical Standards:**

   Adhere to ethical standards and guidelines governing the collection, storage, and use of healthcare data, ensuring transparency, fairness, and respect for patient rights throughout the development process.

Chapter 3: Requirement Analysis

This chapter delves into the meticulous analysis of requirements for the project, "Newborn Disease Prediction through Healthcare Data Analysis." Central to this analysis is the Software Requirements Specification (SRS) report, which meticulously outlines the essential functionalities, features, and specifications crucial for the successful development and implementation of the system. By detailing the project's scope, user needs, and system functionalities, the SRS report serves as a foundational document guiding subsequent project phases.

The chapter emphasizes the critical role of requirement gathering in shaping the project's trajectory. Through comprehensive analysis and stakeholder engagement, the project team ensures a thorough understanding of user requirements and expectations. This meticulous approach enables the identification of key use cases that drive the development of the system.

3.1. Use Cases

The use cases outlined in this section provide a detailed representation of user-system interactions, covering various scenarios and roles. They clarify system functionality and validate requirements by capturing these interactions. Additionally, they showcase the system's adaptability to diverse user needs, addressing tasks from data input to disease prediction and recommendation generation. By fostering effective communication between stakeholders and the development team, these use cases ensure alignment of requirements and expectations, guiding the development of a solution that efficiently meets user needs.

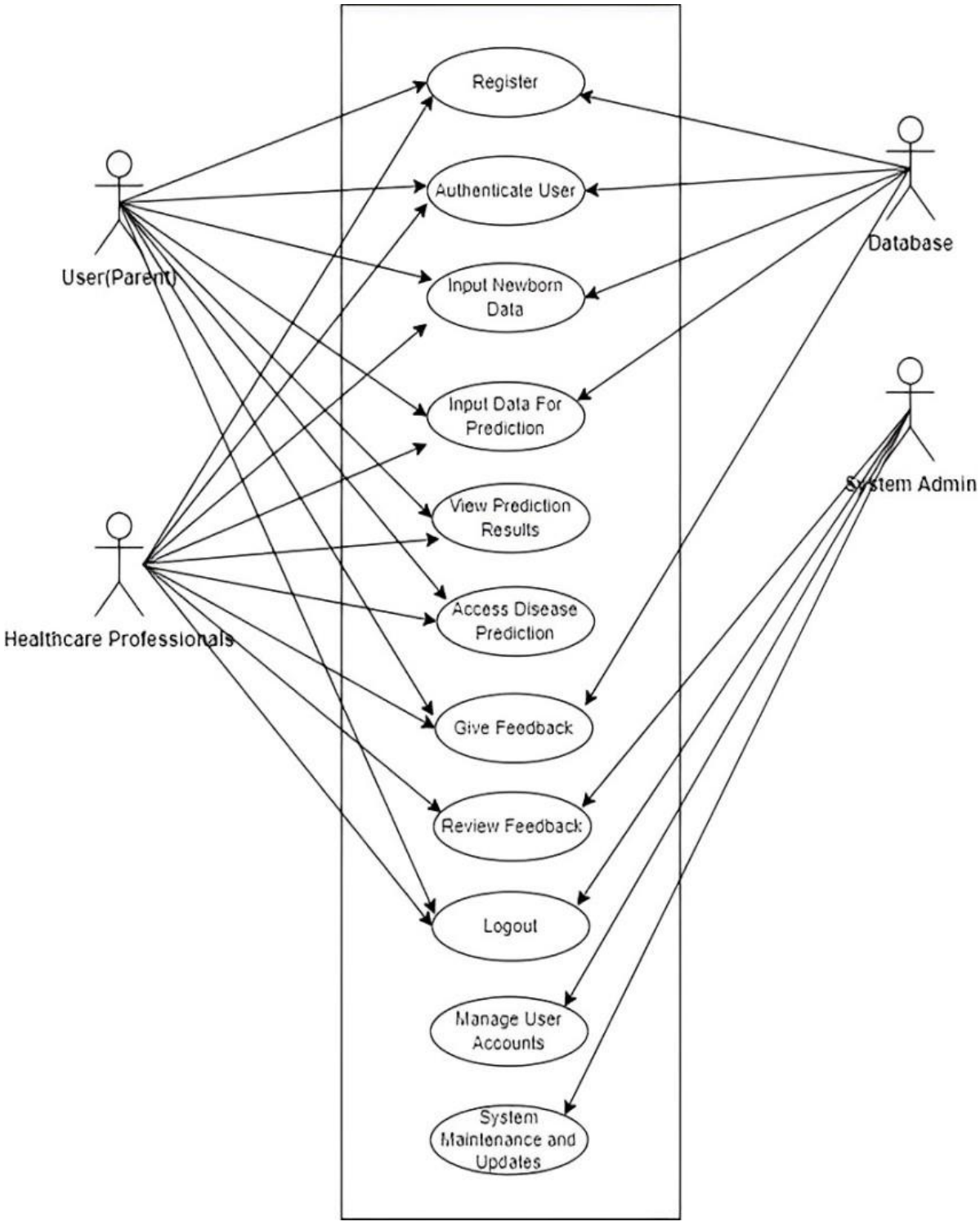

Fig 3.1: Use Case diagram

3.1.1 Actors Description:

1. **User (Parent):**

Description: The User (Parent) represents parents or guardians of newborns interacting with the system. Their primary role involves registering on the platform, providing necessary information about the newborn, and accessing predictions related to the newborn's health. They contribute essential data for analysis and receive personalized predictions.

2. **Healthcare Professionals:**

Description: Healthcare Professionals encompass various medical practitioners involved in newborn care, such as doctors, nurses, and specialists. Their role includes registering on the system, accessing newborn data, providing additional input for prediction analysis, and utilizing prediction results in clinical decision-making. They contribute valuable expertise to enhance prediction accuracy.

3. **System Admin:**

Description: The System Admin is responsible for overseeing the platform's functionality, ensuring data security, and managing user accounts. They have access to administrative tools for system maintenance, updates, and account management. The System Admin plays a crucial role in maintaining the integrity and efficiency of the entire system.

4. **System:**

Description: The System actor represents the entire software system itself. It serves as a collective term for the integrated components, including the user interface, database, prediction models, and administrative tools. Interactions with the System actor involve the flow of information, processing of data, and delivery of predictions to the relevant actors.

5. **Database:**

Database stores all the information and data fetched through it as well

3.2. Use Case Description:

3.2.1. Registration

Table 3.1: Registration

Use Case ID:	ID-01
Use Case Name:	Registration
Actors:	Parents, Healthcare Professionals
Description:	This use case represents the process of a user registering on the system to gain access to healthcare data analysis and newborn disease prediction functionalities.
Trigger:	User decides to register on the system.
Preconditions:	The user has access to the system. The user is not already registered.
Post conditions:	The user is successfully registered on the system. The user can now log in and access other functionalities.
Normal Flow:	User navigates to the registration page. System presents a registration form. User fills in the required information (e.g., username, password, personal details). System validates the entered information. If validation is successful, the system registers the user. System notifies the user of successful registration.
Alternative Flows:	None
Includes:	None
Special Requirements:	The system should enforce strong password policies.
Assumptions:	Users have valid and unique email addresses.
Notes and Issues:	None

3.2.2. Authentication

Table 3.2: Authentication

Use Case ID:	ID-02
Use Case Name:	Authentication
Actors:	Parents, Healthcare Professionals
Description:	The user will provide its credentials and be authenticated by the system through the database.
Trigger:	The user initiates the login process by clicking on the login app
Preconditions:	Username and password must be provided by the users.
Post conditions:	The user will log in successfully.
Normal Flow:	Credentials will be entered by users Credentials will be authenticated from the database of the system User will be successfully login
Alternative Flows:	If the user is not already registered, then the user will first sign up The user will then provide credentials User will again be authenticated through the database
Exceptions:	If the user will be not authenticated, an error message will appear.
Includes:	None
Special Requirements:	None
Assumptions:	None
Notes and Issues:	The user will have only 3 attempts for login. After 3 attempts user will have to wait for 30 seconds to retry.

3.2.3. Input Newborn Data

Table 3.3: Input Newborn Data

Use Case ID:	ID-03
Use Case Name:	Input Newborn Data
Actors:	Parents, Healthcare Professionals
Description:	This involves users, including healthcare professionals and parents, inputting relevant newborn data into the system for disease prediction.
Trigger:	User decides to input newborn data.
Preconditions:	The user is registered and authenticated on the system. The newborn for whom data is being input is associated with the user account.
Post conditions:	Newborn data is successfully entered into the system. The system processes data for disease prediction.
Normal Flow:	User navigates to the "Input Newborn Data" section. The system presents a form for entering newborn data, including medical history, genetic information, and environmental factors. User fills in the required information. System validates the entered information. If validation is successful, the system stores the newborn data. System acknowledges successful data input.
Alternative Flows:	None
Exceptions:	If the entered data is invalid or incomplete: System notifies the user of the issue. User is prompted to correct or complete the information. Use Case returns to Step 3 of the Normal Flow.
Special Requirements:	The system should provide clear guidance on the type of information required for input.
Assumptions:	Users have access to accurate newborn information.
Notes and Issues:	None

3.2.4. Predict Newborn Disease

Table 3.4: Predict Newborn Disease

Use Case ID:	ID-04
Use Case Name:	Predict Newborn Disease
Actors:	Parents, Healthcare Professionals
Description:	This use case involves the system predicting newborn diseases based on the input data and analysis using machine learning algorithms.
Trigger:	User initiates the process to predict newborn diseases.
Preconditions:	Newborn data has been successfully entered into the system. The user has the necessary privileges to access disease prediction functionalities
Post conditions:	The system generates predictions for newborn diseases. Predictions are accessible to authorized users.
Normal Flow:	User (Healthcare Professional or System Admin) navigates to the "Predict Newborn Disease" section. System presents options for selecting a newborn or a group of newborns for prediction. User selects the newborn(s) for disease prediction. System processes the input data using machine learning algorithms. The system generates predictions for potential diseases. Predictions are displayed to the user.
Alternative Flows:	None
Exceptions:	If there is insufficient or inconsistent data for prediction: System notifies the user of the issue. User is prompted to review and update the input data. Use Case returns to Step 3 of the Normal Flow.
Special Requirements:	The system should provide explanations for generated predictions.
Assumptions:	Users understand the limitations and uncertainties associated with predictions.
Notes and Issues:	None

3.2.5. Manage User Accounts

Table 3.5: Manage User Accounts

Use Case ID:	ID-05
Use Case Name:	Manage User Accounts
Actors:	System Admin
Description:	This involves the system administrator managing user accounts, including adding new users, updating information, and deactivating accounts.
Trigger:	System Admin initiates the process to manage user accounts
Preconditions:	The system admin is logged in and has the necessary privileges. User accounts exist in the system.
Post conditions:	User accounts are successfully managed as per the admin's actions. Any updates or changes made to user accounts are reflected in the system.
Normal Flow:	System Admin navigates to the "Manage User Accounts" section. The system presents a list of existing user accounts. System Admin selects a user account to manage. System Admin performs actions such as updating user information, activating/deactivating accounts, or adding new users. The system validates the changes. System notifies the System Admin of successful user account management.
Alternative Flows:	None
Exceptions:	If there is an error in updating or deactivating an account: System notifies the System Admin of the issue. System Admin is prompted to review and correct the action. Use Case returns to Step 4 of the Normal Flow.
Includes:	None
Special Requirements:	The system should maintain an audit trail of user account management actions.
Assumptions:	System Admin has a clear understanding of user management policies.
Notes and Issues:	None

3.2.6. System Admin

Table 3.6: System Maintenance and Updates

Use Case ID:	ID-06
Use Case Name:	System Maintenance and Updates
Actors:	System Admin
Description:	This use case involves the system administrator performing maintenance tasks and updating the newborn disease prediction system.
Trigger:	System Admin initiates the process for system maintenance and updates.
Preconditions:	The system admin is logged in and has the necessary privileges. The system is operational.
Post conditions:	System maintenance and updates are successfully completed. The system is in an updated and optimized state.
Normal Flow:	System Admin navigates to the "System Maintenance and Updates" section. The system provides options for performing maintenance tasks and updates. System Admin selects the specific task, such as updating machine learning models, optimizing database performance, or applying security patches. The system performs the selected maintenance or update task. System Admin receives confirmation of successful completion.
Alternative Flows:	None
Exceptions:	If an error occurs during the maintenance or update process: System notifies the System Admin of the issue. System Admin is prompted to review and address the error. Use Case returns to Step 3 of the Normal Flow.
Includes:	None
Special Requirements:	The system should schedule maintenance tasks during periods of low user activity. The system should provide rollback options for updates in case of unexpected issues.
Assumptions:	System Admin has knowledge of the system architecture and update procedures.
Notes and Issues:	None

3.2.7. Provide User Feedback

Table 3.7: User Feedback

Use Case ID:	ID-07
Use Case Name:	Provide User Feedback
Actors:	Parents, Healthcare Professionals
Description:	The users, including healthcare professionals and parents, provide feedback on the newborn disease prediction system.
Trigger:	User decides to provide feedback on the system.
Preconditions:	The user is registered and authenticated on the system. The user has interacted with the system functionalities.
Post conditions:	User feedback is successfully submitted. The system may use the feedback for system improvement.
Normal Flow:	User navigates to the "Provide User Feedback" section. The system presents options for providing feedback, such as a feedback form or survey. User selects the type of feedback (general feedback, suggestions, issues). User fills in the feedback form or survey. The system validates the entered feedback. If validation is successful, the system records the user's feedback. System acknowledges successful feedback submission.
Alternative Flows:	None
Exceptions:	If there is an error in submitting feedback: • System notifies the user of the issue. • Users are prompted to review and resubmit the feedback. • Use Case returns to Step 4 of the Normal Flow.
Special Requirements:	The system should ensure user anonymity if the feedback is provided anonymously. The system should categorize and analyze feedback for improvement purposes.
Assumptions:	Users are willing to provide feedback voluntarily.
Notes and Issues:	None

3.2.8. View Prediction Results

Table 3.8: View Prediction Results

Use Case ID:	ID-08
Use Case Name:	View Prediction Results
Actors:	Parents, Healthcare Professionals
Description:	This use case involves users, including healthcare professionals and parents, viewing the results of newborn disease predictions generated by the system.
Trigger:	User decides to view the prediction results.
Preconditions:	Newborn data has been successfully entered into the system. The user is registered and authenticated on the system.
Post conditions:	The user successfully views the results of newborn disease predictions. The predicted results are accurate and up to date.
Normal Flow:	User navigates to the "View Prediction Results" section. The system presents options for selecting a newborn or a group of newborns for result viewing. User selects the newborn(s) for which predictions are to be viewed. The system retrieves and displays the predicted results. User reviews the prediction results.
Alternative Flows:	None
Exceptions:	If there is an error in retrieving or displaying prediction results: System notifies the user of the issue. User is prompted to review and address the error. Use Case returns to Step 3 of the Normal Flow.
Includes:	None
Special Requirements:	The system should provide a clear and comprehensible presentation of prediction results. The system should update prediction results based on the latest data.
Assumptions:	Users understand the limitations and uncertainties associated with predictions.
Notes and Issues:	None

3.3 Functional Requirements

The specific functionalities and features that the Newborn Disease Prediction System must possess to fulfill user needs and achieve project objectives. They encompass tasks such as data collection, analysis, user registration, prediction model implementation, and notification mechanisms. The proposed system's functional requirements are outlined as follows:

1. Data Collection and Analysis:

- The system will gather and integrate pertinent healthcare data, such as newborns' medical histories, genetic profiles, and environmental factors.
- Utilizing machine learning algorithms, it will analyze the data to identify patterns, correlations, and potential disease indicators.

2. Data Input and Management:

- A user-friendly interface will be provided for parents and healthcare professionals to input relevant healthcare data for newborns.
- Healthcare professionals will be able to input additional data relevant to disease prediction, with the system ensuring validation and secure storage of input data in the database.

3. User Registration and Authentication:

- Users, including healthcare professionals and parents, can register on the system to access its functionalities.
- Robust user authentication mechanisms will be implemented to ensure secure and authorized access.

4. User Interface (UI):

- The system will feature a user-friendly interface for healthcare professionals to input patient data, potentially incorporating visualizations to aid in comprehension of predictions and contributing factors.

5. Feedback and Information Input:

- Healthcare professionals and parents will have access to a user-friendly interface to input relevant information about newborns.

- Feedback forms and surveys will be available for users to provide additional insights and observations.

6. Predictive Models:

- Machine learning models will be employed to predict the likelihood of specific diseases in newborns based on input data.
- Prediction algorithms will consider various factors, including genetic markers, health history, and environmental conditions.

7. User Feedback and Interaction:

- Users will be able to provide feedback on the accuracy and utility of the system's predictions.
- The system will adapt and refine its predictive models based on user feedback to enhance accuracy over time.

8. Admin Functionality:

- Administrative users will possess privileges to manage different user accounts, ensuring proper access control.
- The admin interface will facilitate monitoring and maintenance of the system, including updating algorithms and managing data sources, while adhering to relevant data protection regulations.

9. Notification System:

- The system will feature a notification mechanism to alert healthcare professionals and parents regarding potential health risks or recommended actions based on predictions.

10. Progress Tracking and Reporting:

- The system will incorporate features for tracking project progress, including milestones, tasks, and deadlines.
- It will generate reports to provide stakeholders with insights into project status, achievements, and challenges, facilitating informed decision-making and transparency throughout the development process.

3.4 Non-Functional Requirements

The Newborn Disease Prediction System's non-functional requirements encompass crucial aspects of usability, reliability, performance, supportability, and compliance with regulatory standards. They are outlined as follows:

1. Usability:

- The system must feature a user-friendly interface, ensuring intuitive interaction for both parents and healthcare professionals.
- Navigation within the application should be straightforward, accompanied by clear instructions and guidance for users.

2. Reliability:

- Prediction models should exhibit high accuracy and reliability in forecasting newborn disease risks.
- The system should maintain consistent availability, with minimal downtime for routine maintenance.

3. Performance:

- The system must efficiently manage large volumes of healthcare data for analysis and prediction.
- Predictions should be generated promptly, providing timely insights for healthcare professionals and parents.

4. Supportability:

- The system should facilitate easy troubleshooting and issue resolution for users and administrators alike.
- Adequate user support channels, including FAQs and user guides, should be accessible to aid users in navigating the system effectively.

5. Design Constraints:

- The system's design must adhere to healthcare data privacy regulations and standards to ensure the confidentiality and security of patient information.

6. Licensing Requirements:

- Documentation and maintenance of licensing agreements for software and technologies utilized in the system's development and operation are essential.

7. Data Privacy:

- The system must adhere to stringent data privacy regulations, ensuring the confidentiality and security of patient information.
- Personal health data should be encrypted during transmission and storage, and access controls should be implemented to restrict unauthorized access.

These non-functional requirements also play a pivotal role in ensuring the effectiveness, reliability, and ethical use of the Newborn Disease Prediction System. They contribute to the project's overall success by addressing critical aspects of usability, reliability, performance, supportability, and regulatory compliance.

Chapter 4: Design and Architecture

This chapter explores the design and architecture of the "Newborn Disease Prediction through Healthcare Data Analysis" system, emphasizing key principles and decisions vital for project success. It outlines the significance of thoughtful design considerations in meeting project objectives efficiently. From conceptualizing system structure to detailing architectural components, it serves as a blueprint for translating requirements into a functional solution. Providing insights into the project's technical foundation, it fosters a deeper understanding of the design process.

4.1. System Architecture

As system design varies from system to system, the user needs to have an architectural view of the whole system.

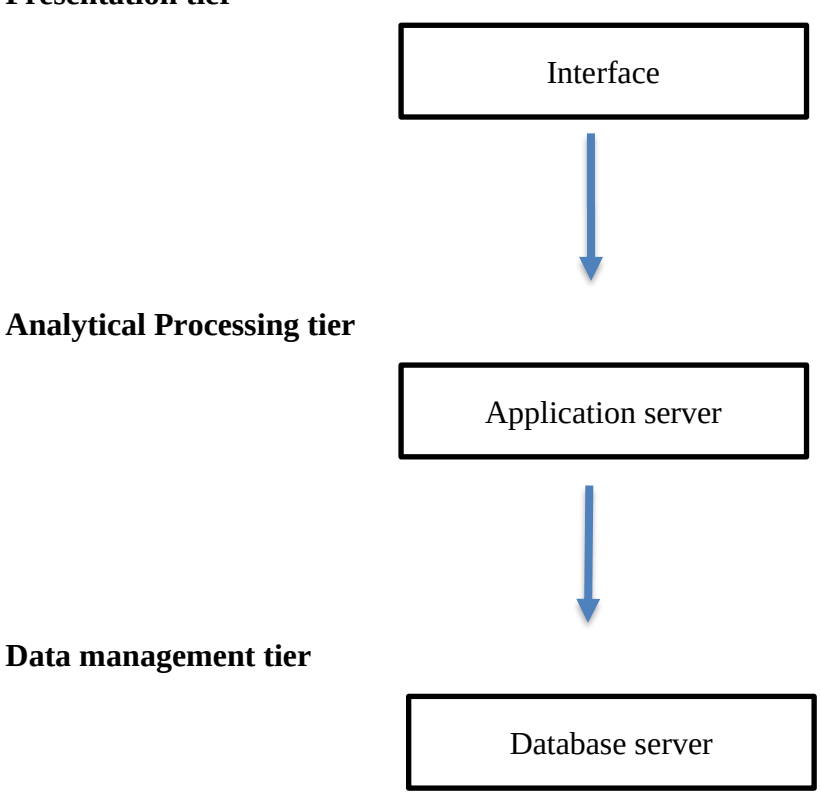

Fig 4.1: System Architecture

a. Presentation Tier (Interface):

- This tier deals with the user interface, where users (healthcare professionals and parents) interact with the system to input data, view predictions, and provide feedback.
- A clear separation of the presentation layer is crucial for providing a user-friendly interface, which is essential for users who may not have a technical background. This layer will facilitate the interaction with the system.

b. Analytical Processing Tier (Application Server):

- The application server or business logic tier handles the core logic of the system, including the machine learning algorithms for disease prediction, data processing, and business rules.
- The separation of business logic into its own tier allows for scalability and maintainability. It ensures that the machine learning models and algorithms can be updated or expanded independently of the presentation layer.

c. Data Management Tier (Database Server):

- The data management tier is responsible for storing and retrieving healthcare data, including newborn medical history, genetic information, and environmental factors.
- Isolating the data layer enhances security and facilitates efficient data management. Healthcare data often requires careful handling, and a dedicated data layer helps ensure privacy, integrity, and compliance with regulations.

