Abstract

Child vaccination is a key factor of ensuring health of children from birth to twelve years old. However, the current manual process is not aligned with today's world and its needs. Parents and guardians of the children face significant challenges due to this, including time consuming manual scheduling and inaccessible vaccination histories. These shortcomings may have negative impact on the users as well as the whole vaccination process. As a solution for this, 7Guard is introduced as an effective digital platform that enables easy and accurate services for the parents and guardians of the children, ensuring their children's well-being with standard health services.

Acknowledgement

We would like to express our gratitude to everyone who contributed to the successful development of this project.

First, we extend our sincere thanks to our supervisor, Ms Nilusha Chamindi Perera, for her invaluable guidance, encouragement, and feedback that directed us to the successful result of the project.

We are deeply thankful to the healthcare professionals and staff of Santa Dora Hospital who shared their knowledge and experience to identify the requirements and shape the solution into a final product that meets the needs of the users.

Finally, a heartfelt thank you goes to all the team members for their unwavering hard work, support, and spirit during this project.

We are profoundly grateful for all the support and contribution we received to make this project a success.

Declaration

We declare that this project report or part of it was not a copy of a document done by any organization, university any other institute or a previous student project group at KIU Campus and was not copied from the Internet or other sources.

Project Details

Project Title	7Guard
Project ID	09

Group Members

Reg. No	Name	Signature
11162	G. M. A. S. Aponsu	
11037	K. G. G. N. D. Weerathunga	
11055	K. K. N. T. Madhudhani	
14633	K. D. M. R. Amada	
11179	K. M. P. U. Bandara	
14634	A. L. L. Wijesiri	
11497	D. R. Ravindya	
11323	M. H. K. Bandara	
11304	S. P. D. Madusanka	
11254	R. G. C. D. Rajamanthri	

Table of Content

List of Figures

List of Acronyms and Abbreviations

Acronyms and Abbreviations	Description
CHDR	Child Health Development Record
BMI	Body Mass Index
OTP	One Time Password
SQL	Structured Query Language

1. Introduction

1.1. Problem Statement

Child vaccination is one of the most critical aspects of ensuring health and well-being of children from age of newborn to twelve. Yet, the process of managing vaccination appointments and records is outdated and hard work. Parents, or the guardians of the children often face challenges, such as, time consuming manual scheduling, lack of accessible vaccination histories, and not having accurate records of children's health. These hardships lead to an inconvenient and risky health care process.

"7Guard" is introduced as an innovative solution to these issues. By offering a well-designed, user-friendly digital platform, the system addresses the key points of vaccine registration and management. It gives the opportunity to the parents or the guardians to easily book vaccination appointments online, get real-time updates of vaccination details, get medical advises for side effects of vaccination if they might occur, get health tips based on the BMI value, and have an online accessible CHDR card as well. 7Guard ensures that including parents, guardians, children, healthcare providers, and all the stakeholders are served with quality service, enhancing greater efficiency, accuracy and awareness in child healthcare.

1.2. Product Scope

7Guard is a digital platform that modernizes the child vaccination process of children between birth and twelve years old. It acts as a connector between the parents or guardians of the children and the healthcare providers by offering efficient and user-friendly services related to vaccination such as booking vaccine appointments, supplying advises for the children, recording vaccine histories and overall child healthcare.

The main objective of 7Guard is to streamline the child vaccination process by providing a user-friendly interface that provides easy vaccination appointments, real-time updates and reminders, and an online accessible CHDR card. 7Guard also maintains a secure database that stores all the

vaccination records of the children. It also aims to promote good healthcare within the system by sending health advises to the parents and guardians.

7Guard is a convenient online platform that allows parents and guardians to make vaccine appointments and access the health records through a digital interface. The system stores clear and accurate health records and ensure the security of the data. It also reduces errors that can happen by the user's side by providing reminders through emails. It facilitates the users by providing health tips such as advices based on the BMI value and side effects. The users can also submit feedback and let the administration know about their experience. 7Guard is a beneficial opportunity for anyone who values their children's health and encourages proper vaccination process.

The introduced software aligns with corporate goals by highlighting the need of innovation in the healthcare management and child immunization. It aims to enhance customer satisfaction by offering a user-friendly platform and promoting preventive health care of children, to create a healthier community.

The current process of child vaccination appointments and storing those data are done manually, which are prone to errors, delays, and misplacement of data. The parents or guardians of children faces challenges due to these such as difficulty in making appointments specified for the process, lack of reminders, limited awareness of their child's health, and not having a CHDR card that can be accessed at any time. These demands a timely and accurate solution. 7Guard addresses these issues by providing quality features.

1.3. Project Report Structure

The report structure follows several chapters focused on the aspects of the project to offer a clear explanation of 7Guard.

- **Methodology:** Outlines the systematic approach used to develop the project. It includes four subsections that elaborates on the methodology.