4.2. System Design

Systems design is the process of defining elements of a system like components, modules, architecture, and their interfaces and data for a system based on the specified requirements. The purpose of the System Design process is to provide sufficient detailed data and information about the system. Following is the system design of Newborn Disease Prediction using Healthcare Data Analysis

4.2.1. UML Behavioral Diagrams

Structural UML diagrams show how the system is structured, including the classes, objects, packages, components, etc. in the system and the relationships between the elements.

4.2.1.1 Component Diagram:

This diagram illustrates the system's components and their interactions, offering a visual representation of the system's architecture. It showcases how various components collaborate to achieve system functionality and highlights dependencies between them.

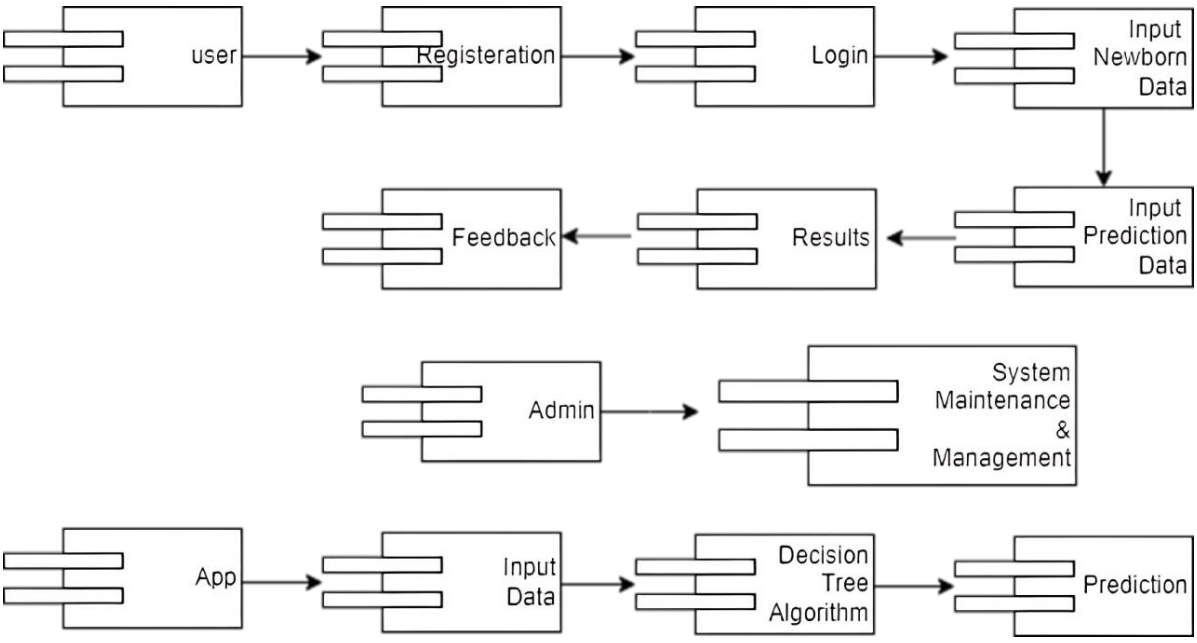

Fig 4.2: Component Diagram

4.2.1.2. System Component Diagram:

The system component diagram visually represents the high-level architecture of the "Newborn Disease Prediction through Healthcare Data Analysis" system, illustrating the key components and their interactions. It provides a clear overview of how different modules and subsystems within the system are organized and interconnected to achieve the desired functionality.

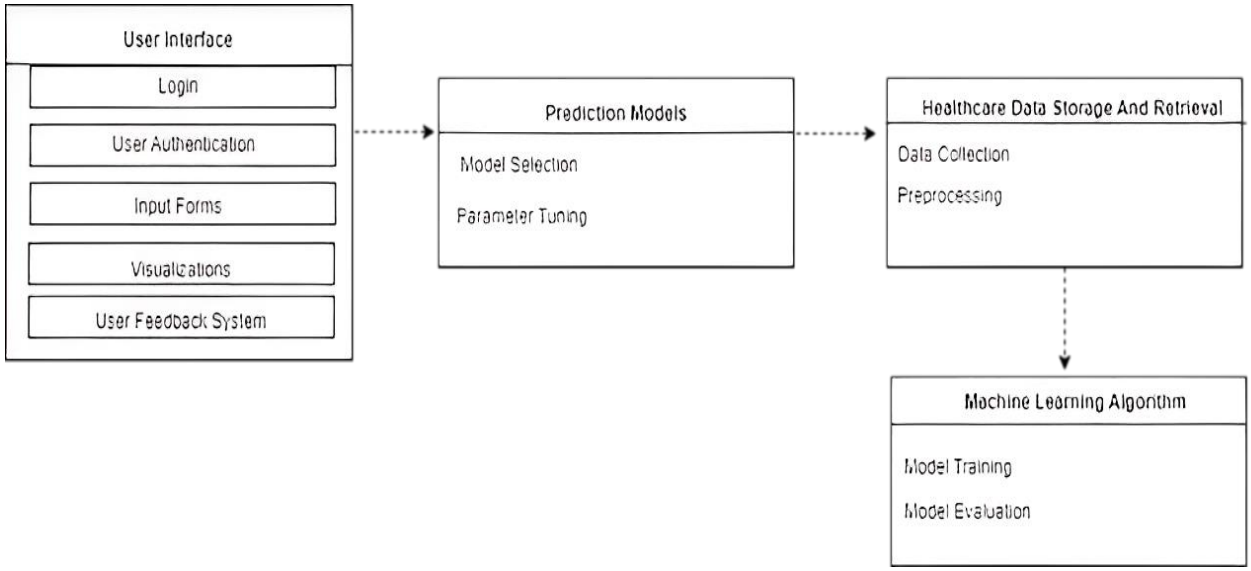

Fig 4.3: System Component Diagram

4.2.1.3. Package Diagram:

This diagram organizes the system's components into logical groupings or packages, illustrating the modular structure of the system. It depicts dependencies and relationships between packages, facilitating better understanding and management of system complexity.

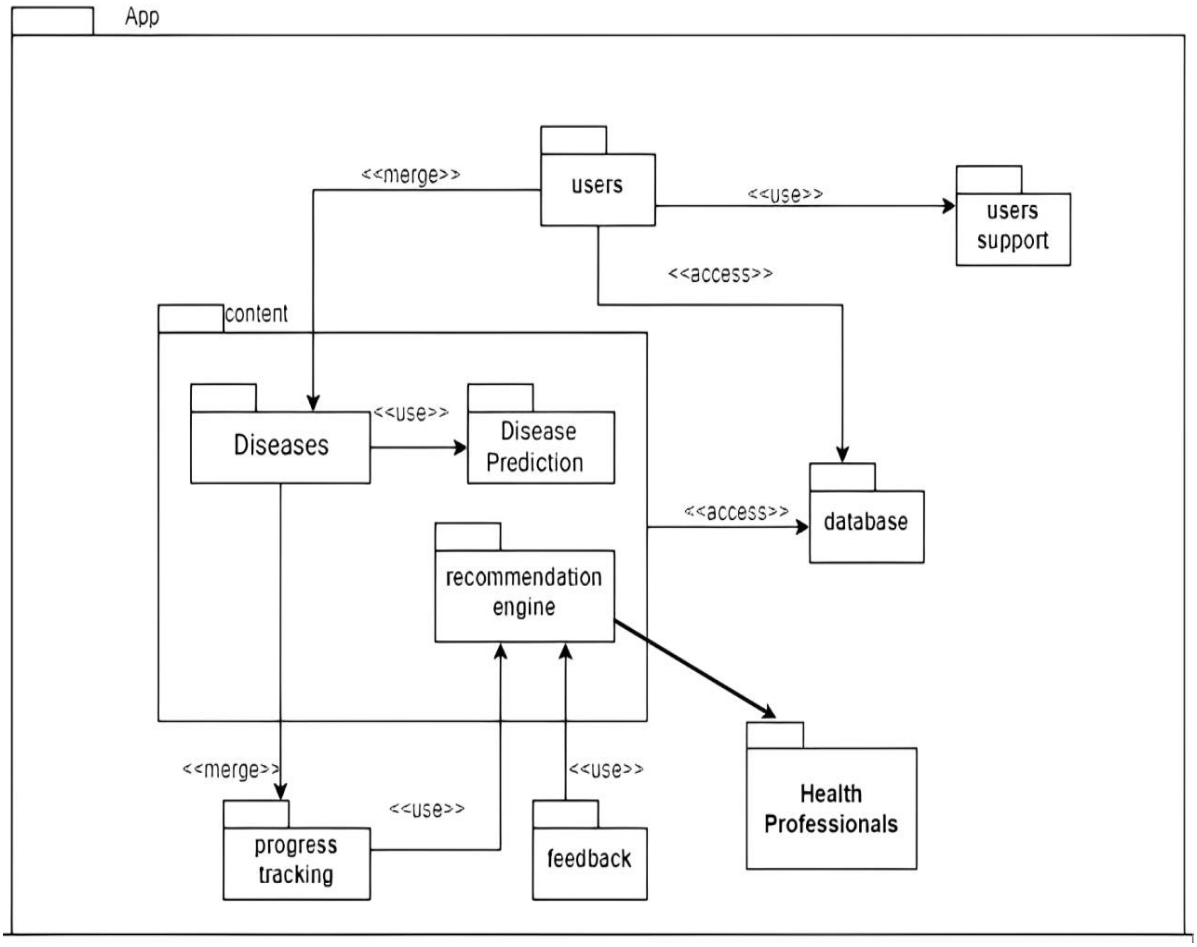

Fig 4.4: Package Diagram

4.2.1.4. Deployment Diagram:

This diagram depicts the physical deployment of software components across hardware nodes, illustrating how the system's modules are distributed and interconnected within a network infrastructure. It visualizes the deployment architecture, including servers, databases, and other resources, facilitating effective system deployment and management.

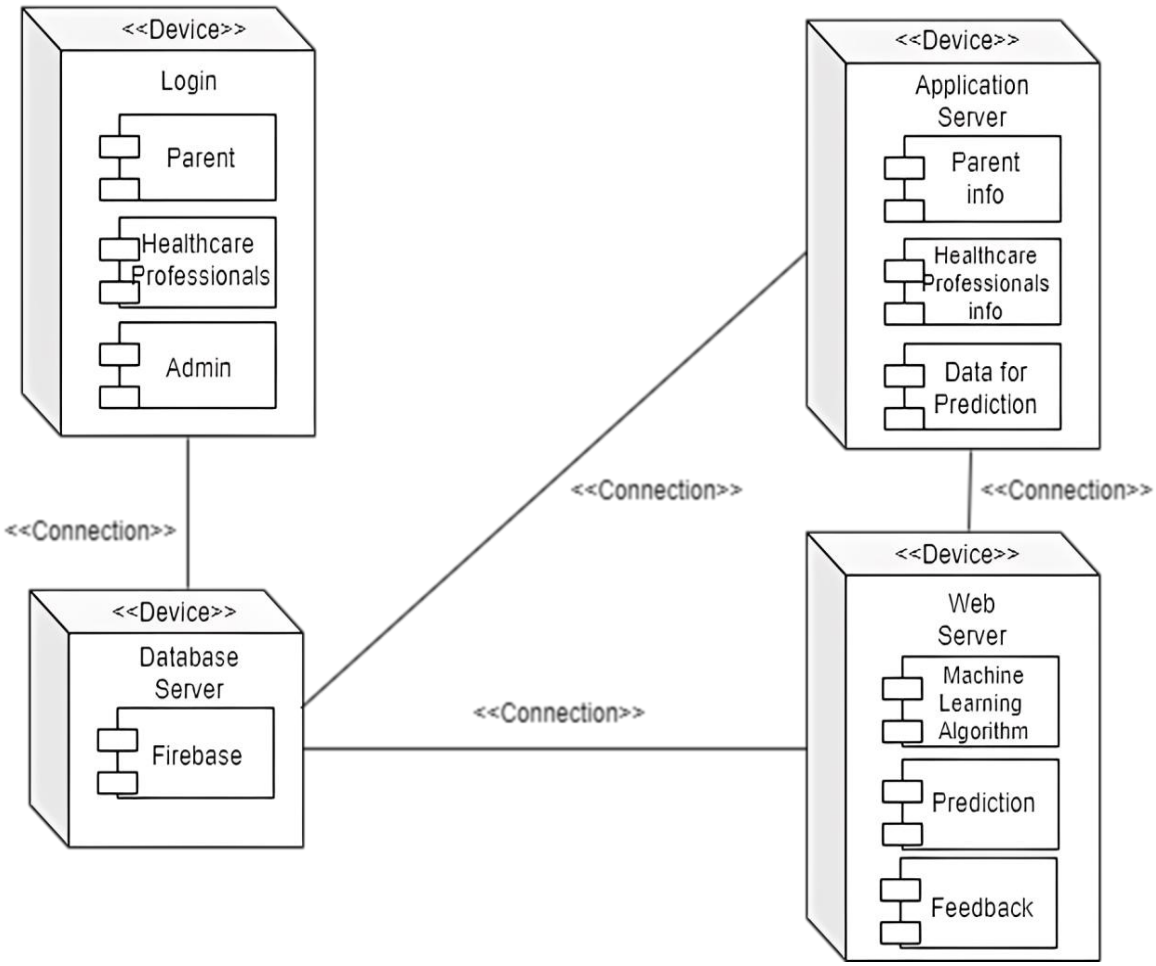

Fig 4.5: Deployment Diagram

4.2.2. UML Behavioral Diagrams

UML Behavioral Diagrams offer a clear visual representation of dynamic system behavior, interactions among components and entities, and task accomplishment. They aid in comprehending functionality, message passing, and state transitions, promoting efficient system design and stakeholder communication.

4.2.2.1. Activity Diagrams:

The flow of activities within a system, illustrating the sequence of actions or steps required to accomplish a specific task or behavior.

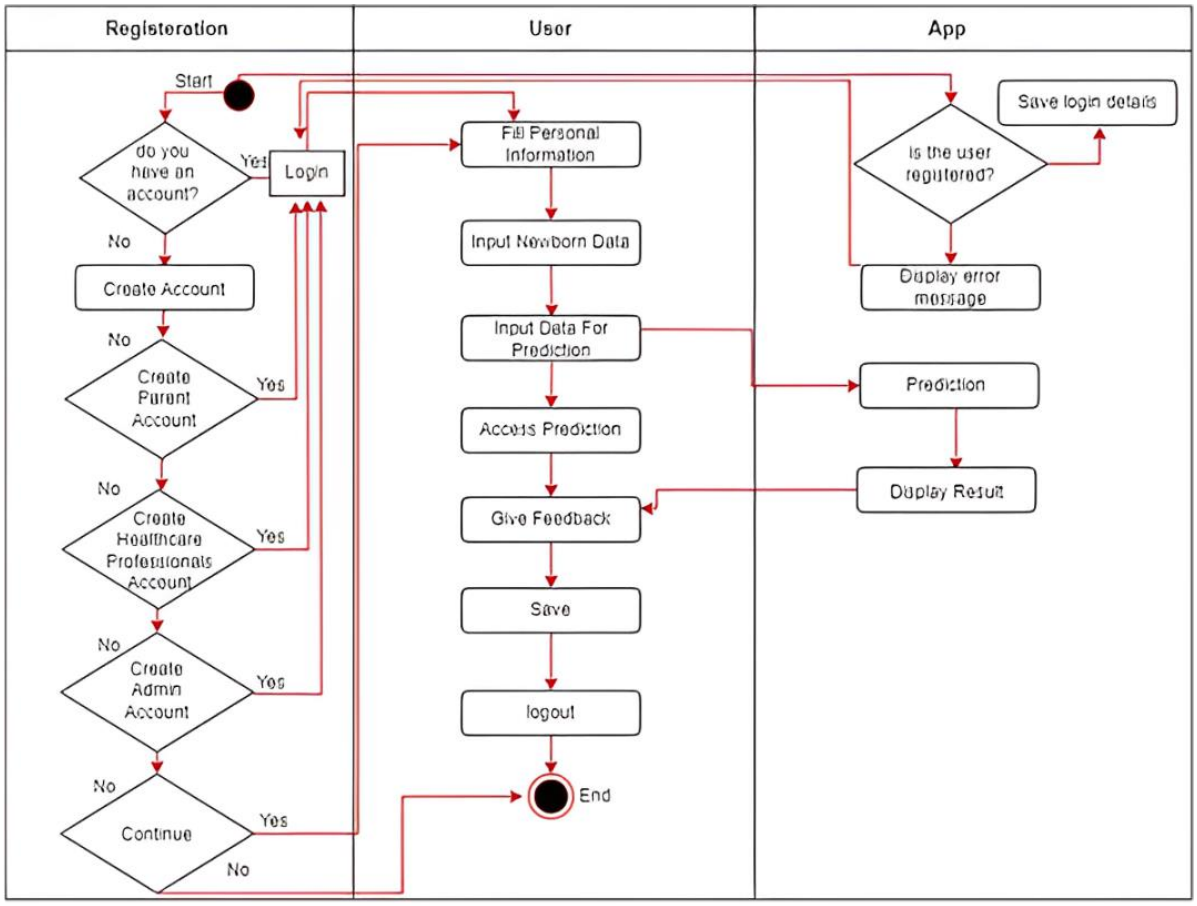

Fig 4.6: Activity Diagram

4.2.2.2. State Machine Diagrams:

This diagram models the behavior of individual components or objects within the system, depicting their states and transitions between them. It offers a visual representation of how the system responds to events and stimuli, guiding the understanding of its dynamic behavior.

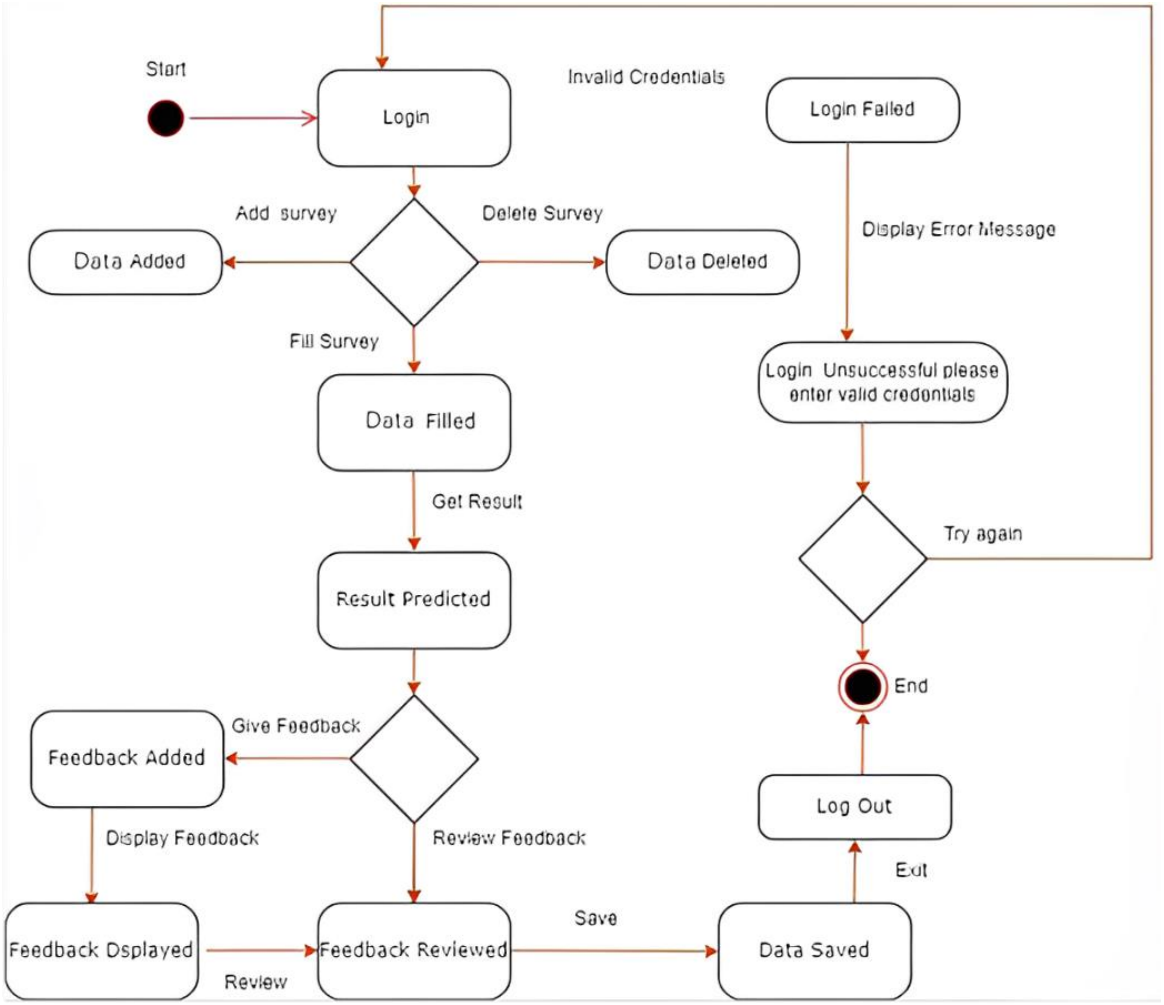

Fig 4.7: State Machine Diagram

4.2.3. UML Interaction-diagrams

UML Interaction Diagrams illustrate dynamic system behavior, emphasizing object interactions during runtime. Sequence Diagrams, showing message flow chronologically.

4.2.3.1. Sequence Diagrams

Sequence diagrams in UML illustrate the interactions between objects or components in a system over time, depicting the flow of messages between them.

4.2.3.1.1. Registration sequence Diagram

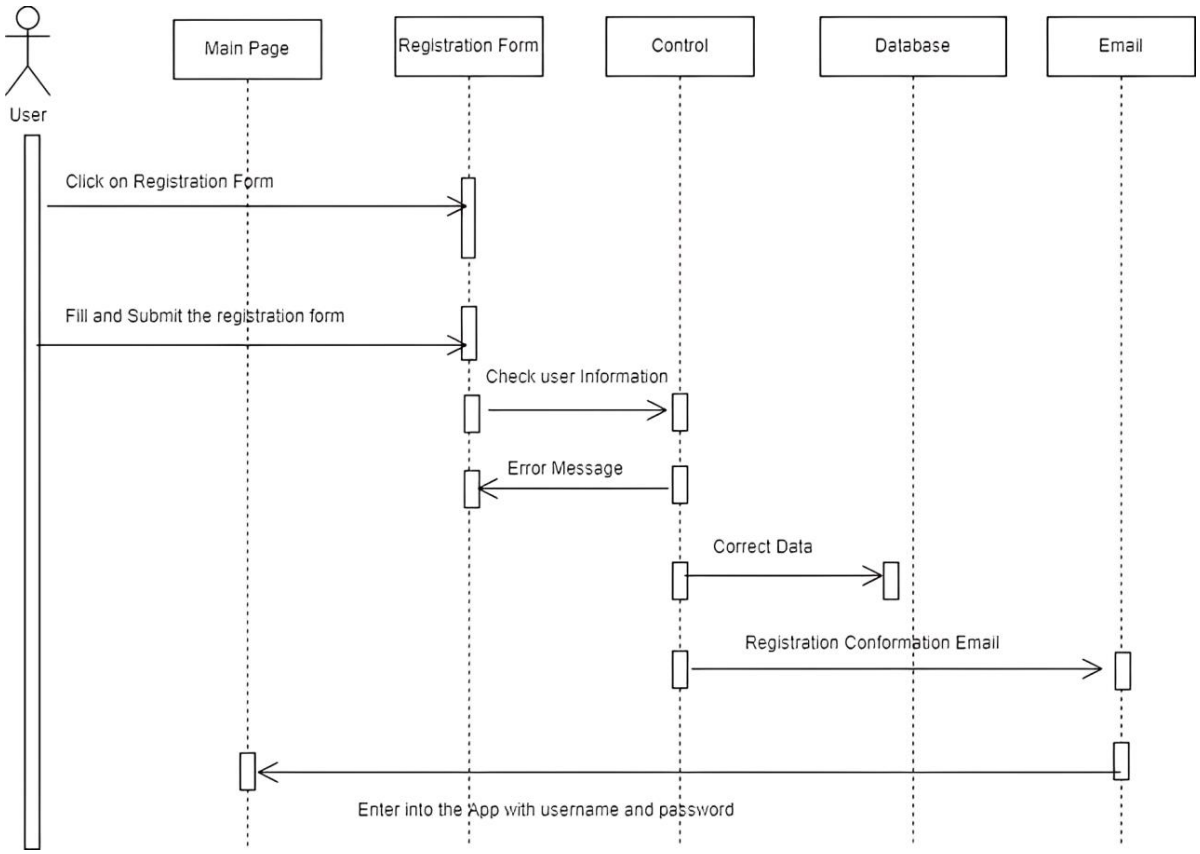

Fig 4.8: Register Sequence Diagram

4.2.3.1.2. Login Sequence Diagram

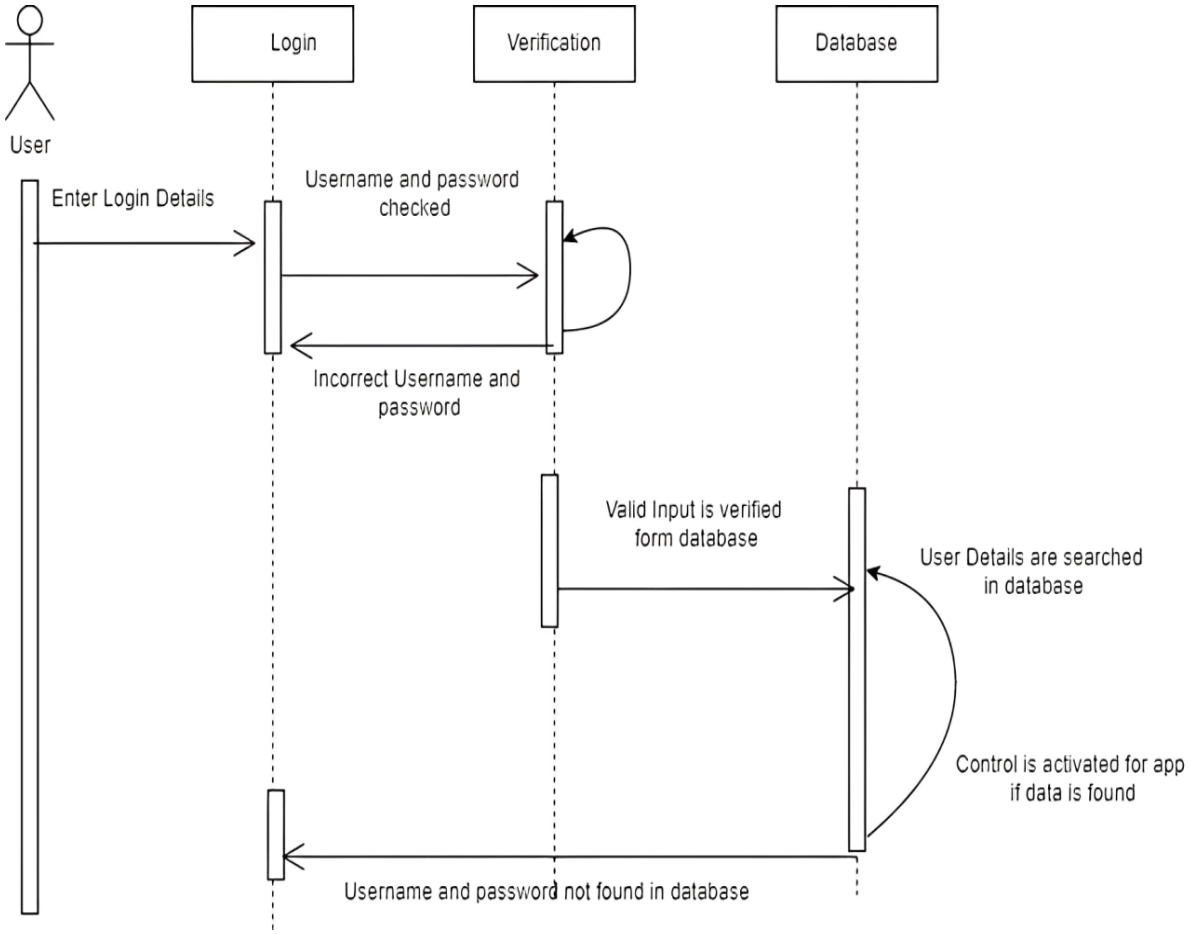

Fig 4.9: Login sequence diagram

4.2.3.1.3. User Activity Sequence Diagram

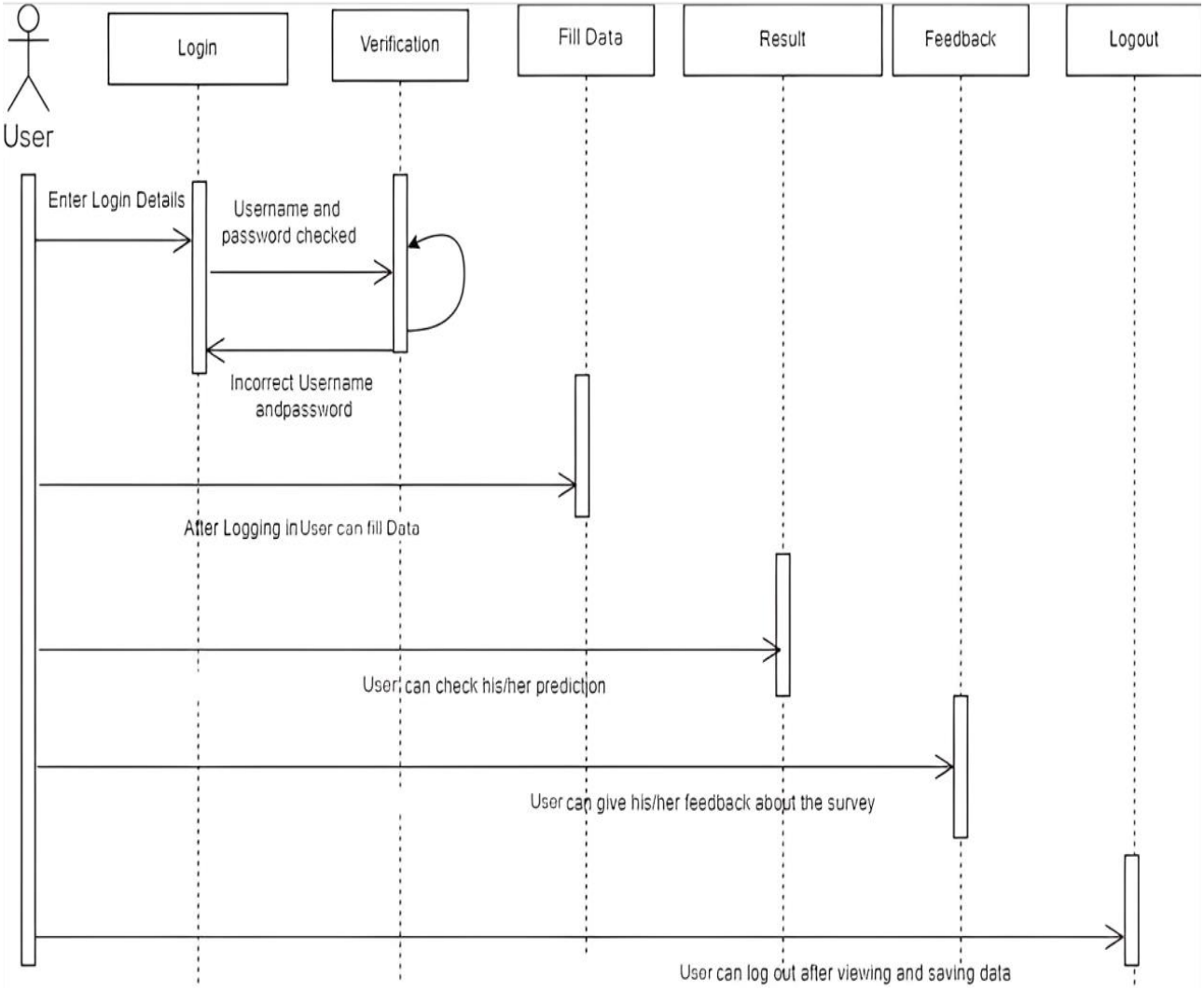

Fig 4.10: User Sequence Diagram

4.2.4. Node Structure

Node structure refers to the arrangement and organization of nodes within a network or hierarchical system, defining their relationships and connectivity. It establishes the framework for data transmission, processing, and communication across the network.

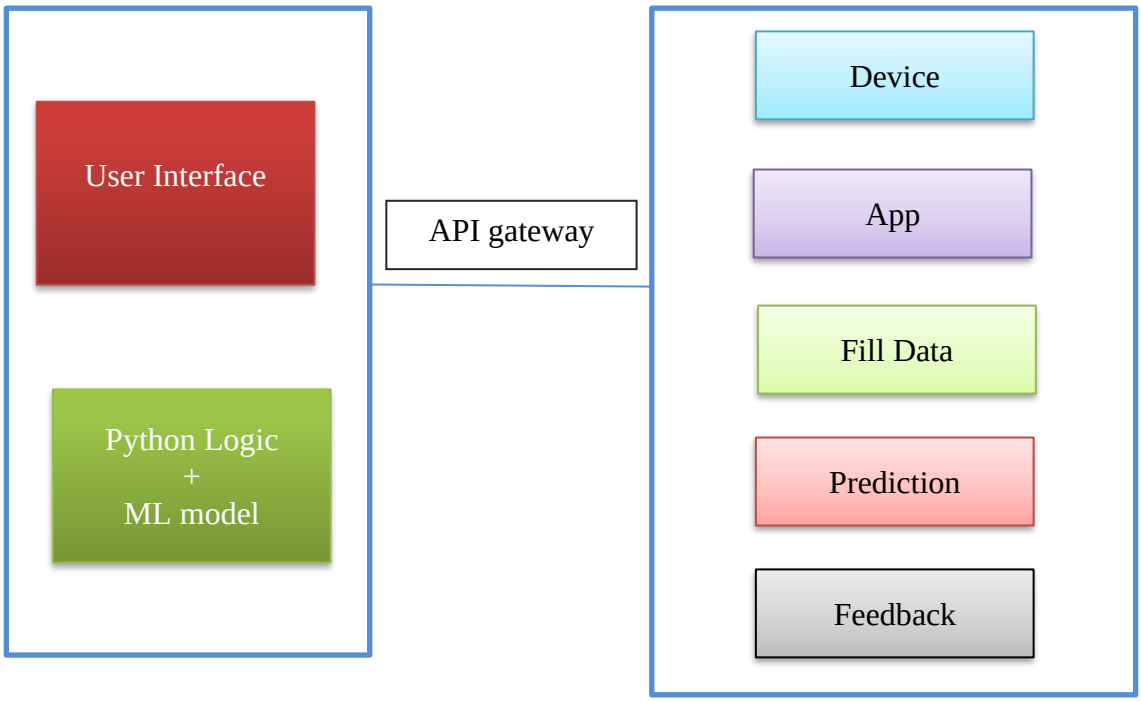

Fig 4.11: Node Structure Diagram

4.2.5. Communication Design Protocol

Communication Design Protocol is the exchanging of information. It defines protocols, formats, and procedures to ensure seamless communication within the system architecture.

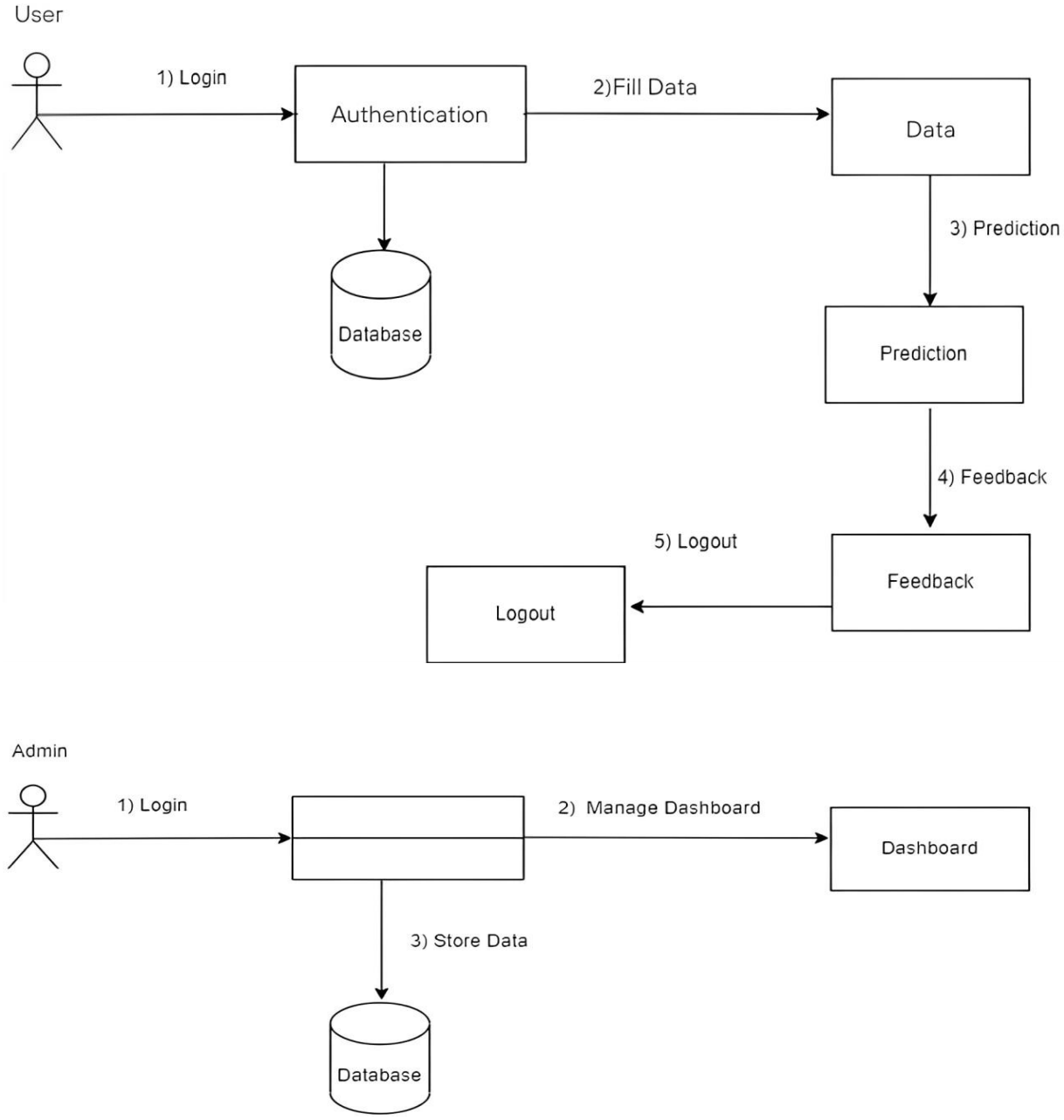

Fig 4.12: Communication Design Protocol

Chapter 5: Implementation

This chapter provides a comprehensive insight into the architecture, communication protocols, and user interaction mechanisms utilized within the Newborn Disease Prediction System through healthcare data analysis. It outlines the component diagram, network infrastructure, protocol selection, middleware choice, and user interface design to offer a thorough understanding of the Newborn Disease Prediction System of the application.

5.1. Component Diagram

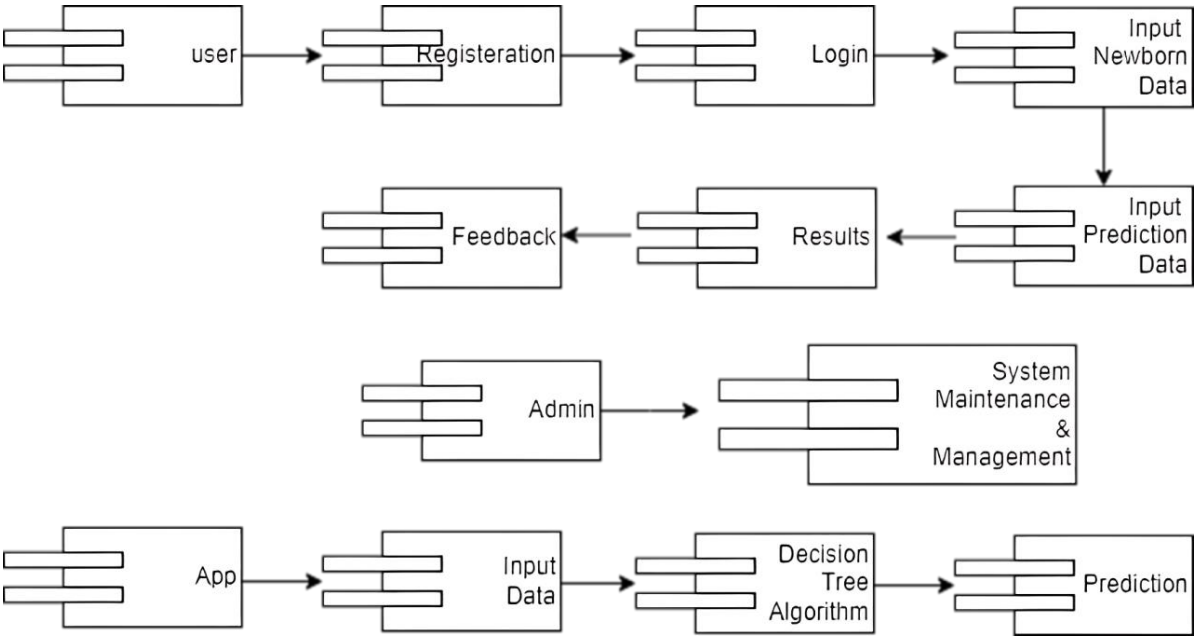

Fig 5.1: Component Diagram

5.1.1. User Component:

In the Newborn Disease Prediction System (NDPS), the user component refers to healthcare professionals, such as pediatricians or neonatologists, who interact with the platform to predict and manage potential health risks for newborns. These users can access patient data, input clinical observations, and receive predictive insights and recommendations from the system to aid in early intervention and preventive care for newborns.

- **Login Page:** Users can access the system by entering their credentials, ensuring secure authentication.
- **Registration Page:** New users can easily create an account, providing necessary information for registration.
- **Dashboard:** Users are presented with a comprehensive overview of their account details, recent activities, and available options for navigation.
- **Survey Form:** Users input specific newborn data, which serves as the basis for disease prediction, ensuring accurate results.
- **Prediction:** The system generates predictions based on the input data, offering valuable insights into potential health risks for newborns.
- **Recommendation:** Based on predicted results, users receive tailored recommendations for consulting relevant healthcare professionals, ensuring proactive health management.
- **Book an appointment:** Users can conveniently schedule appointments with recommended doctors directly from the platform, streamlining the healthcare process.
- **Details of Appointment:** After booking, users can access comprehensive details about their appointments, including date, time, and location, facilitating smooth communication and coordination with healthcare providers.

5.1.2. Admin Component:

The Admin Component facilitates comprehensive system management, empowering administrators with tools for overseeing operations efficiently. It enables access to doctor chat messages, doctor management functionalities like addition and deletion, and monitoring of popular doctors based on user feedback. Additionally, administrators handle feedback management and analyze user survey results to drive continuous system improvement.

- **Admin Component:** Enables administrative access to manage various aspects of the system.

- **Access to Doctor Chat Messages:** Admins can view and monitor conversations between doctors and users, ensuring compliance with communication guidelines and addressing any issues promptly.
- **Add/Delete Doctors:** Admins have the authority to add new doctors to the system, ensuring the availability of qualified healthcare professionals. They can also remove doctors when necessary, maintaining an updated database.
- **Popular Doctor Management:** Admins can track and manage the popularity of doctors based on user feedback and ratings, ensuring the visibility of top-rated healthcare providers.
- **Feedback Management:** Admins can review and analyze user feedback, gaining insights into the user experience and making necessary improvements to enhance system functionality.
- **User Survey Results:** Admins have access to survey results submitted by users, allowing them to analyze trends, identify areas for improvement, and make data-driven decisions to optimize system performance.

5.1.3. Doctor Component:

The Doctor Component empowers doctors with essential functionalities for managing appointments seamlessly. Doctors can efficiently check, cancel, or reschedule appointments as needed, ensuring optimal scheduling. Additionally, they have access to a chat feature enabling direct communication with administrators for any inquiries or assistance required, facilitating effective collaboration and support within the system.

- **Check Appointment:** Doctors can review their upcoming appointments, allowing them to prepare adequately and manage their schedule efficiently. This feature ensures that doctors are aware of their patient appointments and can plan their day accordingly.
- **Cancel or Reschedule:** In case of emergencies or scheduling conflicts, doctors have the ability to cancel or reschedule appointments as necessary. This flexibility ensures that patients receive timely notifications and can adjust their plans accordingly, minimizing inconvenience.

- **Chat with Admin:** The chat feature enables direct communication between doctors and administrators within the system. Doctors can seek clarification, discuss patient cases, or address any administrative issues promptly and efficiently. This real-time communication streamlines the workflow and fosters effective collaboration between medical staff and administrative personnel.

5.1.4. Database Component:

The system employs Firebase for user authentication and data storage. Firebase Authentication manages user login and registration, while Firestore stores user profiles, survey results, appointments, chat messages, and feedback. This combination ensures secure access and efficient data management.

5.1.5. Notification Component:

The notification system is designed to inform doctors promptly about new appointments, any alterations to their schedule (like cancellations or reschedules), and incoming messages from the admin. These notifications facilitate timely updates on patient appointments and streamline communication between doctors and the administrative team. Additionally, the system offers reminders for upcoming appointments and essential tasks, enhancing time management and patient care efficiency.