    - Requirements and Analysis: Explains the functional and non-functional requirements of the system. It specifies stakeholder needs with use case and activity diagrams to understand the system's workflow.
    - Design: Explains the architecture and structural aspects of 7Guard with activity diagrams. The database design is explained through an ER diagram.
    - Implementation: Explains the major modules and their structures. This includes the framework, development tools, libraries used to create the system, choice of database management system, programming languages used, and unique algorithms used.
    - Testing: Explains the testing plan of the project.

- **Evaluation:** Outlines the results of the project, lessons learned, and future developments of the system.
    - Assessment of the Project Results: Explains the tools used to analyze the system performance, the errors and limitations faced while testing the project, and areas to improve.
    - Lessons Learned: Explains the knowledge gained throughout the project, such as technical lessons, management lessons, and collaborative lessons.
    - Future Work: Explains the suggestions for future development such as, enhancing features and advanced analytics for vaccination process.

- **Conclusion:** Outlines a summary of the project with key factors.

2. Methodology

2.1. Requirements and Analysis

2.1.1. Functional Requirements:

- User Registration: Parent or the guardian details are registered with their child details. A profile is created for the child and the parent.

- Assigning Doctors for Appointments: A doctor assigns themselves for a child according to their available date and time.
- Vaccine Appointments: Allows parents or guardians to book vaccination appointments online. Allows receptionist to manage and confirm the appointments.
- Displaying Vaccine List, Stock and Importance: Vaccine list and importance of those vaccines are displayed. Vaccine stock is stored and interact with the pharmacist database of the hospital (Assumed).
- Vaccine Scan: Scans the vaccine codes and store records of it as vaccination history.
- Calculating BMI: BMI value of children is calculated and displayed as graph in the CHDR card.
- Health Advises Based on BMI: Provides health tips and advises through emails based on children's BMI value and possible side effects that might occur after vaccination.
- Health Advises Based on Vaccination Side Effects: If any side effects might occur after vaccinating a child, medical advises related to those are sent to the parent/guardian through their email.
- Vaccination Appointment Reminders: Reminders are sent to the parent/guardian for the upcoming appointments.
- Feedback: Allows users to submit feedback about the services they receive.

2.1.2. Non-Functional Requirements:

- Performance: System can handle a large number of concurrent users without having an issue.
- Usability: Provides a user-friendly interface for parents, guardians, and healthcare providers.
- Reliability: The system is up to time with real-time updates to avoid errors and disruption in healthcare services.
- Security: Authorizes the log in by providing an OTP for the users. Hospital staff is given a security password by the Administration for log in, ensuring security.

2.1.3. Use Case Diagram Explanation

- Parent/Guardian:

  Parent or guardian of the children can book vaccination appointments. To have a confirmed booking, they receive an email from the hospital first. They can also update the appointment under valid circumstances within a limited time period and delete an appointment if needed.

  Parents/Guardians can view the CHDR card through their child's profile. Along with that they can also view the vaccine list, importance of vaccines, health advises based on the BMI value, medical advises about side effect of vaccinations, feedback they make, and reminders sent to them. They can edit and delete the feedback if needed.

- Nurse:

  Nurse adds parent/guardian and child details and register the users, view those details when needed, update the details to maintain true data. Nurse adds the scanned barcode of vaccines to store those details in the database. The BMI calculation is also conducted by the nurse. They can also add the vaccine list and importance of the vaccines, view, update, and delete those details if needed. Nurse can view the CHDR card when needed as well. Managing reminders such as sending OTP, sending reminders, viewing them, and delete a reminder if needed is also done by the nurse.

- Doctor:

  Doctor can add prescription, advises based on BMI, advises based on the possible side effects that might occur after vaccination. Doctor can view CHDR card, the prescriptions and advises they sent. Prescription updating has a limited time. The advises can be updated and deleted if needed.

- Pharmacist:

  Pharmacist can add vaccine stock details into the database and view those details if needed.

- Receptionist:

  Receptionist can add doctor appointments; allocating doctors to the user appointments. They can also view them. Confirming user vaccination appointments, viewing them, updating, and deleting is done by the receptionist as well.

- Administration:

  Admins add staff details to the system. They can also view and update those details if needed.

Figure 1: Use Case Diagram for the Whole System

Figure 2: Activity Diagram

2.2. Design

2.2.1. Sequences Diagrams for the main functionalities

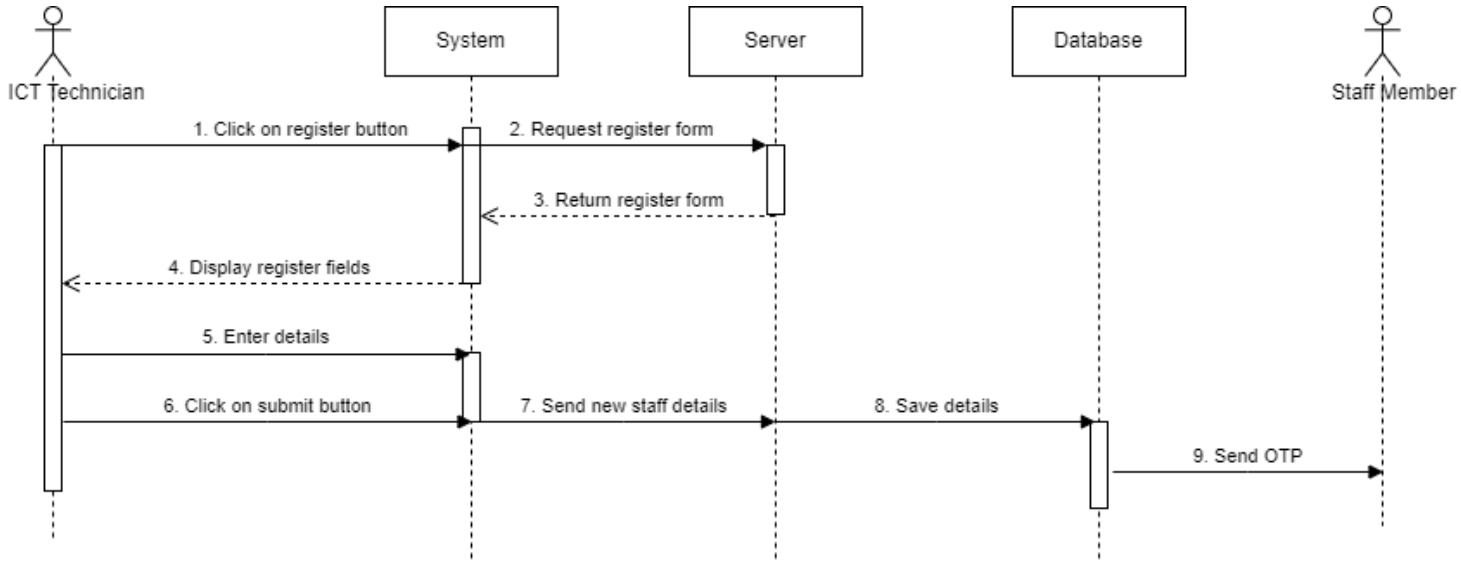