5.2. Network and Protocol Choice:

The network and protocol choice in the system design using HTTP(S) with RESTful APIs and JSON for data exchange, you can ensure compatibility, flexibility, and security in communication between your Flutter frontend and Python backend.

- **Network Protocol:** HTTP (Hypertext Transfer Protocol) or HTTPS (HTTP Secure) are commonly used for transmitting data between the Flutter frontend and Python backend. HTTP is suitable for most communication needs, while HTTPS adds an extra layer of security through encryption.
- **Communication Protocol:** RESTful APIs (Representational State Transfer) are often used with HTTP(S) for communication between frontend and backend. RESTful APIs

provide a standardized way of accessing and manipulating resources over the web and are well-supported by both Flutter and Python.

- **Data Exchange Format:** JSON (JavaScript Object Notation) is a popular choice for data exchange between Flutter and Python. JSON is lightweight, easy to parse, and widely supported by both frontend and backend frameworks.

5.3. Choice of Object Middleware

For a newborn disease prediction mobile app using Flutter frontend and Python backend, selecting the right middleware is pivotal. A RESTful API architecture, implemented with Flask or Django in Python, emerges as a suitable choice. RESTful APIs provide a standardized approach for designing web services, facilitating seamless communication between frontend and backend components. With Flask or Django, you can create robust APIs, handling HTTP requests, routing, data serialization, and database interactions efficiently. This approach ensures a clear separation of concerns, allowing frontend and backend to evolve independently while ensuring interoperability with various technologies.

5.4. User Interface

The user interface (UI) of the newborn disease prediction mobile app should prioritize simplicity and intuitiveness, enabling easy navigation and interaction for users. A clean and visually appealing design with clear data visualization elements enhances user understanding and engagement with the app's predictive functionalities.

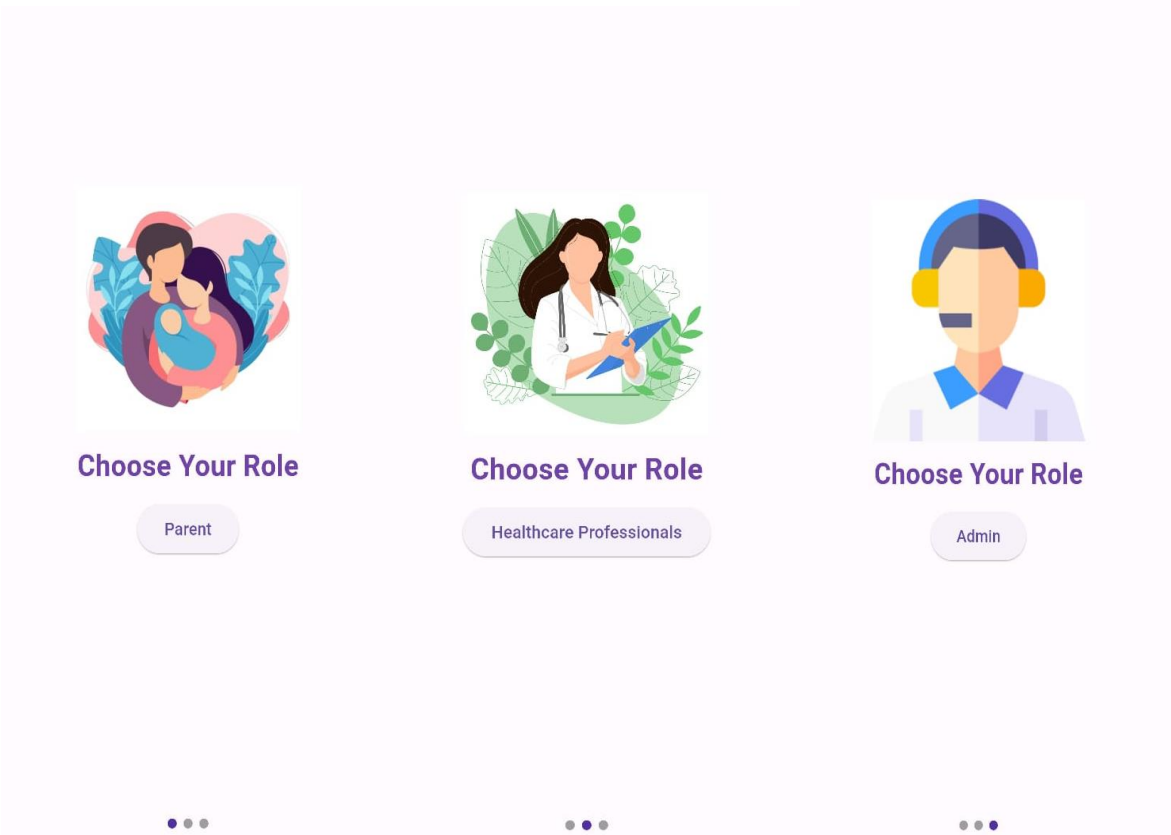

Fig 5.2: Choose your Role

The user selects their role from three options: Parent, Healthcare Professional, or Administrator. As a Parent, they focus on managing and supporting their child's health and well-being. As a Healthcare Professional, they provide medical care and guidance to patients. As an Administrator, they oversee and manage healthcare systems and operations.

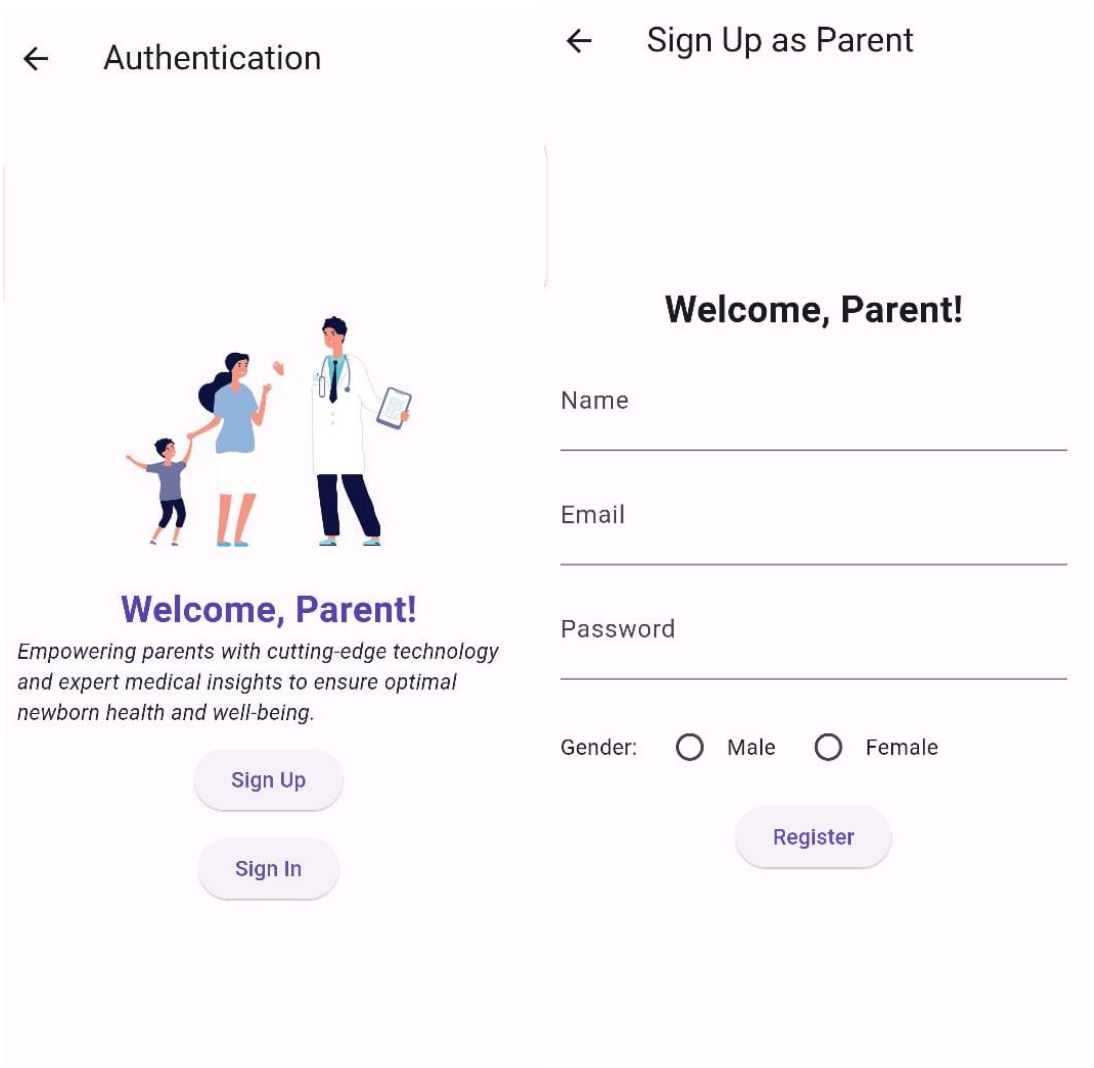

Fig 5.3: Role Parent

The user selects the Parent role, allowing them to manage and support their child's health and well-being through the platform. Once registered, they can access various tools and resources tailored for parents.

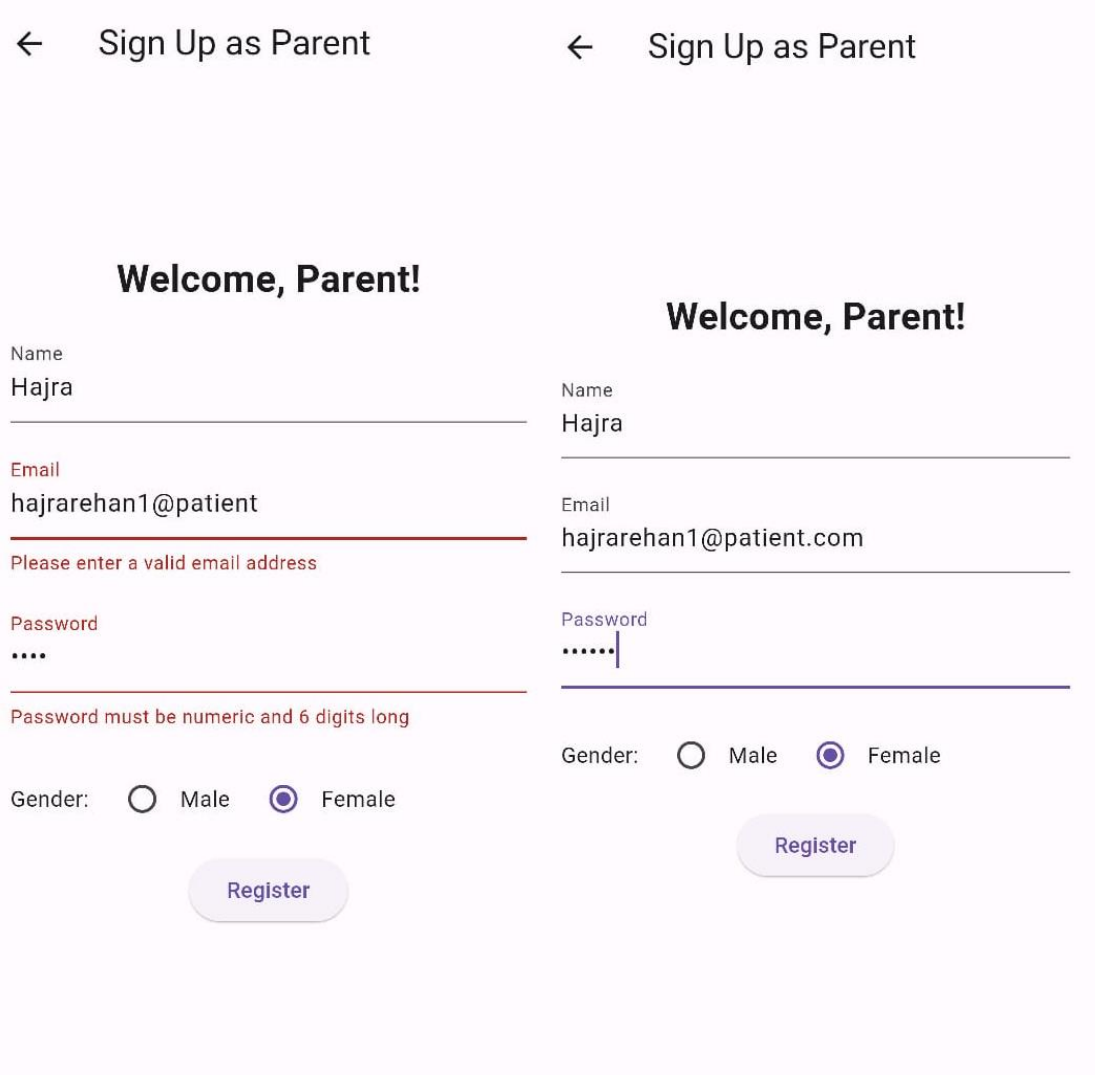

Fig 5.4: Sign Up as Parent

The user selects the Parent role and then has the option to sign up. To sign up, they must provide a valid email address (e.g., example@domain.com) and create a password that meets the requirements: at least 6 characters long, including one uppercase letter, one lowercase letter, one number, and one special character. Once registered, they can manage and support their child's health and well-being through the platform.

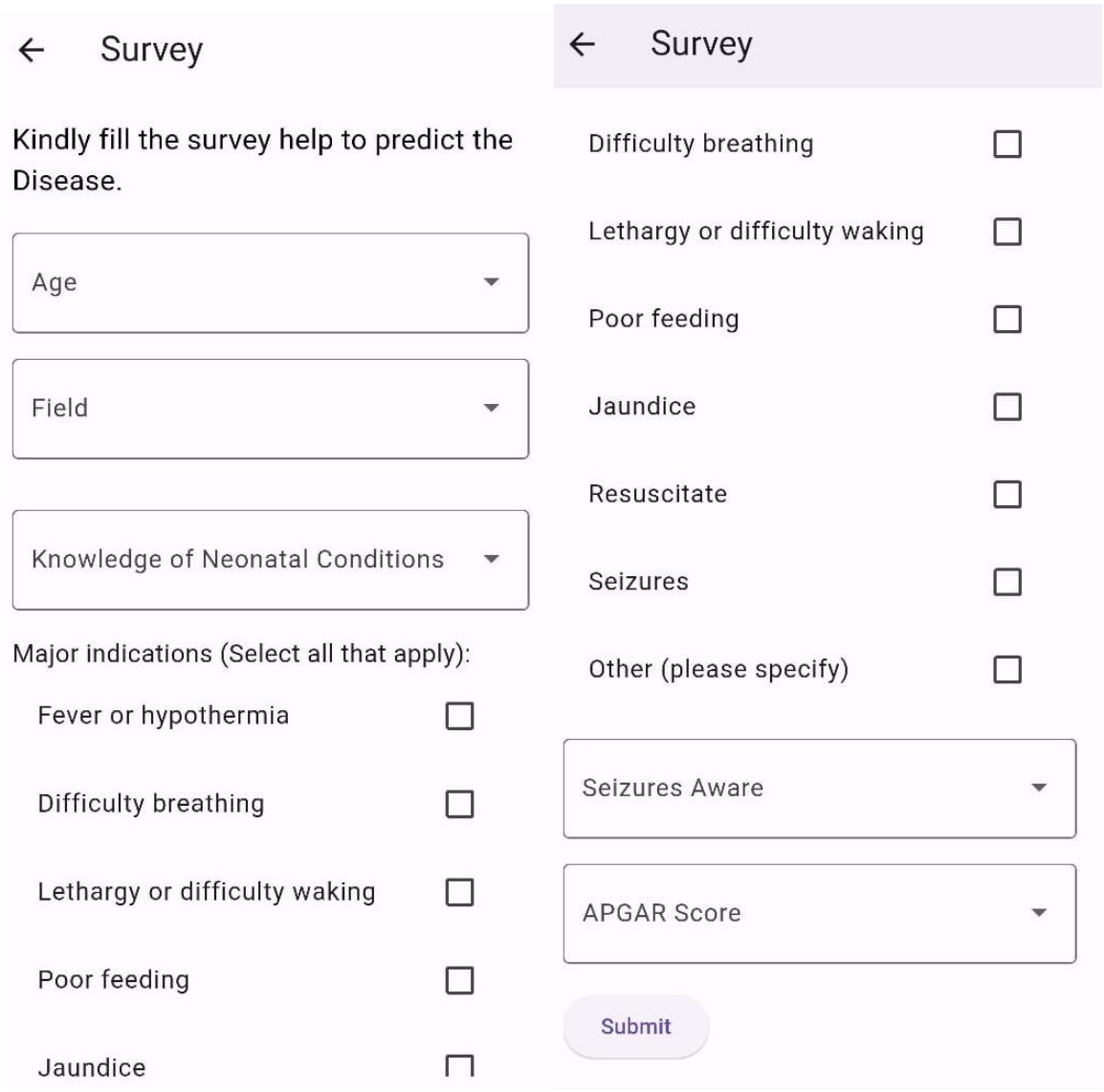

Fig 5.5: Survey Form

The Parent views the survey form, which contains medical-related questions designed to help predict their child's health needs. By completing the form, parents provide valuable information that aids in assessing and managing their child's well-being.

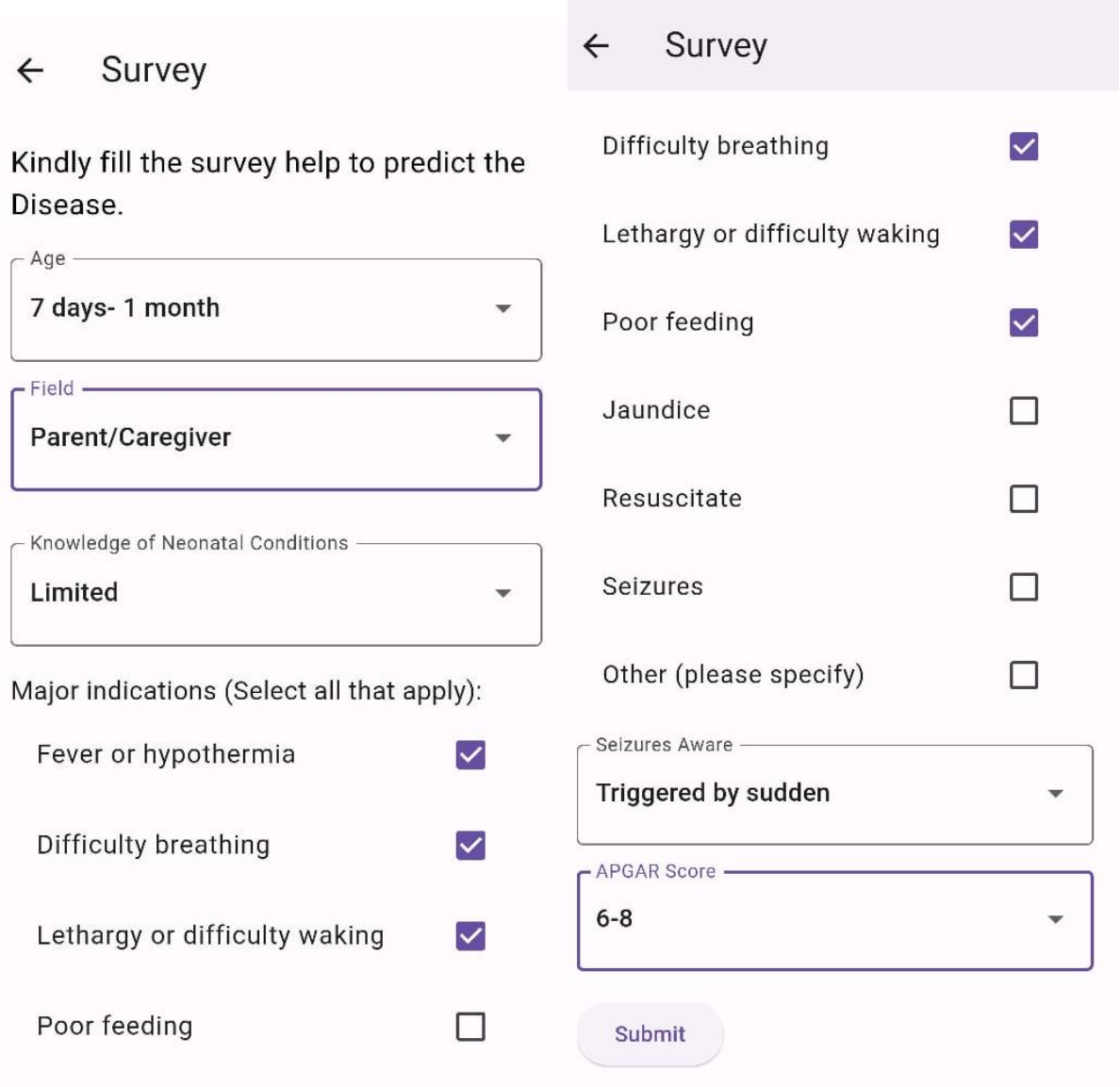

Fig 5.6: Filling Survey

The Parent views the survey form, which contains medical-related questions designed to help predict their child's health needs. The Parent must completely fill out all fields with correct information to receive the predicted results of potential diseases. This ensures accurate assessment and management of their child's well-being.

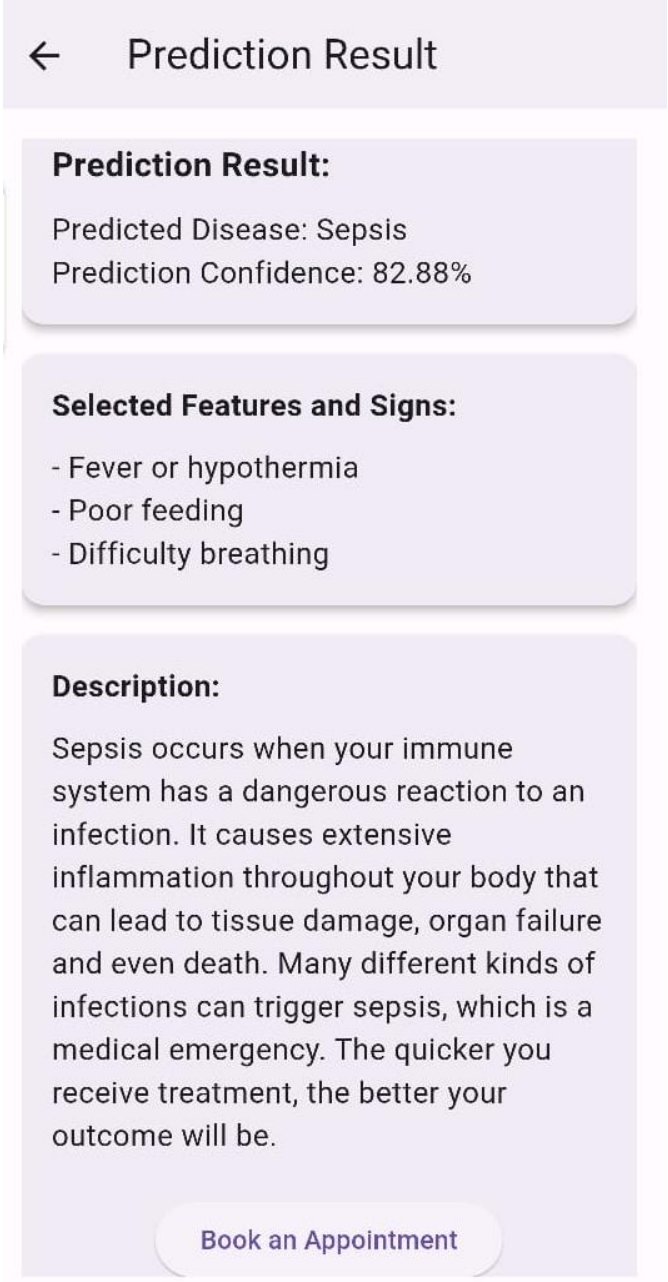

Fig 5.7: Correct Predict Disease

The correct disease is predicted if the Parent accurately fills out all fields with correct information. This ensures reliable assessment and effective management of their child's well-being.