Figure 3: Sequence Diagram for Staff Registering

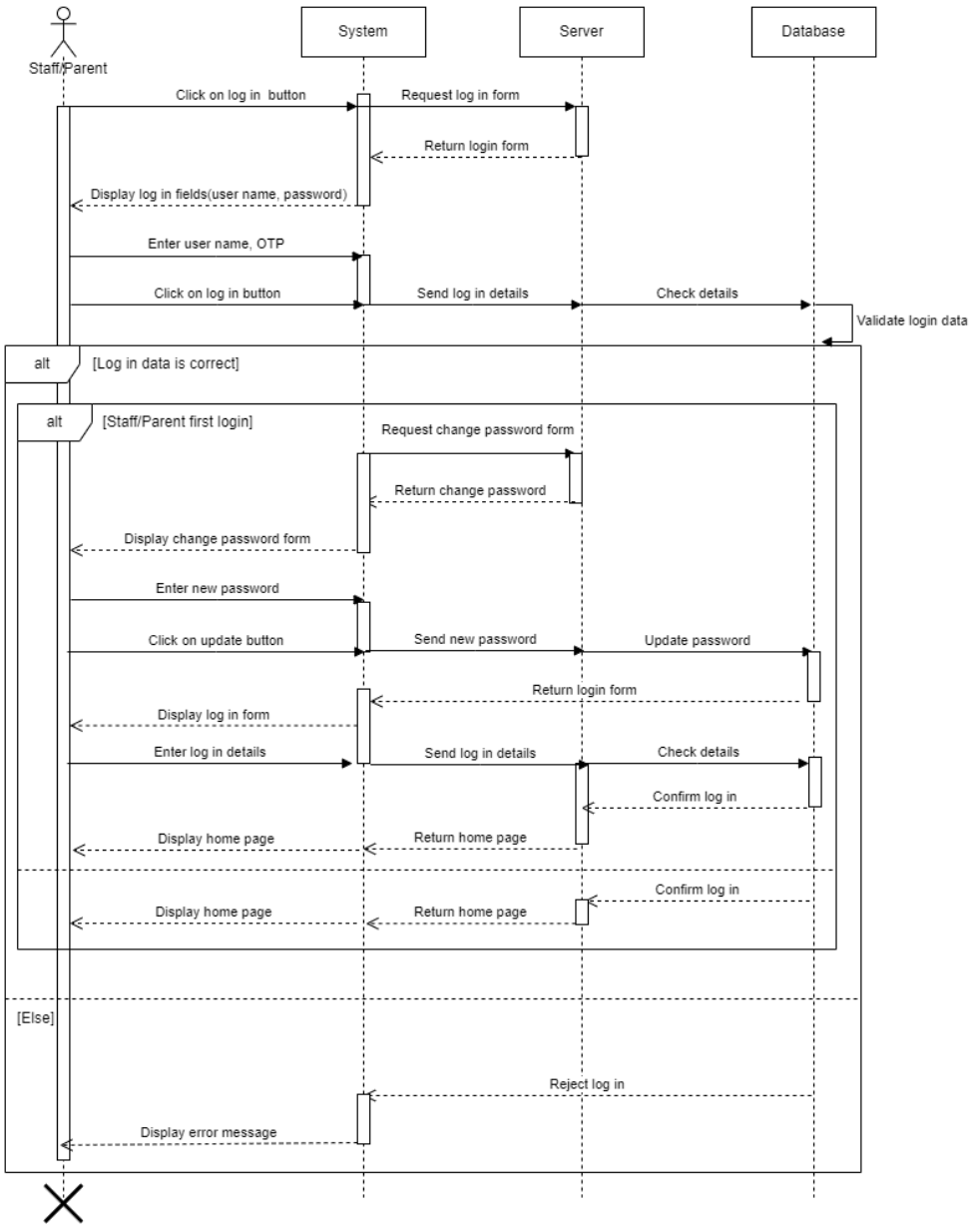

Figure 4: Sequence Diagram for Staff Log In

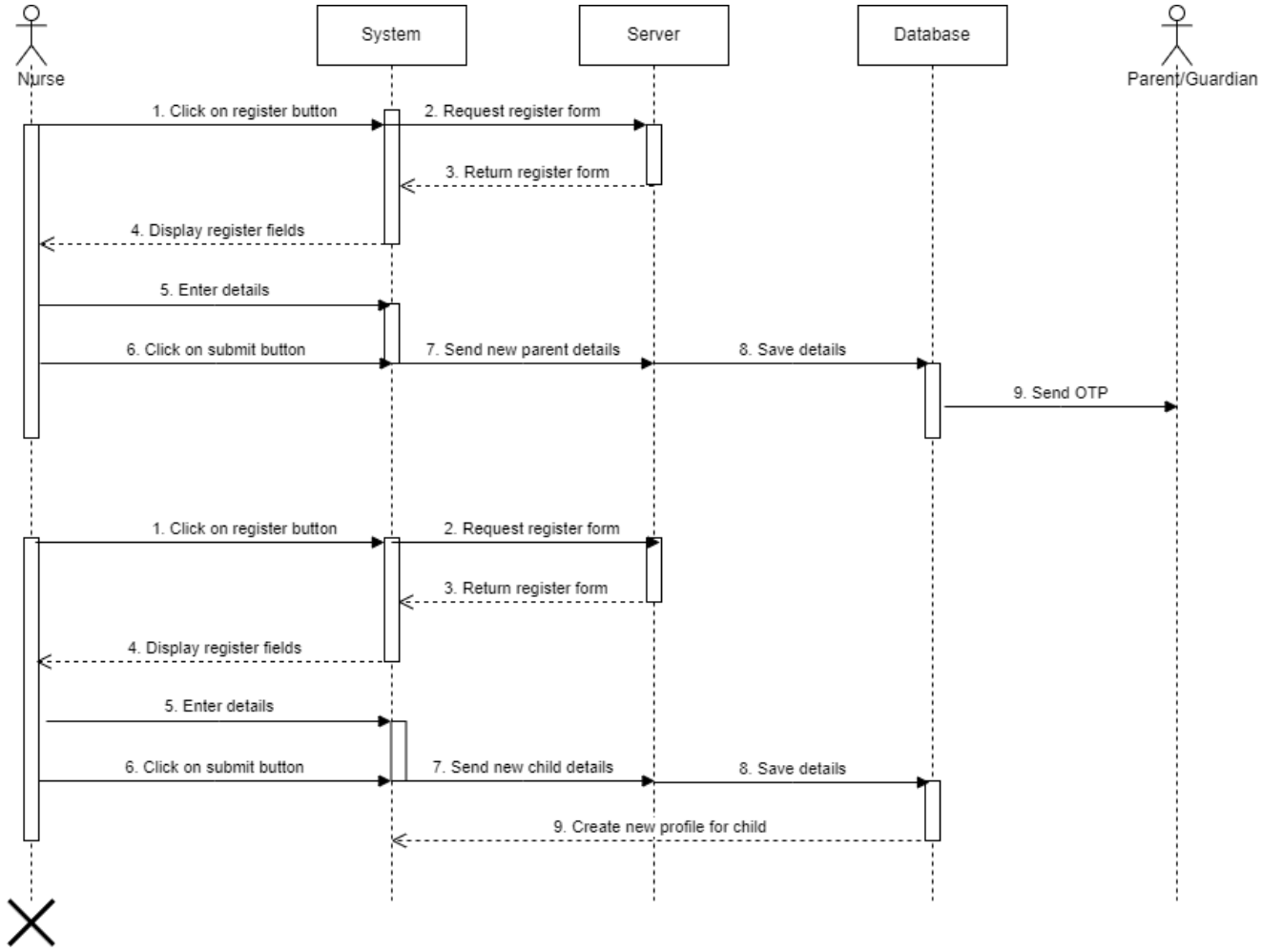