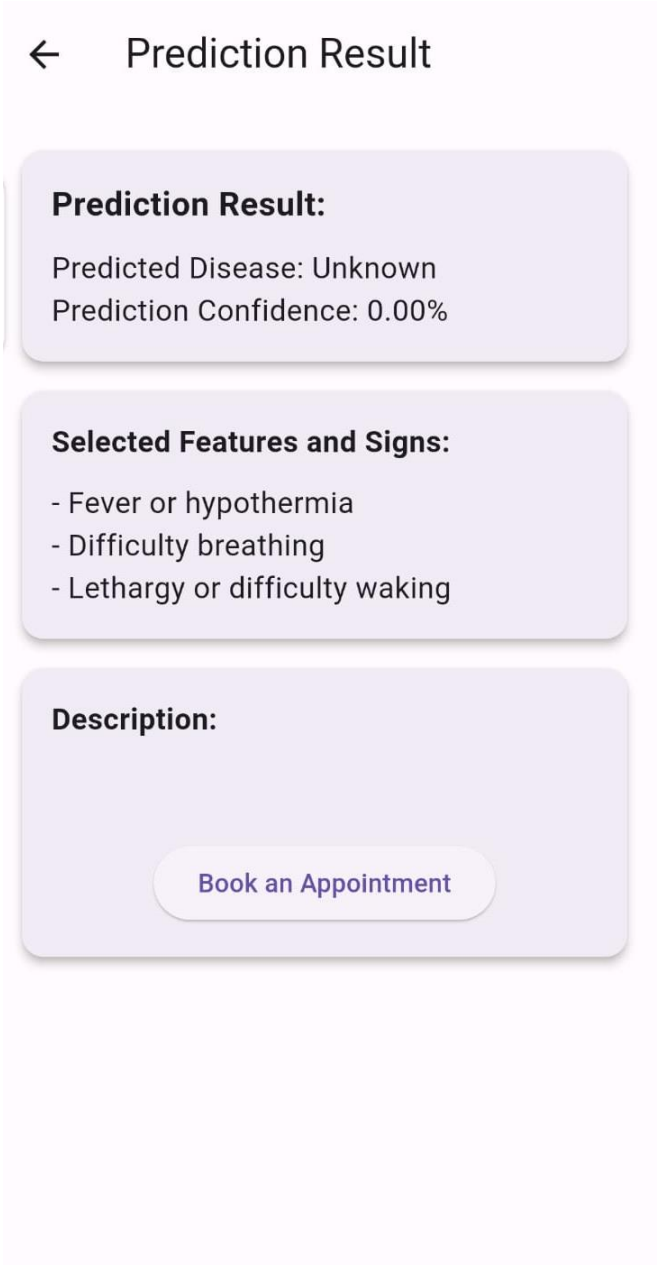

Fig 5.8: Unknown Predict Disease

If the Parent does not fill the survey form correctly, an unknown disease may be predicted, compromising the reliability of the assessment.

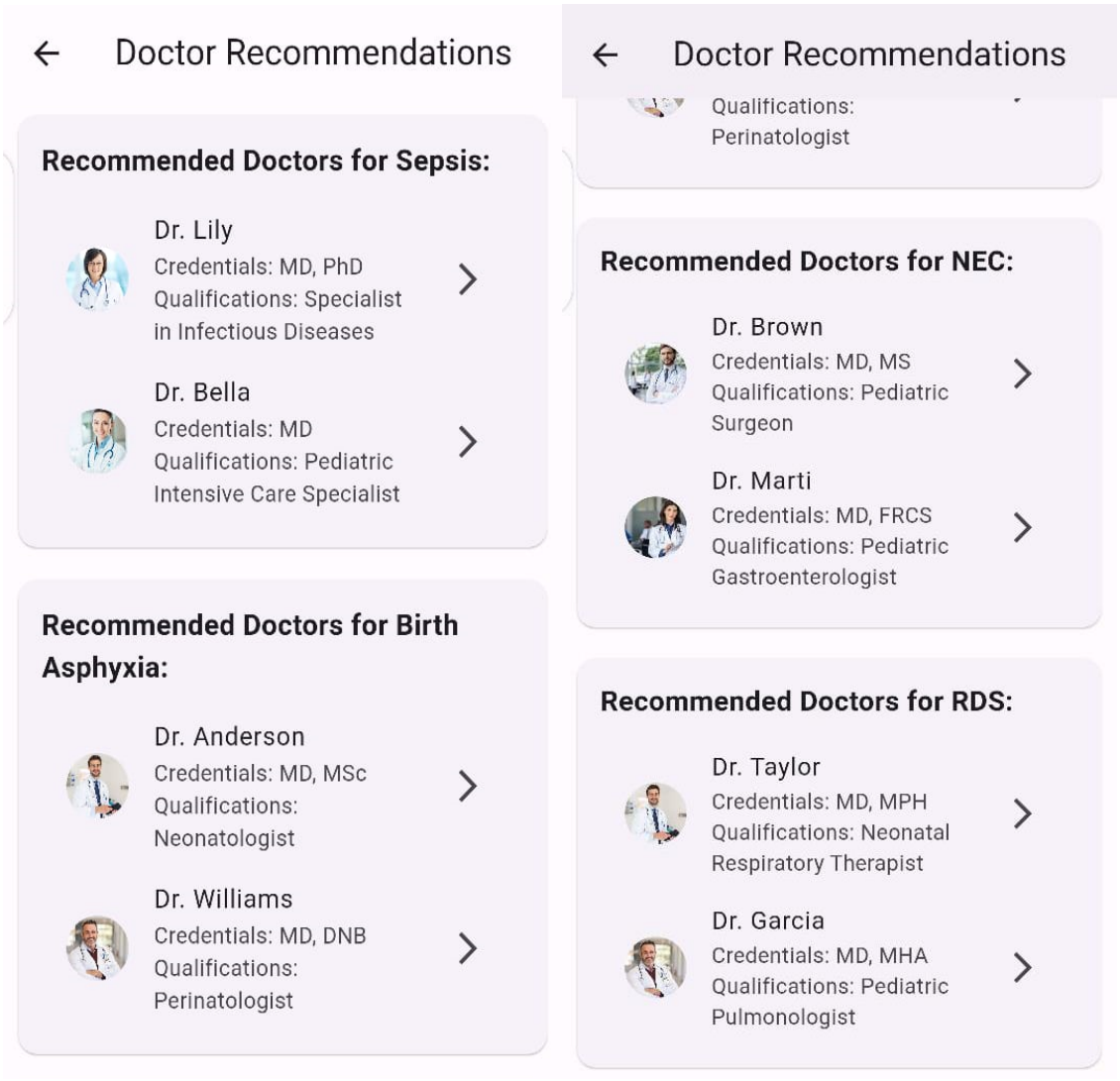

Fig 5.9: Doctor Recommendation for Book an Appointment

Based on the predicted disease, the Parent can access a list of relevant doctors specialized in treating that condition. They can view detailed profiles of these doctors, including their credentials, specialties, and contact information. This enables the Parent to easily reach out to the appropriate healthcare professionals for further consultation and treatment guidance.

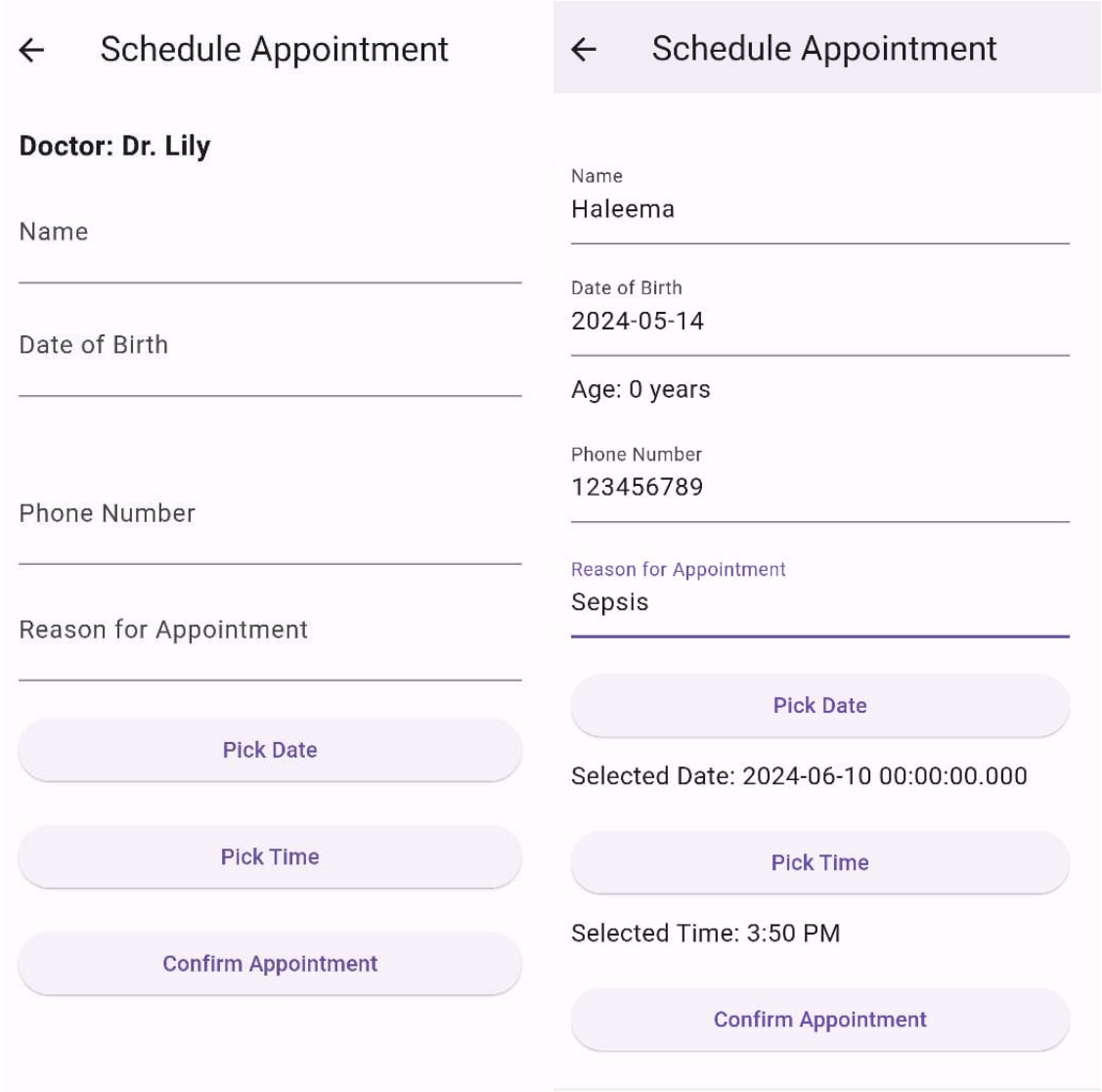

Fig 5.10: Schedule Appointment

Upon receiving the predicted results of their child's potential disease, the Parent can seamlessly book an appointment with the relevant doctor directly through the platform. This streamlined process allows the Parent to quickly schedule a consultation with the healthcare professional specialized in treating the identified condition, ensuring prompt and efficient access to medical care for their child.

Fig 5.11: Appointment Details

Appointment successfully scheduled after the Parent diligently fills out all necessary information accurately. This ensures a smooth process and facilitates effective communication between the Parent and the healthcare provider, guaranteeing that the child receives timely and appropriate medical attention.

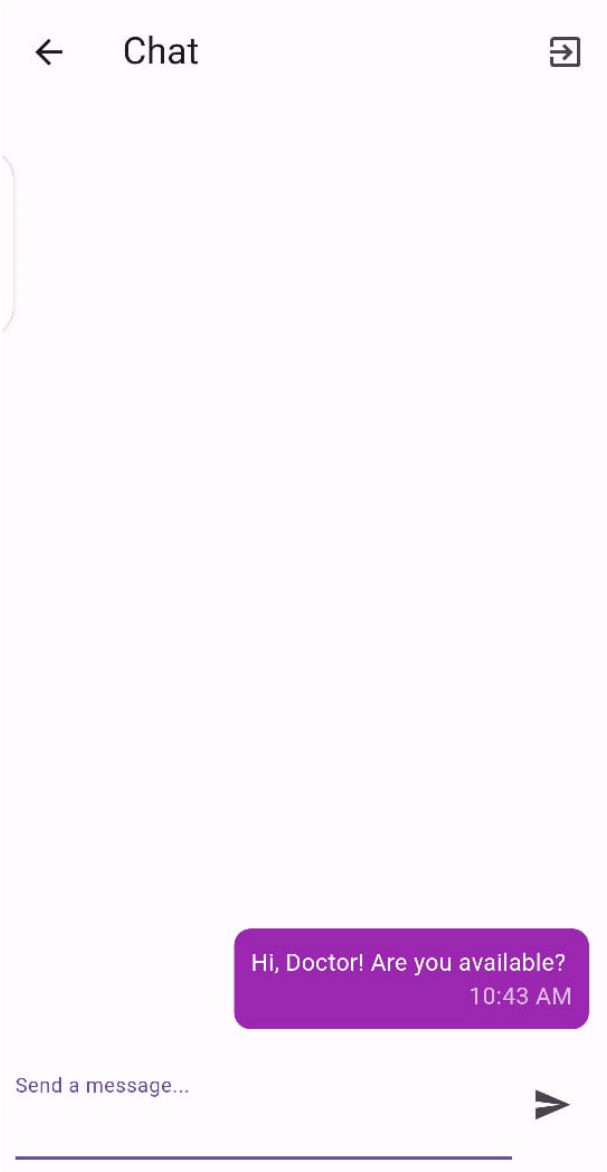

Fig 5.12: Patient Chat with Doctor

The Parent can engage in direct communication with the Doctor to discuss any concerns or questions regarding their child's health. This chat feature is facilitated through Firebase, ensuring secure storage and proper connection of data between the Parent and the Doctor. This integration allows for seamless and confidential exchange of information, enhancing the quality of healthcare communication and support for the Parent.

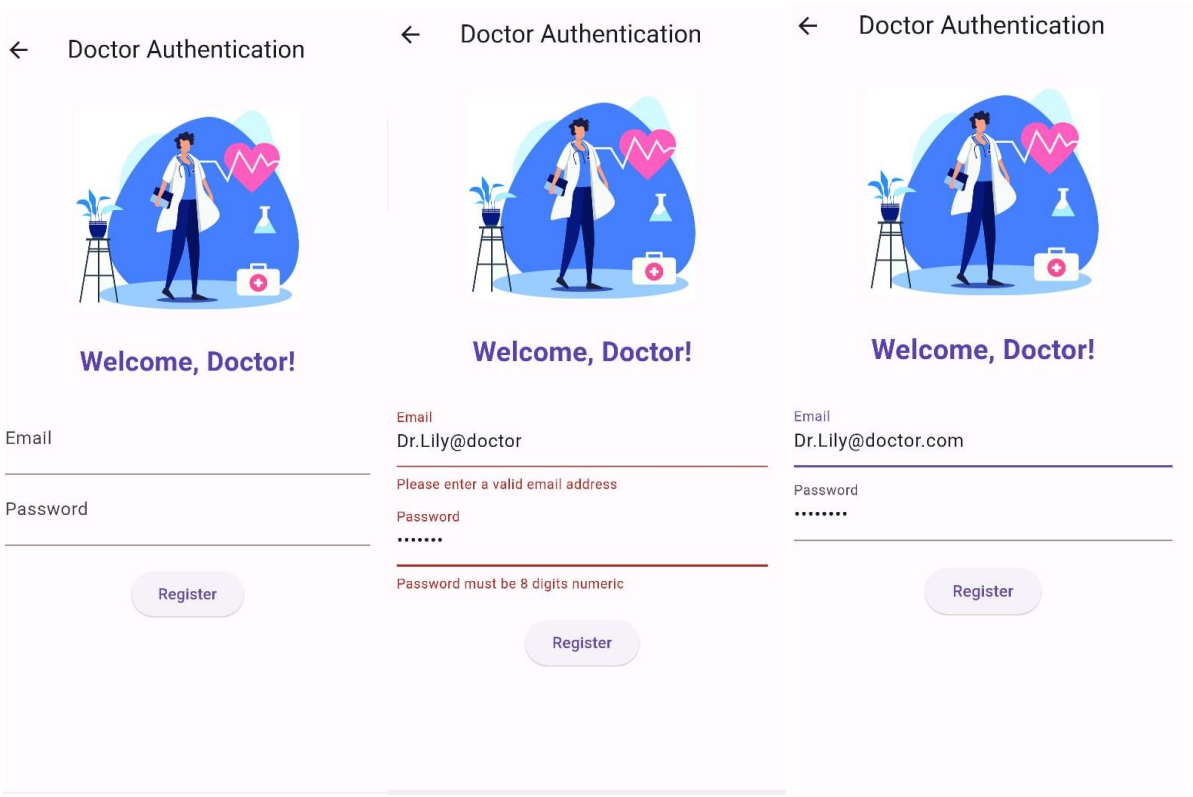

Fig 5.13: Sign up as Doctor Role

The user, selecting the Doctor role, enters their information for authentication via Firebase. They must provide a valid email address (e.g., example@domain.com) and create a password adhering to the required format: at least 8 characters long, including one uppercase letter, one lowercase letter, one number, and one special character. This ensures secure access and protects sensitive healthcare data, maintaining the integrity of the platform.

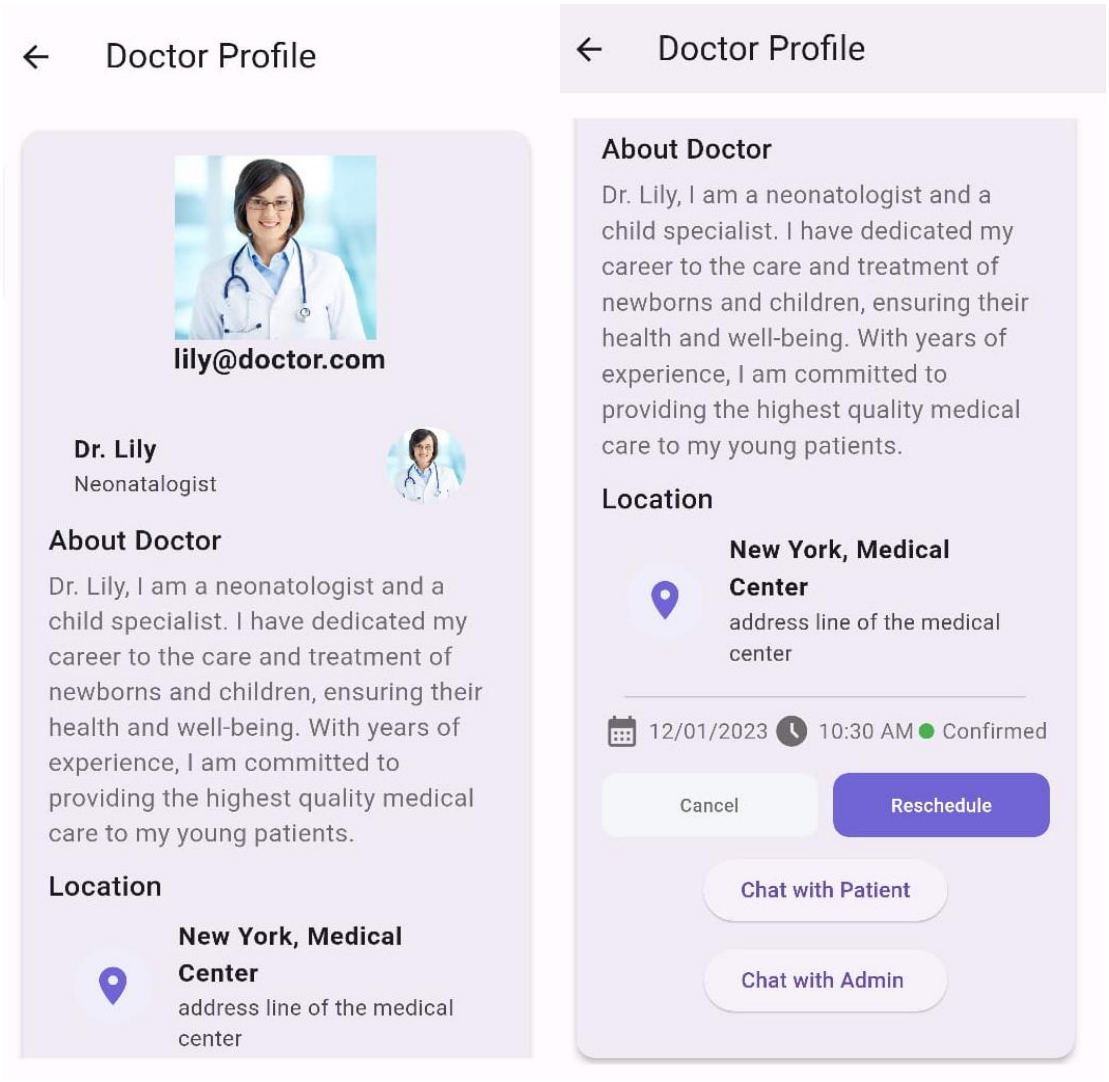

Fig 5.14: Doctor Profile

The Doctor's profile is comprehensive, displaying their details, qualifications, location, and scheduled appointments. This allows patients and administrators to access essential information about the Doctor, ensuring transparency and facilitating informed decision-making. From qualifications to upcoming appointments, all pertinent details are readily available, streamlining the healthcare process and enhancing the patient-doctor relationship.

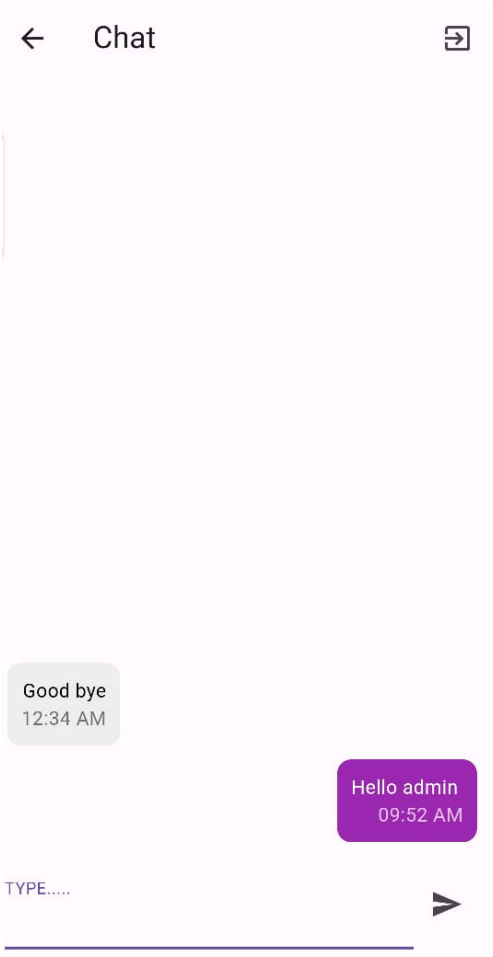

Fig 5.15: Doctor Chat with ADMIN

The Doctor has the capability to engage in direct communication with the Administrator through a dedicated chat feature within the platform. This enables seamless and secure exchange of information, facilitating collaboration, coordination, and efficient management of healthcare services. From discussing administrative matters to seeking assistance or providing updates, this communication channel enhances the overall functionality and effectiveness of the healthcare system.

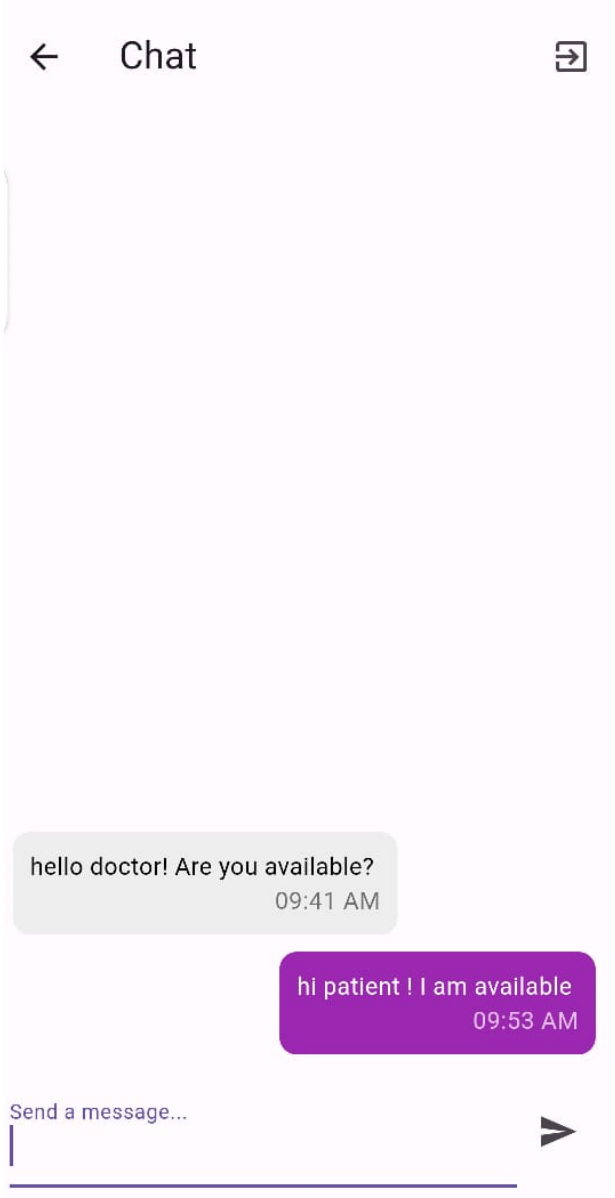

Fig 5.16: Doctor Chat with Patient

The Doctor can initiate conversations with patients using the chat feature, allowing for clear communication regarding appointments and important updates. This includes options to cancel or reschedule appointments as needed, ensuring flexibility and responsiveness to patients' needs. Additionally, they can convey critical information marked as "important," prioritizing timely communication and enhancing the patient-doctor relationship.

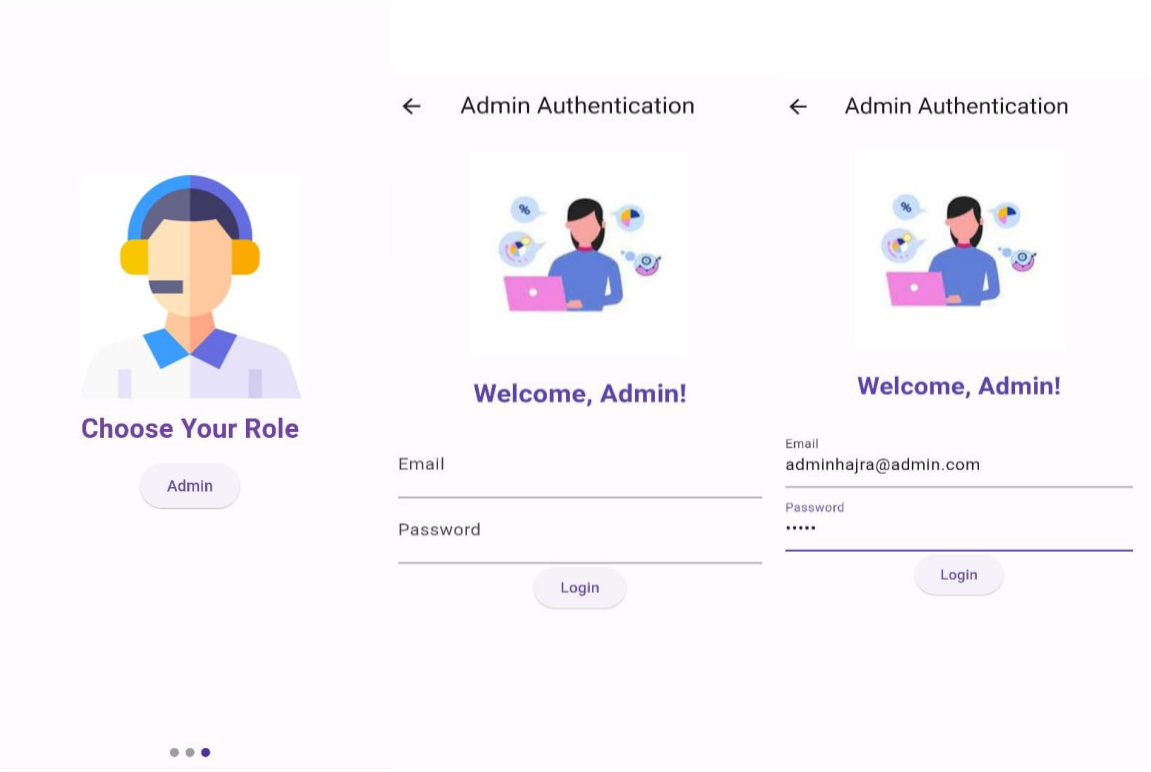

Fig 5.17: Signup as Admin Role

The user selects the Admin role and proceeds to enter their information for login authentication. Once authenticated, the Admin gains access to administrative functionalities within the platform. This includes managing user accounts, overseeing appointments, accessing data analytics, and facilitating communication between healthcare professionals and patients.

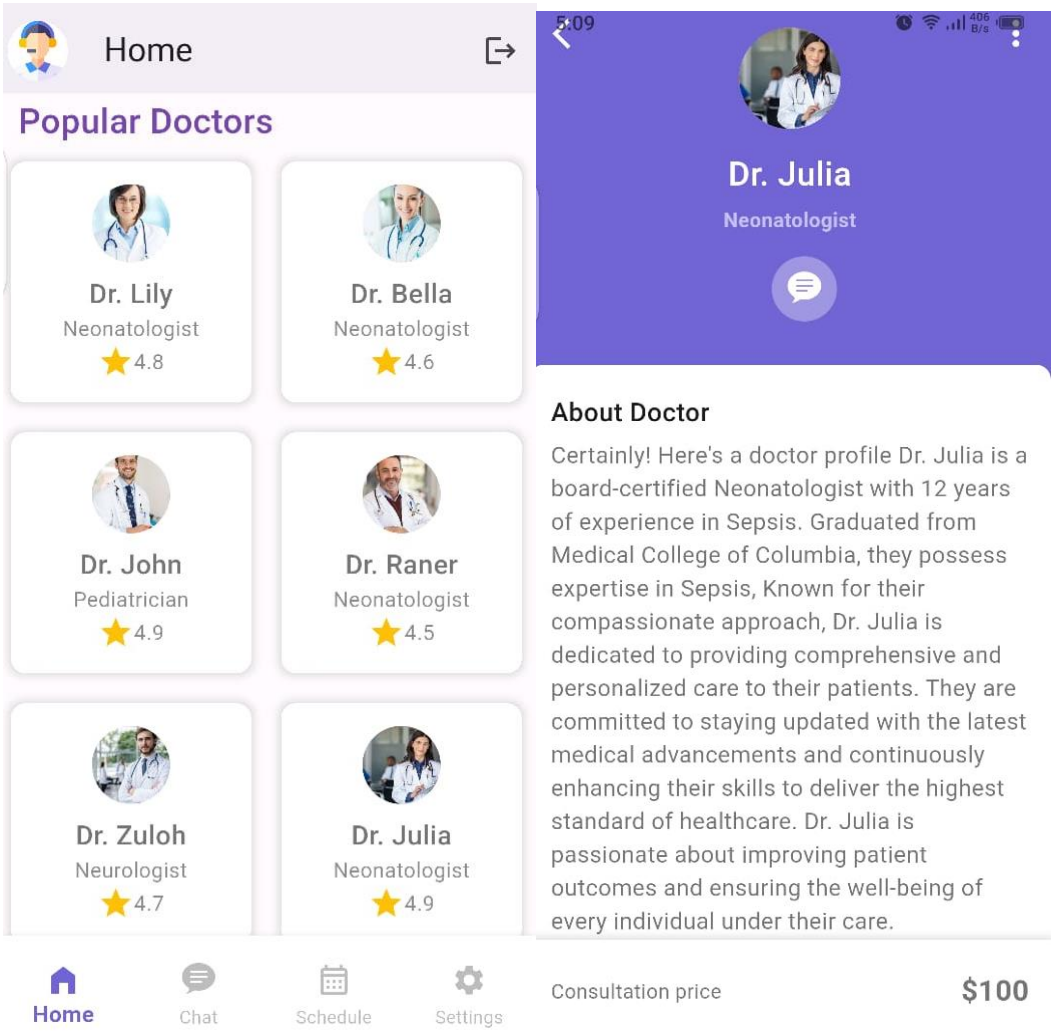

Fig 5.18: Popular Doctors and their profiles

The Admin, equipped with their role, has the privilege to access and view comprehensive details of all doctors registered on the platform. This includes their credentials, qualifications, specialties, and other relevant information necessary for authenticity verification. With this oversight, the Admin ensures the integrity of the healthcare professionals within the application, promoting trust and reliability among users.

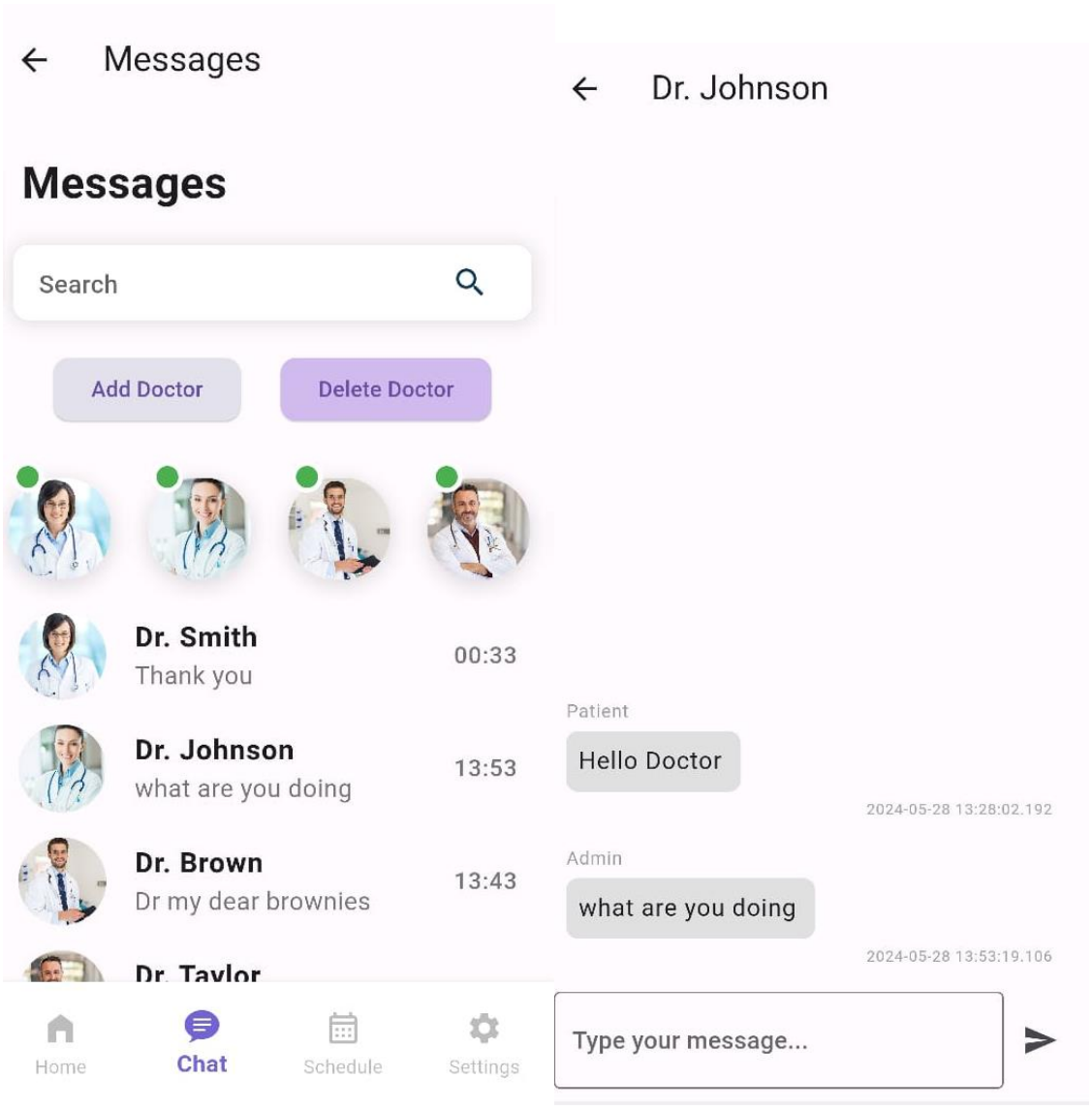

Fig 5.19: Chat with Doctors

The Admin possesses the capability to engage in direct communication with doctors through a dedicated chat feature within the platform. This facilitates seamless communication for administrative purposes, enabling the Admin to convey important announcements, discuss operational matters, or address any inquiries or concerns raised by the doctors. This communication channel enhances coordination and collaboration between the Admin and healthcare professionals, contributing to the efficient functioning of the platform.

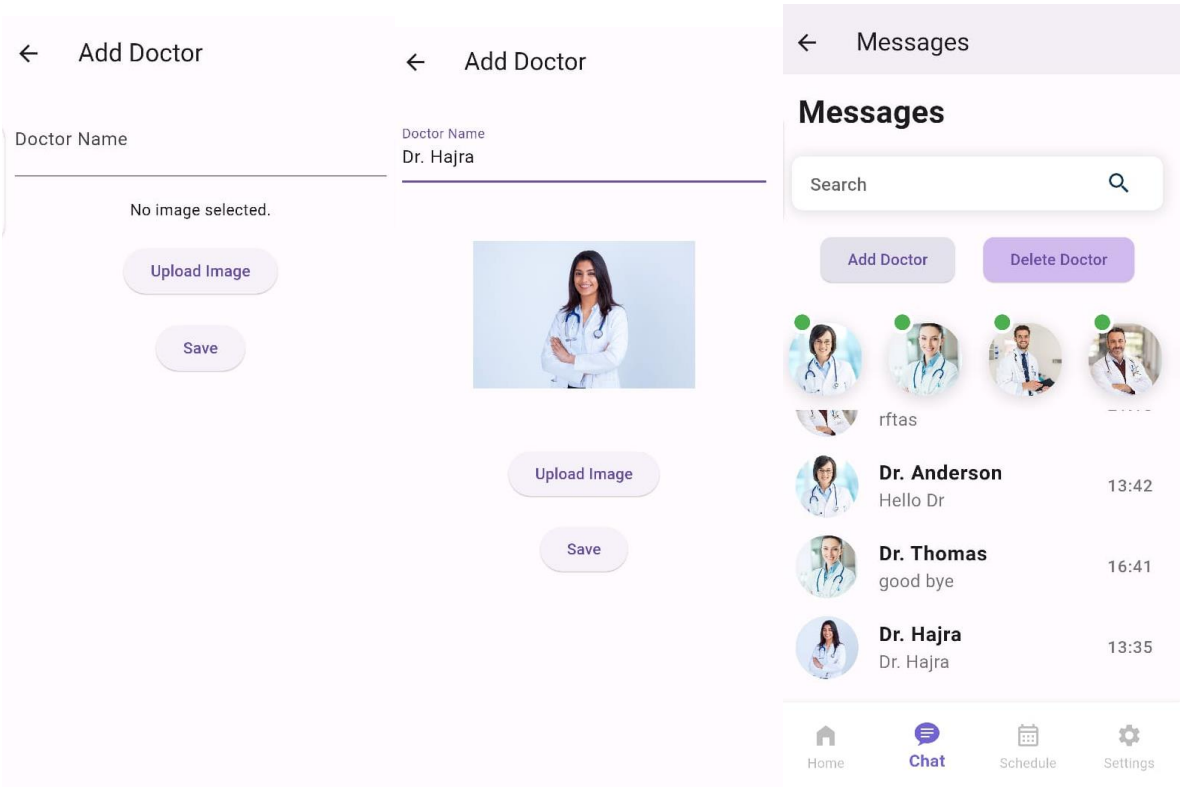

Fig 5.20: Add a new Doctor

Upon clicking the "Add" button, the Admin assumes the authority to include new doctors in the application. This process entails entering the doctor's name and uploading a profile picture, ensuring accurate identification and representation within the platform. By exercising this capability, the Admin expands the pool of healthcare professionals available to users, enriching the platform's services and enhancing its overall functionality.

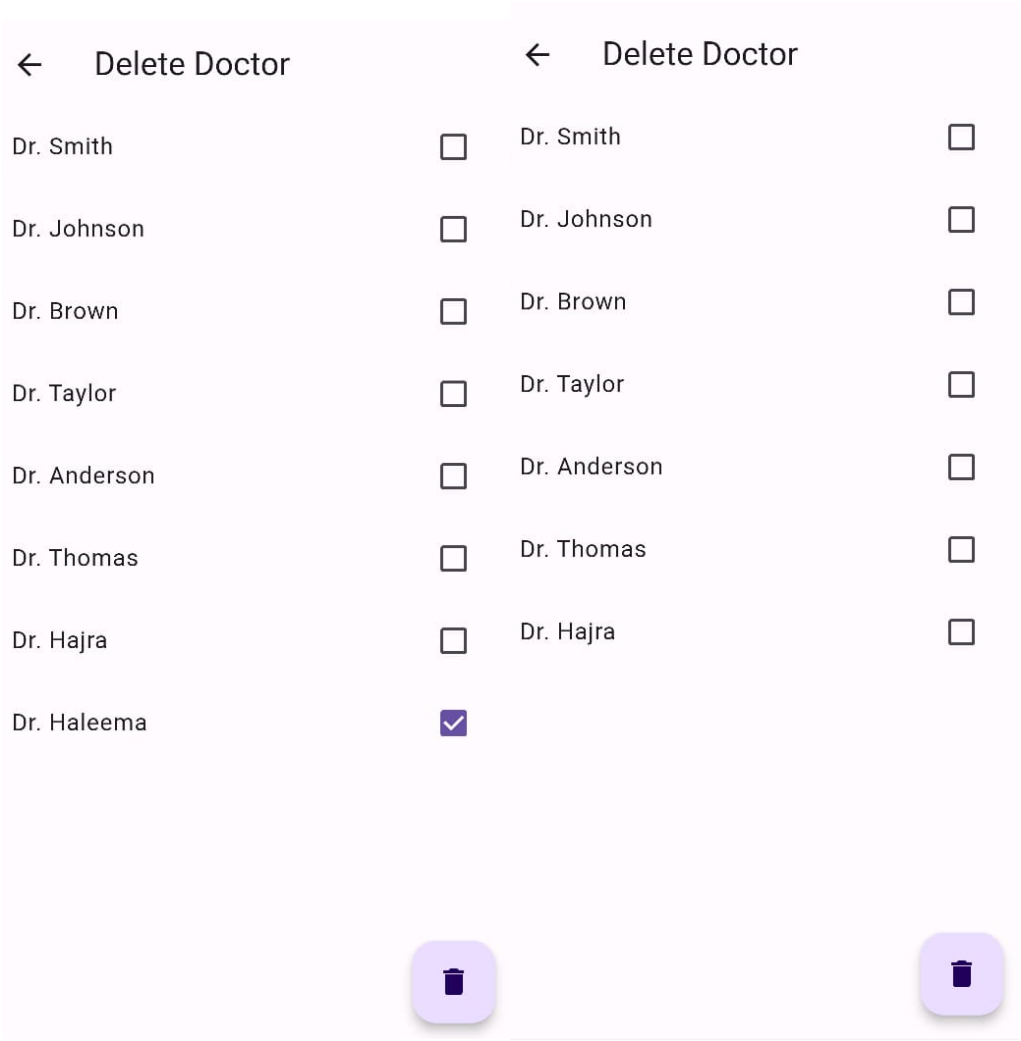

Fig 5.21: Delete a doctor

The Admin holds the authority to remove doctors from the application, ensuring the maintenance of a reliable and trustworthy healthcare network. This capability allows the Admin to manage the roster of healthcare professionals, ensuring that only authorized and qualified doctors remain accessible to users. Whether due to changes in accreditation status or other considerations, the ability to remove doctors underscores the Admin's responsibility for upholding the platform's standards and ensuring the quality of care provided to patients.