Figure 5: Sequence Diagram for Parent and Child Registration

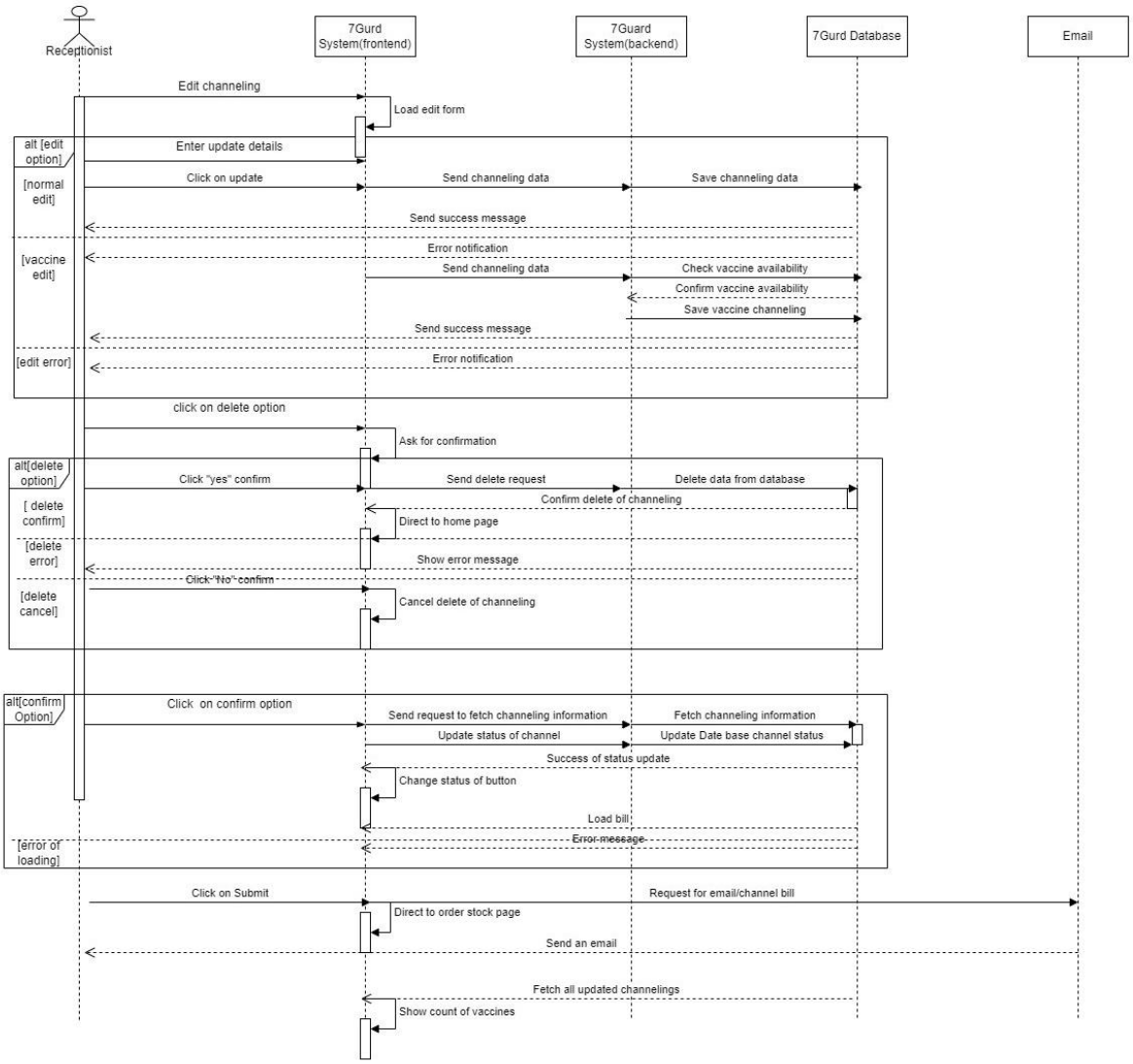

Figure 6: Sequence Diagram for Channeling (Admin Side)