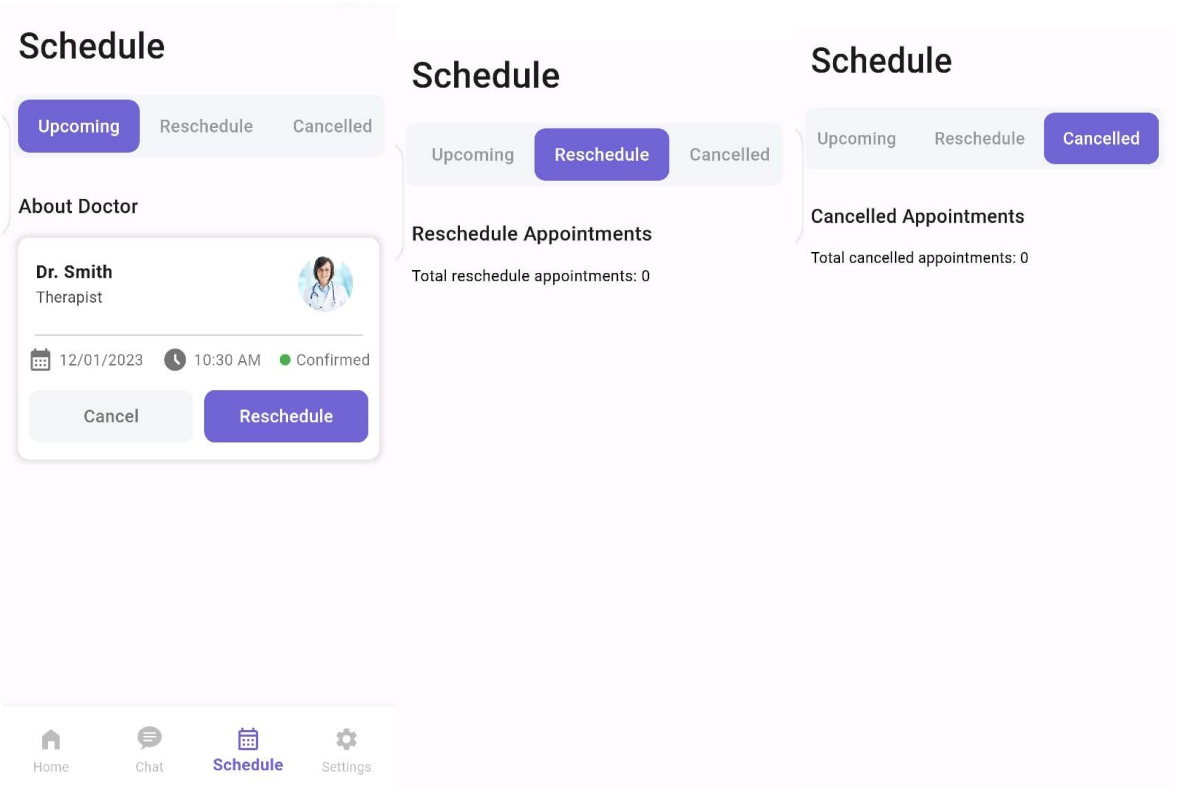

Fig 5.22: Schedule of Appointments

The Admin possesses the capability to view the appointment schedules of both doctors and patients within the platform. This comprehensive oversight enables the Admin to monitor and manage appointment bookings, ensuring efficient scheduling and allocation of resources. By having access to these schedules, the Admin can facilitate coordination between healthcare professionals and patients, optimize workflow management, and address any scheduling conflicts or issues that may arise.

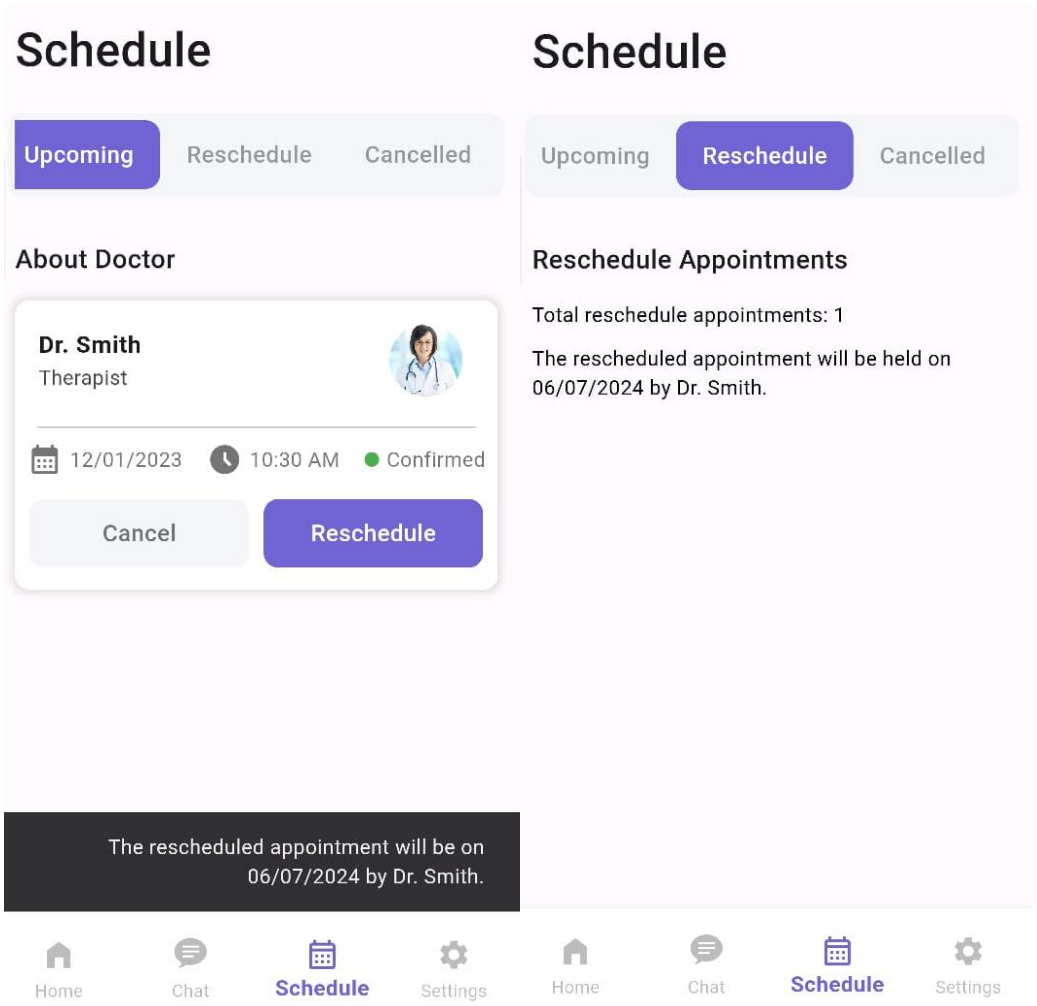

Fig 5.23: Reschedule an Appointment

The Admin possesses the authority to reschedule appointments as needed, allowing for flexibility in the scheduling system. Furthermore, the platform displays the total number of appointments, providing a comprehensive overview of the scheduling activity. This capability empowers the Admin to effectively manage appointment bookings, accommodate changes in scheduling, and ensure optimal utilization of resources.

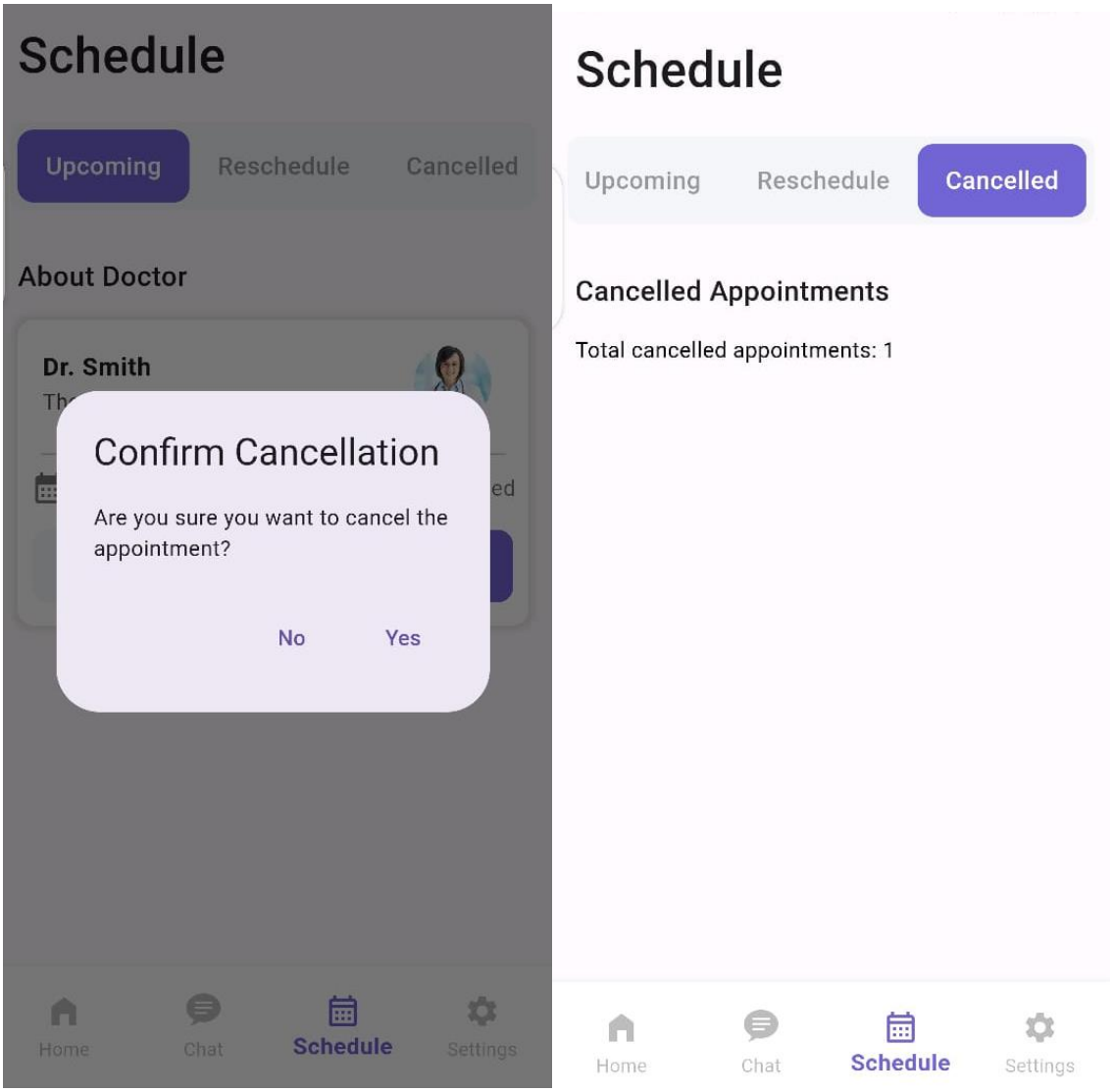

Fig 5.24: Cancel an Appointment

The Admin holds the authority to cancel appointments when necessary, facilitating efficient management of the scheduling system. Additionally, the platform displays the number of appointments that have been canceled, providing valuable insight into scheduling trends and potential areas for improvement. This capability empowers the Admin to optimize appointment scheduling processes, ensuring smooth operations and enhancing the overall experience for both doctors and patients.

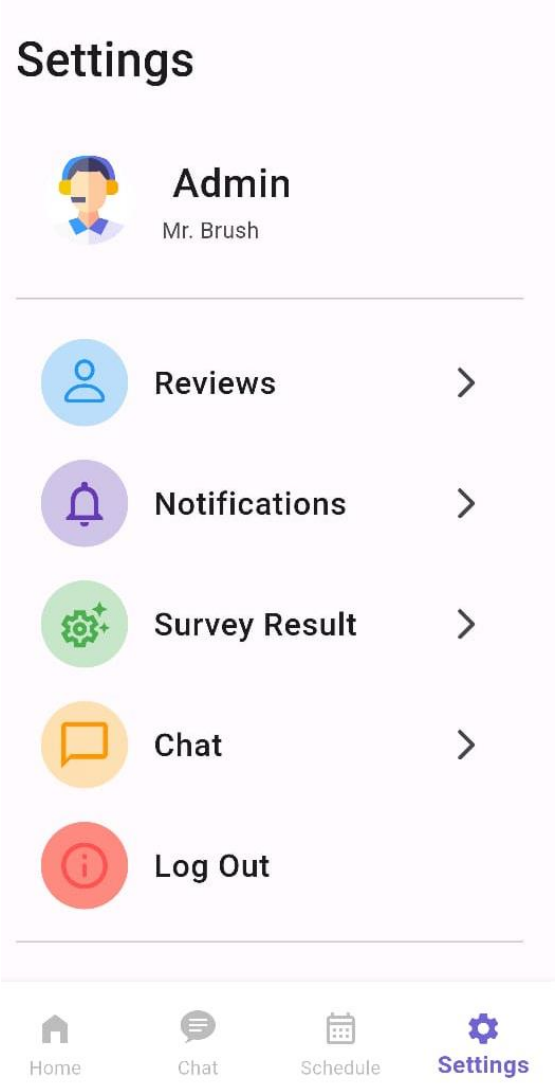

Fig 5.25: Settings

Settings within the Admin panel are accessible, allowing the Admin to customize various aspects of the platform to suit specific needs. This includes options to configure notification preferences, adjust security settings, and modify administrative permissions. By having access to these settings, the Admin can tailor the platform to optimize functionality and ensure alignment with organizational requirements, ultimately enhancing user experience and operational efficiency.

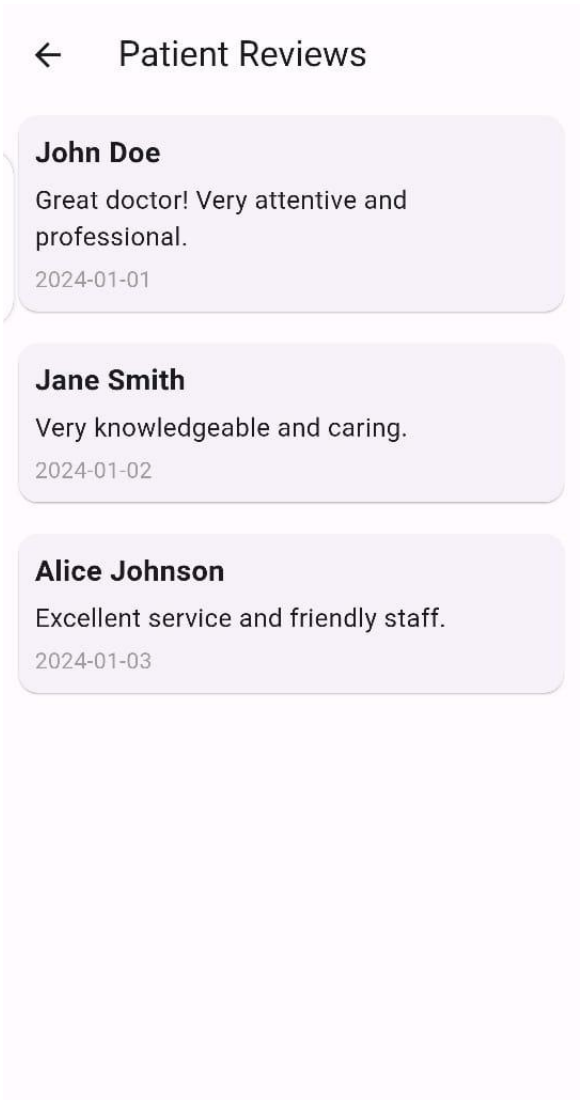

Fig 5.26: Patient Reviews

Patient reviews are shared with the Admin to provide insight into the quality of care and overall user experience within the platform. This allows the Admin to assess feedback, identify areas for improvement, and address any concerns raised by patients. By gaining visibility into patient reviews, the Admin can make informed decisions to enhance the platform's services, maintain user satisfaction, and continually improve the healthcare experience for all users.

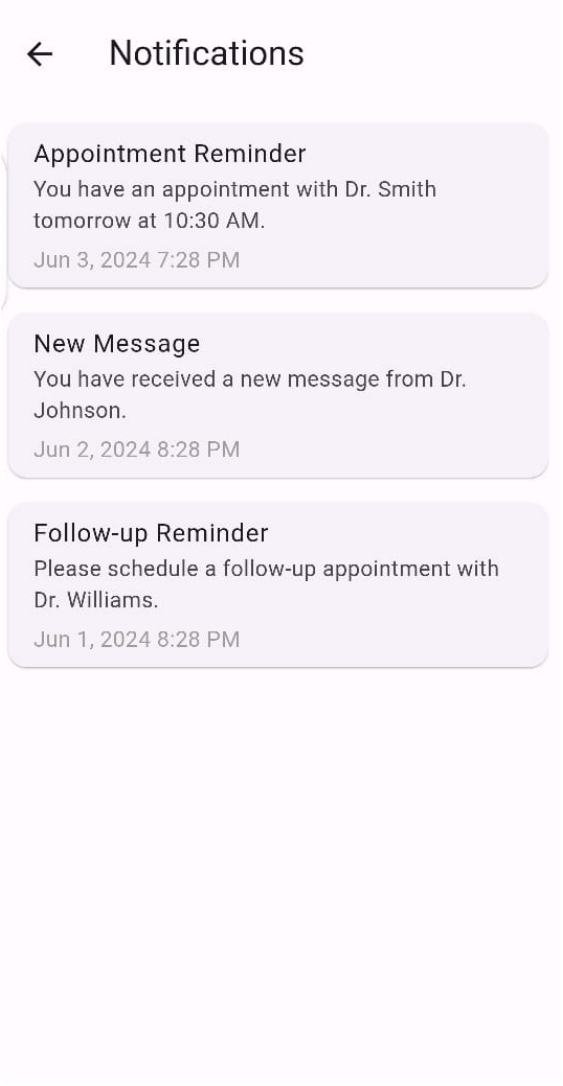

Fig 5.27: Notifications

The platform sends notifications to users, including appointments, updates on doctors, and follow-up reminders, ensuring timely communication and facilitating seamless healthcare management. These notifications help users stay informed about upcoming appointments, changes in doctor availability, and important follow-up tasks, enhancing overall efficiency and patient engagement. By receiving these notifications, users can effectively manage their healthcare schedules and stay connected with their healthcare providers, promoting better health outcomes and continuity of care.

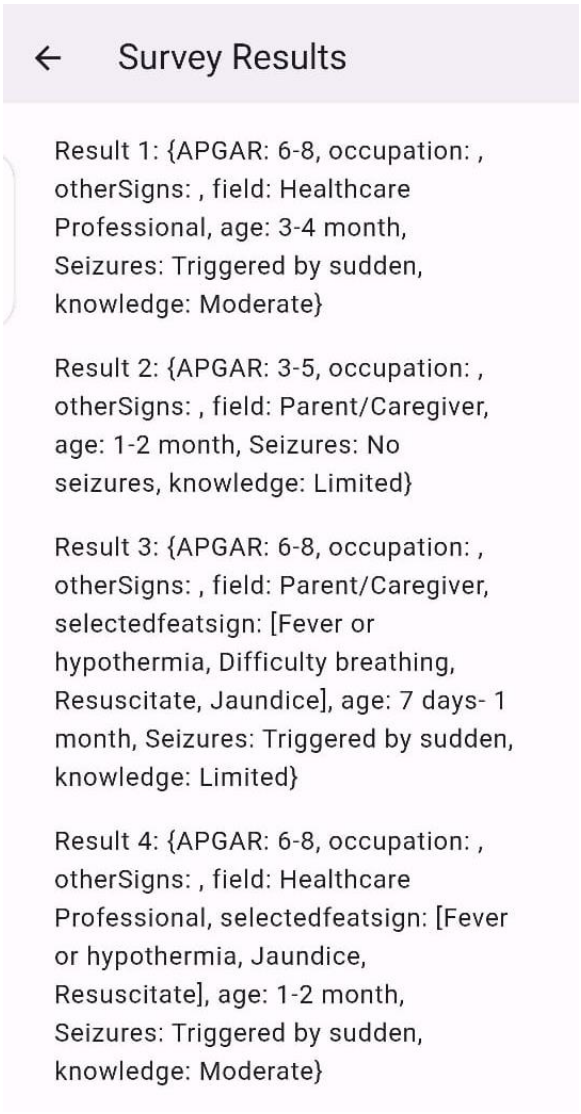

Fig 5.28: Survey Results

The Admin has access to view the predicted results of the surveys conducted within the platform. This enables them to gain insights into the health trends and potential conditions affecting users. By reviewing these predicted results, the Admin can better understand the healthcare needs of the community and make informed decisions to improve services and support. This visibility empowers the Admin to contribute to better health outcomes and overall well-being for users.

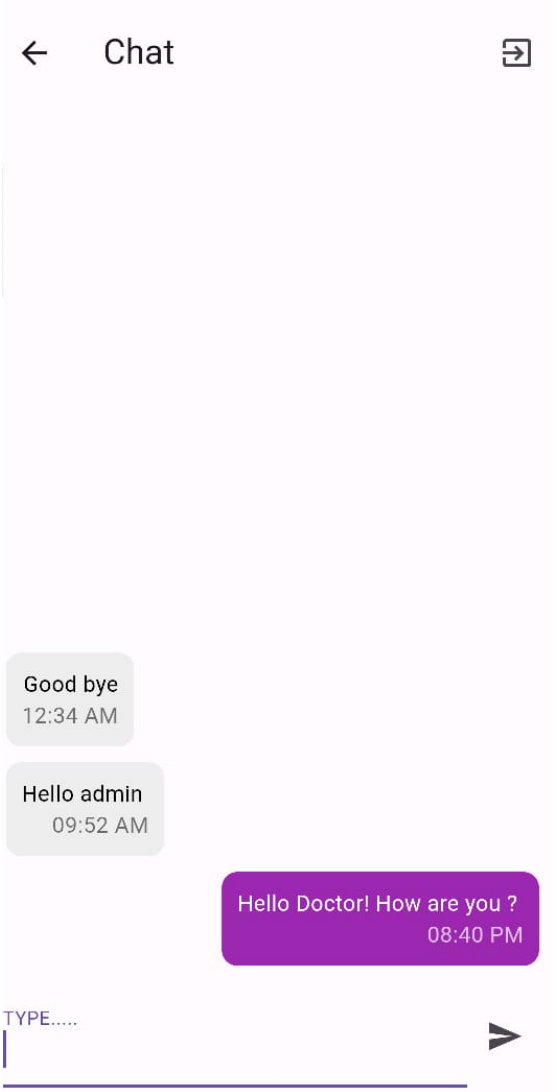

Fig 5.29: Admin Chat with Doctor

The Admin can engage in private, one-on-one conversations with doctors through the platform's chat feature. This allows the Admin to address specific inquiries, discuss sensitive matters, or provide personalized support directly to the doctor. By facilitating personal communication between the Admin and doctors, the platform enhances collaboration, streamlines administrative processes, and fosters a supportive environment for healthcare professionals.

Chapter 6: Testing and Evaluation

This chapter explores the thorough testing and evaluation procedures undertaken for the Newborn Disease Prediction System. From verification and functionality testing to deployment and maintenance, the chapter details the meticulous steps implemented to guarantee the reliability, usability, and efficacy of the system. By employing a blend of testing methodologies and evaluation techniques, the chapter showcases the methodical strategy employed to authenticate and enhance the Newborn Disease Prediction System, ultimately striving to provide a seamless and beneficial healthcare solution for users.

6.1. Verification

The newborn disease prediction system ensures thorough module completion through rigorous verification, meticulous unit testing, and compliance with specified requirements. Utilizing documentation from earlier stages, including requirements review and design/code evaluation, the system maintains integrity throughout the development lifecycle.

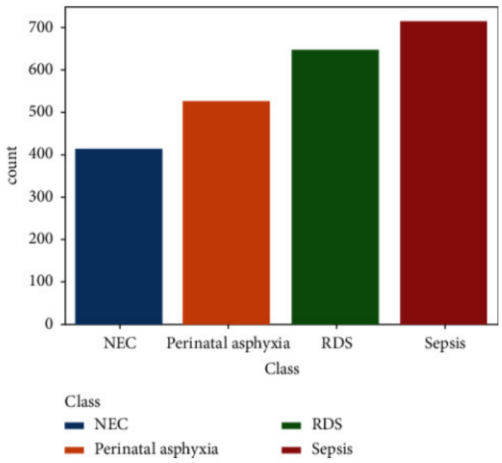

Fig 6.1: Disease Prediction

6.1.1. Functionality Testing:

All functionalities within the newborn disease prediction application, such as database connections and data submission, modification, retrieval, and deletion forms, are subjected to thorough testing. Developers conduct this testing, focusing on specific functionality requirements.

- **Security Testing:** Authentication, authorization, data encryption, input validation, session management, and error handling are rigorously examined to fortify protection against unauthorized access, data breaches, and security vulnerabilities in the newborn disease prediction system.
- **Database Testing:** The precision, consistency, and reliability of data stored in Firebase Realtime Database are meticulously evaluated within the newborn disease prediction system. Furthermore, the system's performance, compliance with security regulations, and the efficacy of backup and recovery protocols are scrutinized.

6.1.2. Static Testing

Static testing within the context of the newborn disease prediction system involves comprehensive review processes without executing the program. It encompasses the following key aspects:

- **Code Review:** Peer reviews are conducted to identify potential code defects, security vulnerabilities, and deviations from coding standards. Static code analysis tools are utilized to automate reviews, identifying code smells, potential bugs, and security vulnerabilities.
- **Documentation Review:** Rigorous examination of software documentation, including requirements specifications, design documents, and user manuals, ensures completeness, accuracy, and alignment with the implemented system. Any modifications during development are reflected in the documentation.
- **Standards and Conventions Compliance:** Verification of adherence to coding standards, naming conventions, and architectural guidelines established for the newborn disease prediction system. Compliance with relevant industry standards and regulatory requirements is also ensured.

6.2. Validation

Developers rigorously test the application's validations to ensure accuracy in newborn disease prediction. They verify that the system adheres to specified requirements, compiles successfully, and effectively performs its designated functions. Additionally, the application

is evaluated to ensure it meets specific criteria tailored to newborn health prediction, aligning with the system's intended purpose and objectives.

Table 6.1: Test Case for Sign-up with Valid Information

Test Case ID	**TC-001**
Test Case Name	**Choose a role – Parent** **Create Account - Email Format Validation**
Description	Verify that the email format correctly identifies valid email addresses.
Prerequisite	Connection with database established properly.
Steps	• The NeoCare App was opened. • The user clicked on "Sign In". • They inserted the email "hajrarehan@gmail.com" and the 6 digits numeric password "123456". • Afterwards, they clicked on "SIGN UP".
Expected Result	User will be successfully Registered.
Results	User is successfully Registered.
Pass/Fail	Pass

Table 6.2: Test Case for Sign-up with Invalid Information

Test Case ID	**TC-002**
Test Case Name	**Create Account - Invalid Email Format Validation**
Description	Verify that the email format correctly identifies invalid email addresses.
Prerequisite	Connection with database established properly.
Steps	• The NeoCare App was opened. • The user clicked on "Sign Up". • They inserted the email "hajragmail.com" and the password "1234". • Afterwards, they clicked on "SIGN UP".
Expected Result	User will not be able to Register.
Results	User is not Registered.
Pass/Fail	Pass

Table 6.3: Test Case for Log-in

Test Case ID	**TC-003**
Test Case Name	**Login Account**
Description	Users must log into their account so they can see services provided by the application
Prerequisite	• Connection with database established properly. • Account is created by user.
Steps	• Application will login after inserting correct email and password. • Successfully login on correct information. • Easy access of account to user after successful login.
Expected Result	Successful login
Results	• Successfully login on inserting correct information. • Failed login issue on invalid information.
Estimated Time	• Successfully login within 15 seconds. • Also depends on internet speed.
Error	• Wrong email or password. • Account not created. • Internet connection error.

Table 6.4: Test Case for Prediction Result

Test Case ID	**TC-004**
Test Case Name	**View Prediction Result**
Description	Users must register to fill a survey form provided by the application.
Prerequisite	• Database connection established properly. • User account created.
Steps	Survey Form related to disease symptoms. User fill the survey form. The prediction result shows to user.
Expected Result	Both admin and user can access the prediction result.
Results	• Both admin and users can access the prediction result successfully.

	• The prediction result appears within 15 seconds, depending on the internet speed. • Errors may occur if the fields of survey form not completely filled. • Incorrect fields are filled.
Estimated Time	• Prediction result appears within 15 seconds, subject to internet speed.
Error	• Users unable to view prediction due to slow internet connection or lack of internet connectivity. • Errors may occur if the fields of survey form not completely filled. • Incorrect fields are filled. Incorrect Results.

Table 6.5: Test Case for Provide Feedback

Test Case ID	**TC-005**
Test Case Name	**Provide Feedback**
Description	Users can submit feedback about the prediction results, and they are able to view their feedback data for each prediction. After the prediction, the doctor recommendation screen is displayed.
Prerequisite	User must be logged in and have performed a prediction activity.
Steps	Admin views feedback data submitted by users. User views and provides feedback data for the prediction. Doctor recommendation screen is displayed after prediction.
Expected Result	To monitor and improve the prediction accuracy, and provide users with relevant doctor recommendations.
Results	Users successfully submit feedback for each prediction. Admin can view the feedback data provided by users.
Estimated Time	• Feedback submission within 15 seconds, subject to internet speed.
Error	• Admin fails to view feedback data due to a slow internet connection. • User fails to provide feedback due to a slow internet connection. • Internet connectivity issues.

Table 6.6: Test Case for Book an Appointment

Test Case ID	**TC-006**
Test Case Name	**Book an Appointment**
Description	Users book an appointment with a doctor and view their appointment details.
Prerequisite	User must be logged in and have selected a doctor.
Steps	User selects a doctor. User chooses a suitable date and time for the appointment. User confirms the appointment booking.
Expected Result	Appointment details are stored in the database, and users can view their appointment information.
Results	Users successfully book appointments with doctors. Appointment details are accurately stored in the database.
Estimated Time	• Appointment booking completed within 30 seconds. • Time may vary depending on internet speed.
Error	• Appointment booking fails due to a slow internet connection. • Appointment details are not stored in the database due to connectivity issues. • Users encounter errors when trying to confirm the appointment.

Table 6.7: Test Case for Admin Authority - Add and Delete

Test Case ID	**TC-007**
Test Case Name	**Admin Authority-Add and Delete Doctor**
Description	Admin has the authority to add new entities and delete existing ones.
Prerequisite	Admin must be logged in and have the necessary permissions.
Steps	Admin accesses the management section. Admin selects the option to add a new entity. Admin fills in the required details and submits to add. Admin selects the option to delete an existing entity. Admin chooses the entity to delete and confirms the action.

Expected Result	Admin successfully adds a new entity and deletes an existing one.
Results	New entity added successfully. Existing entity deleted successfully
Estimated Time	• Adding a new entity: 1 minute. • Deleting an existing entity: 30 seconds. (Times may vary based on server response and internet speed.)
Error	• Admin encounters an error while adding due to incomplete information or server issues. • Deletion fails due to connectivity issues or unauthorized access.

6.3. Usability Testing

In the usability testing phase for the newborn disease prediction system, specific criteria and objectives were established to align with user expectations and application requirements. Representative users were enlisted to partake in testing sessions, during which they engaged with the system to assess its intuitiveness, navigation flow, and overall user experience. Feedback gathered from these interactions was carefully analyzed to pinpoint areas for enhancement in the design and interface of the application, ensuring its effectiveness in aiding newborn disease prediction.

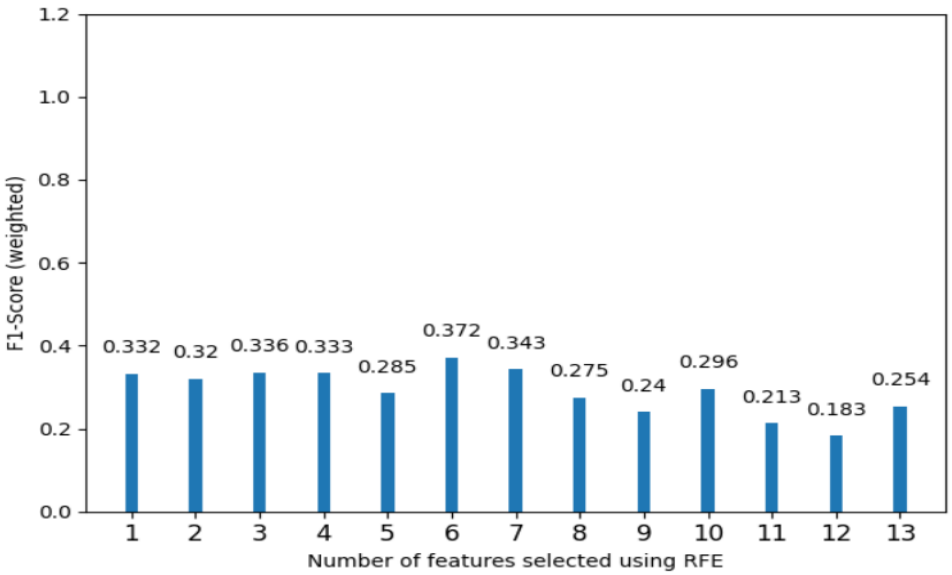

Fig 6.2: Usability Testing of Disease Prediction

6.4. Module / Unit Testing

Module /unit testing was performed to assess the functionality and behavior of individual components or modules in isolation. Test cases were designed and executed to verify the accuracy and effectiveness of each module independently. Following the identification and resolution of any defects uncovered during unit testing, the integration of these thoroughly tested modules into larger system components was undertaken.

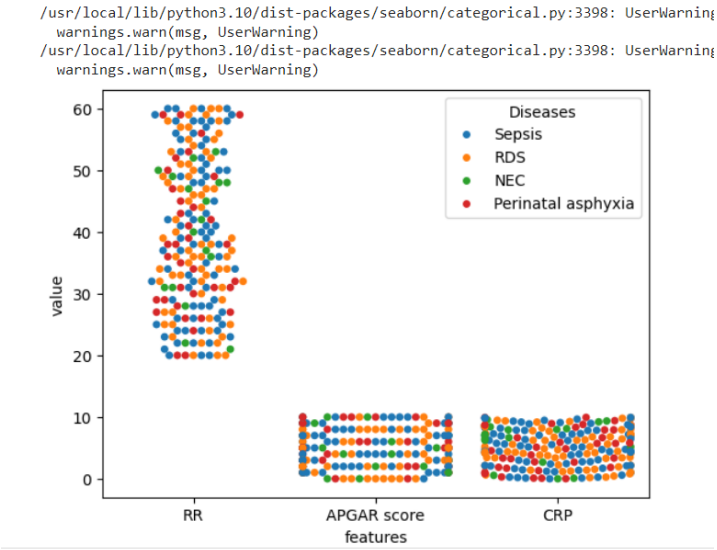

Fig 6.3: Unit Testing of Disease Prediction

6.5. Integration Testing

Integration testing for the newborn disease prediction system focused on identifying integration points between various modules and components. Test cases were designed to validate the interaction and data flow between these integrated components, ensuring smooth communication and compatibility throughout the system. Any issues or discrepancies discovered during testing were addressed to guarantee the seamless operation of the entire newborn disease prediction system.

6.6. System Testing

System testing for the newborn disease prediction system involved assessing the application's entirety, comprising integrated components and modules. Test cases were meticulously

designed to cover functional and non-functional requirements, and tests were executed to verify the system's behavior, performance, and reliability across diverse scenarios. Any identified defects were systematically documented, reported, and rectified to uphold the application's quality standards.

6.7. Acceptance Testing

Acceptance testing for the newborn disease prediction system involved validating the application against predefined acceptance criteria to ensure it met the needs and expectations of stakeholders. Test scenarios were tailored based on stakeholder requirements and project objectives, encompassing tasks such as user registration, survey form submission, doctor recommendation, appointment booking, and viewing appointment details. Stakeholders or representative users participated in acceptance testing to provide feedback, which was then used to address any issues or concerns and obtain stakeholder approval for deployment.

6.8. Stress Testing

Stress testing for the newborn disease prediction system evaluated the application's performance under extreme conditions, including high volumes of user activity and resource constraints. Test scenarios were created to assess the system's responsiveness, scalability, and stability when faced with stress conditions. Performance bottlenecks were identified, and optimizations were implemented to enhance the application's reliability under stress.

6.9. Hardware Configuration for Testing

Hardware configuration testing for the newborn disease prediction system involved validating the application's performance across a range of hardware configurations. Test environments were set up with different devices, operating systems, and browsers to ensure compatibility and optimal functionality. Test cases were executed to assess consistent performance across diverse hardware setups, addressing any compatibility issues to guarantee a seamless user experience across various platforms.

6.10. Evaluation

Evaluation activities for the newborn disease prediction system comprised reviewing testing artifacts, analyzing test results, and documenting findings and recommendations for future enhancements. Post-mortem meetings were organized to review testing outcomes and lessons learned. Feedback from these evaluation activities was carefully considered and incorporated into the project's improvement process to refine future testing and development cycles, ensuring ongoing optimization of the system's performance and user experience.

6.11. Deployment

Deployment activities for the newborn disease prediction system encompassed various tasks, including preparing the application for deployment, scheduling deployment timelines, and executing deployment processes to production environments. Following deployment, smoke tests were conducted to validate the system's functionality, and any deployment-related issues were promptly resolved. Continuous monitoring throughout the deployment process ensured a seamless transition to the live environment.

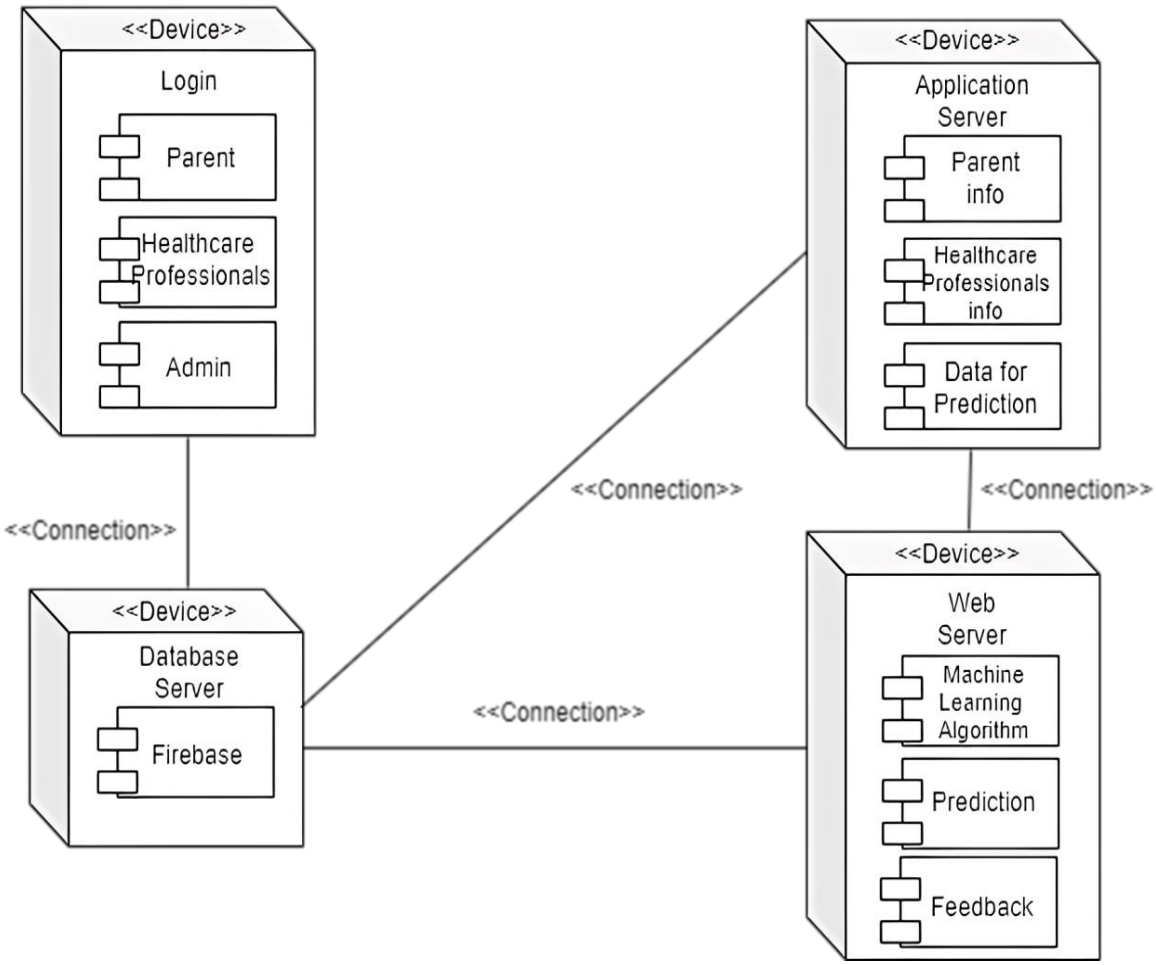

Fig 6.4: Deployment Diagram

6.12. Maintenance

Maintenance activities for the newborn disease prediction system involved providing ongoing support and updates to the application post-deployment. This included addressing bugs, implementing security patches, and enhancing features as needed. Regression testing was carried out to verify the stability of the system after each update. Documentation of maintenance activities was maintained for future reference and auditing, ensuring the continued reliability and usability of the system.

Chapter 7: Conclusion and Future Work

7.1. Conclusion

The NeoCare app is committed to transforming newborn healthcare by offering a tailored, inclusive, and interactive platform for parents. It strives to empower parents to confidently monitor their newborn's health while nurturing a proactive approach to early disease detection and prevention. NeoCare aims to establish a harmonious synergy between technology, parents, and healthcare providers, ultimately improving the standard of newborn care and making a positive difference in families' lives.

7.2. Future Work

For future enhancements, the NeoCare app could target several key areas to maximize its impact and effectiveness in newborn disease prediction and care:

1. **Continuous Refinement:** Regular updates and refinements based on user feedback and advancements in medical technology can ensure the app remains up-to-date and efficient in predicting and preventing newborn diseases.
2. **Expansion of Medical Content:** Increasing the depth and breadth of medical content available on the app can cater to a wider audience and offer more comprehensive insights into various newborn health conditions.
3. **Personalization Features:** Enhancing the app's ability to tailor disease prediction and care recommendations based on individual newborn profiles and medical histories can improve its effectiveness in providing personalized healthcare solutions.
4. **Parental Engagement Tools:** Introducing features that encourage and facilitate parental involvement in newborn care, such as health tracking, communication with healthcare providers, and access to educational resources, can empower parents in managing their newborn's health.
5. **Collaborations and Partnerships:** Collaborating with healthcare institutions, medical experts, and research organizations can enrich the app's medical content and ensure users have access to the latest advancements in newborn healthcare.

6. **Accessibility Measures:** Ensuring that the app is accessible to parents from diverse backgrounds and with varying levels of medical knowledge, including providing multilingual support and user-friendly interfaces.
7. **Research and Evaluation Initiatives:** Conducting research studies to evaluate the app's impact on newborn health outcomes and user experiences can provide valuable insights for further development and refinement.

By focusing on these areas, the NeoCare app can continue to evolve as a trusted resource for newborn disease prediction and care, empowering parents and healthcare providers in safeguarding the health and well-being of newborns.

References

1. Shirwaikar R. D., Mago N., U D. A., Makkithaya K. Supervised learning techniques for analysis of neonatal data. Proceedings of the 2nd International Conference on Applied and Theoretical Computing and Communication Technology; July 2016; Bangalore, India. iCATccT); pp. 25–31. [Google Scholar]
2. Sheikhtaheri A., Zarkesh R., Moradi R., Kermani F. Prediction of neonatal deaths in NICUs: development and validation of machine learning models. BMC Medical Informatics and Decision Making. 2021;21(1):p. 131. [PMC free article] [PubMed] [Google Scholar] Chowdhury D. R., Chatterjee M., Samanta R. K. An artificial neural network model for neonatal disease diagnosis. International Journal of Artificial Intelligence and Expert Systems. 2011;2(3):96–106. [Google Scholar]
3. Safdari R., Kadivar M., Langarizadeh M., Nejad A. F., Kermani F. Developing a fuzzy expert system to predict the risk of neonatal death. Acta Informatica Medica. 2016;24(1):p. 34. [PMC free article] [PubMed] [Google Scholar]
4. Shirwaikar R. D., Acharya D. U., Makkithaya K., Surulivelrajan M., Lewis L. E. S. Machine learning techniques for neonatal apnea prediction. Journal of Artificial Intelligence. 2016; 9:33–38. doi: 10.3923/jai.2016.33.38. [CrossRef] [Google Scholar]
5. Bitew F. H., Nyarko S. H., Potter L., Sparks C. S. Machine learning approach for predicting under-five mortality determinants in Ethiopia: evidence from the 2016 Ethiopian Demographic and Health Survey. Genus. 2020;76(1):1–16. [Google Scholar]
6. Sun P., Wang D., Mok V. C., Shi L. Comparison of feature selection methods and machine learning classifiers for radiomics analysis in glioma grading. IEEE Access. 2019; 7:102010–102020. doi: 10.1109/access.2019.2928975. [CrossRef] [Google Scholar]