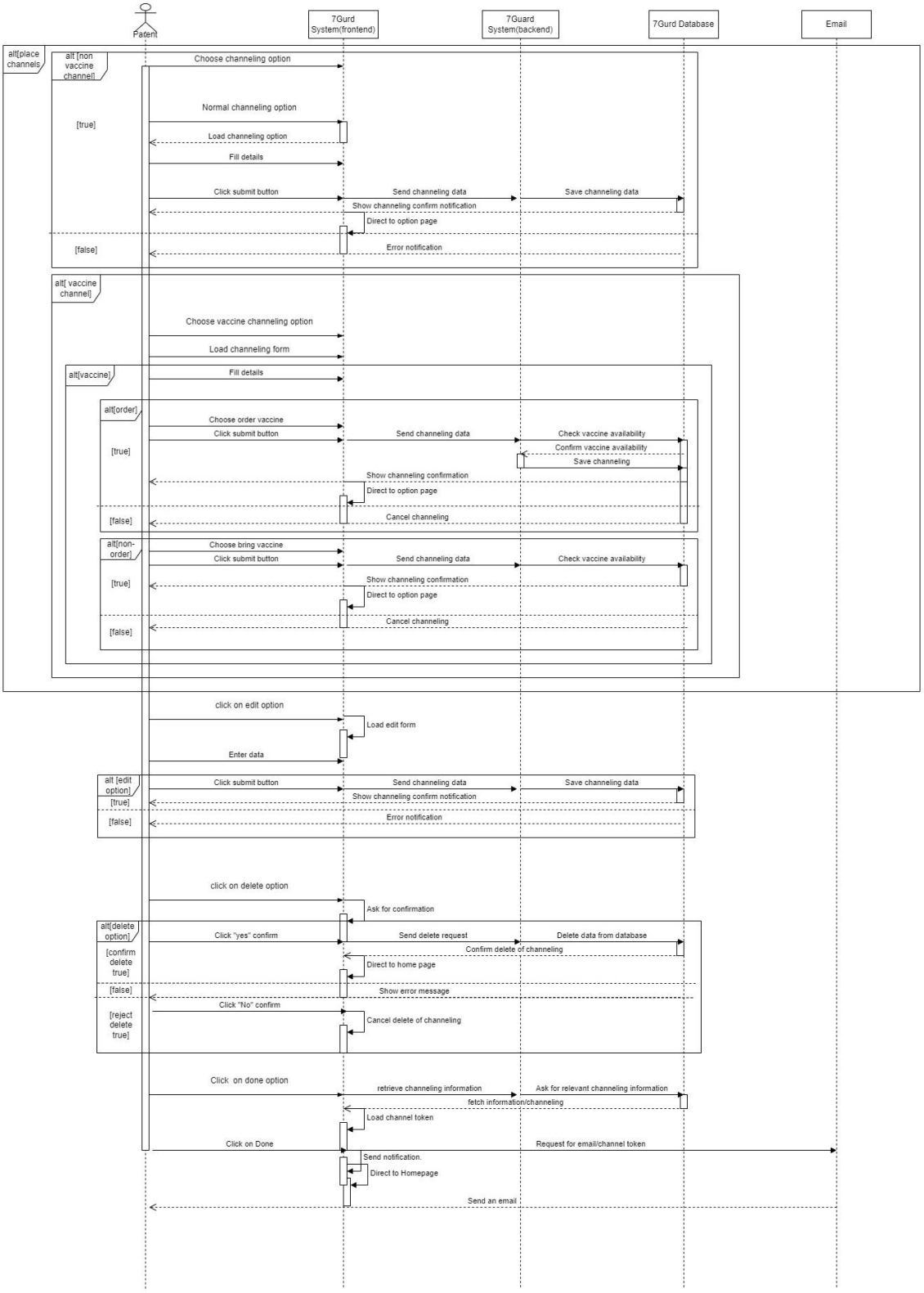

Figure 7: Sequence Diagram for Channeling (Parent Side)

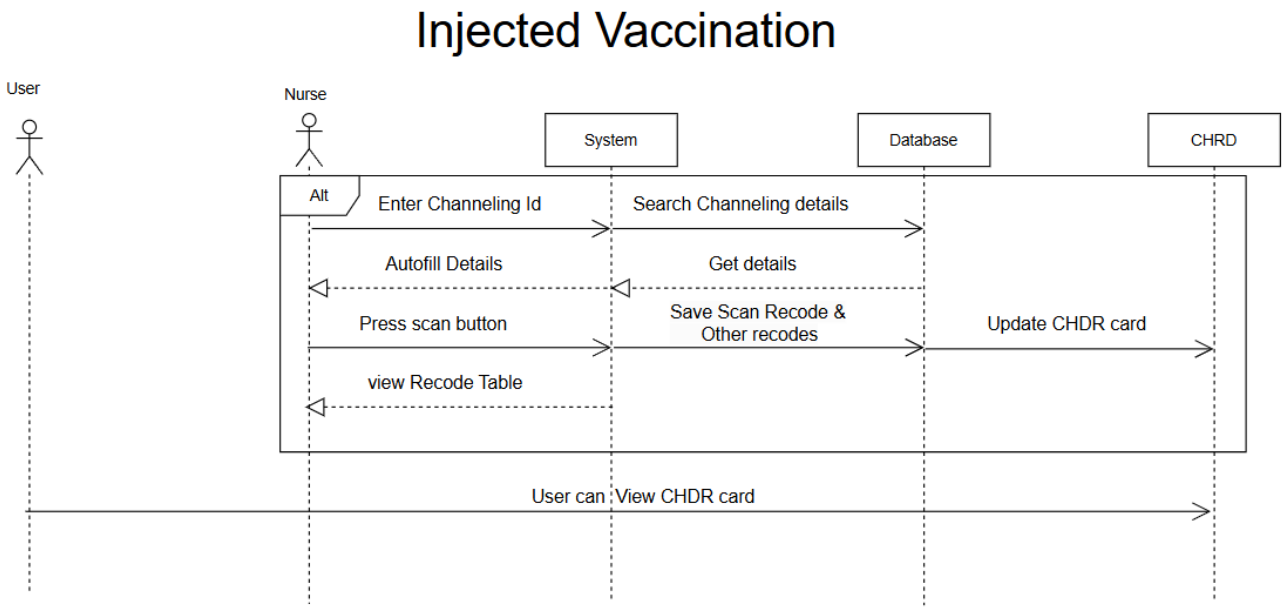

Figure 8: Sequence Diagram for Scanning Injected Vaccine

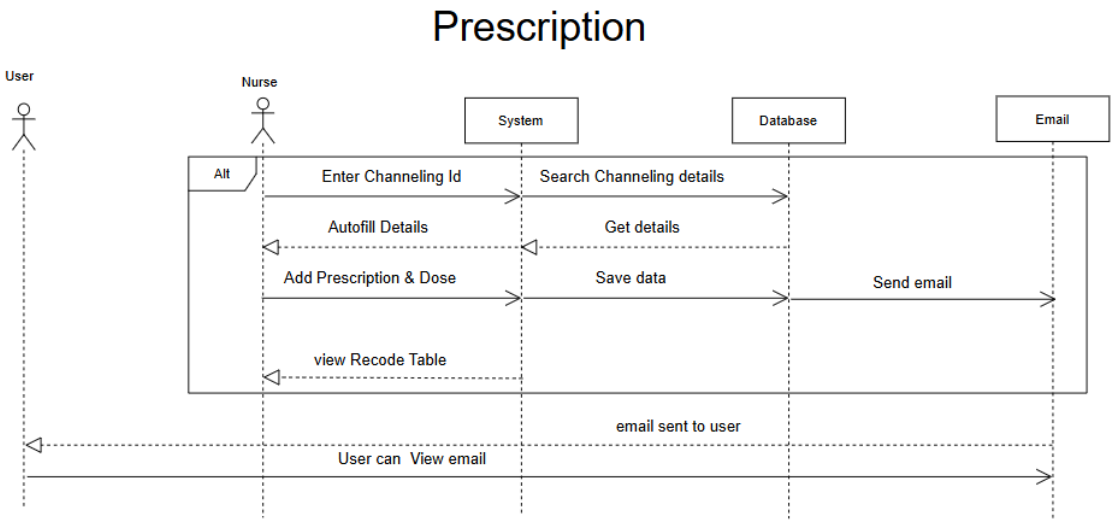

Figure 9: Sequence Diagram for Prescription issuing

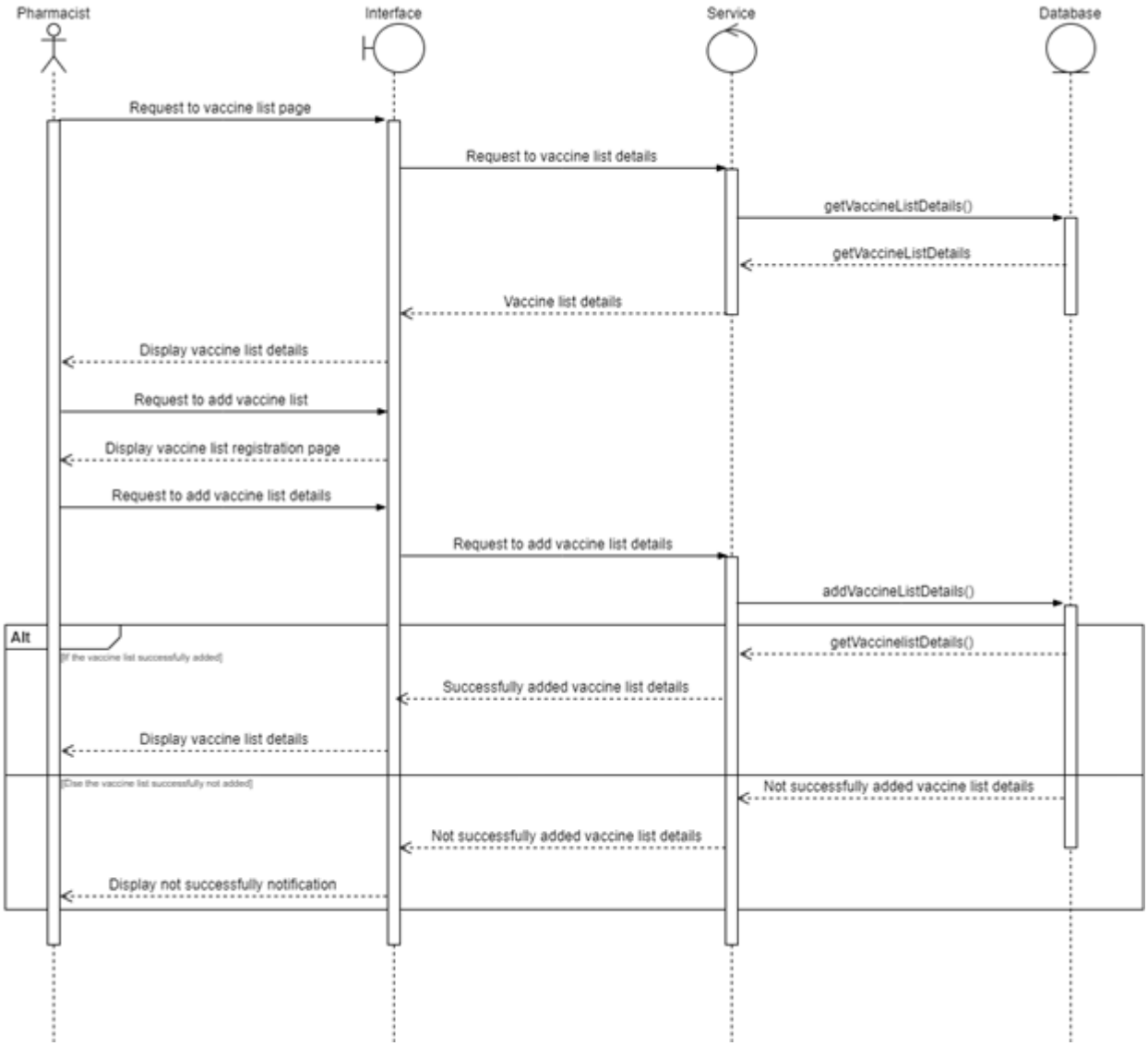

Figure 10: Sequence Diagram for Vaccine List

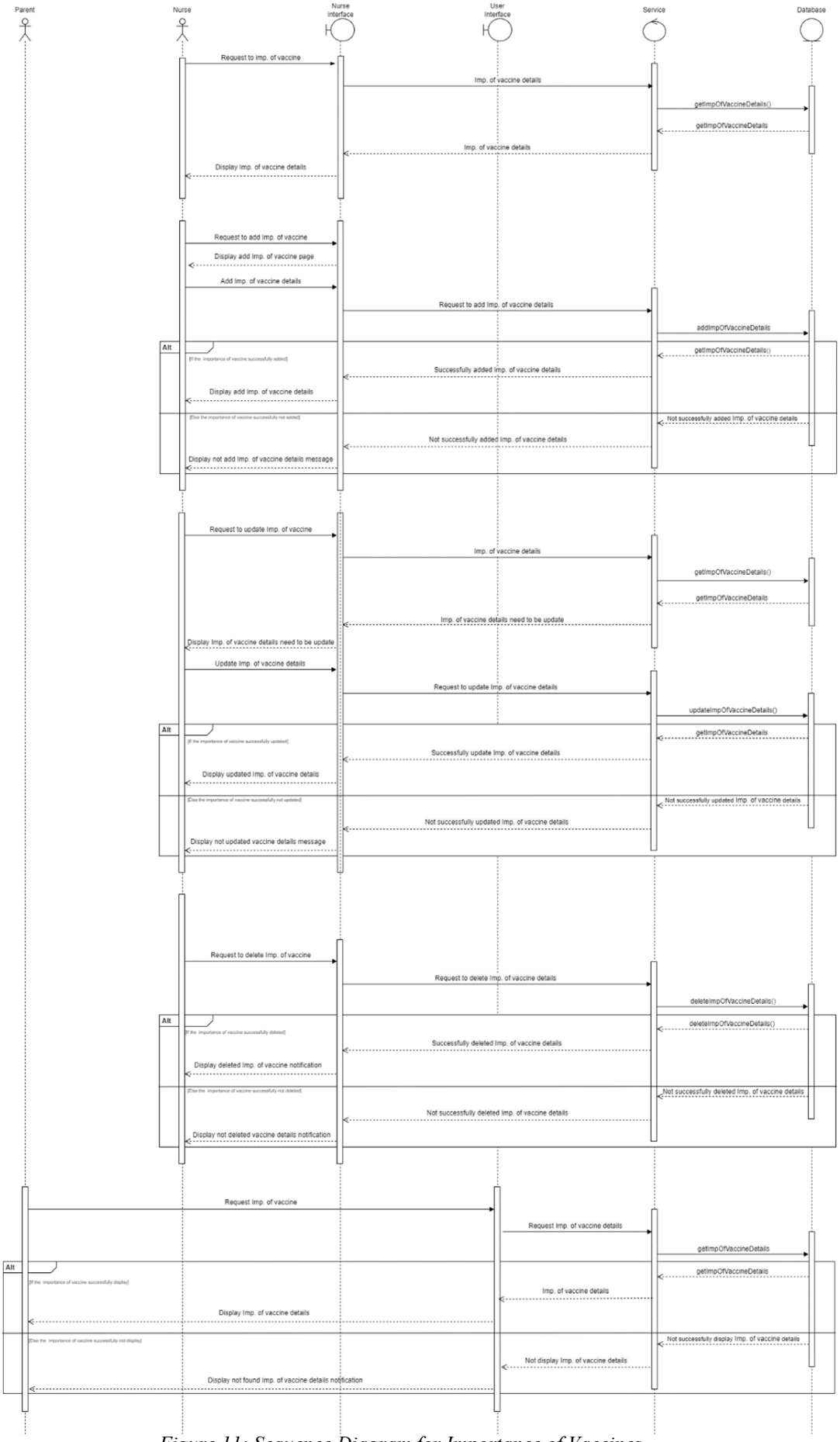

Figure 11: Sequence Diagram for Importance of Vaccines

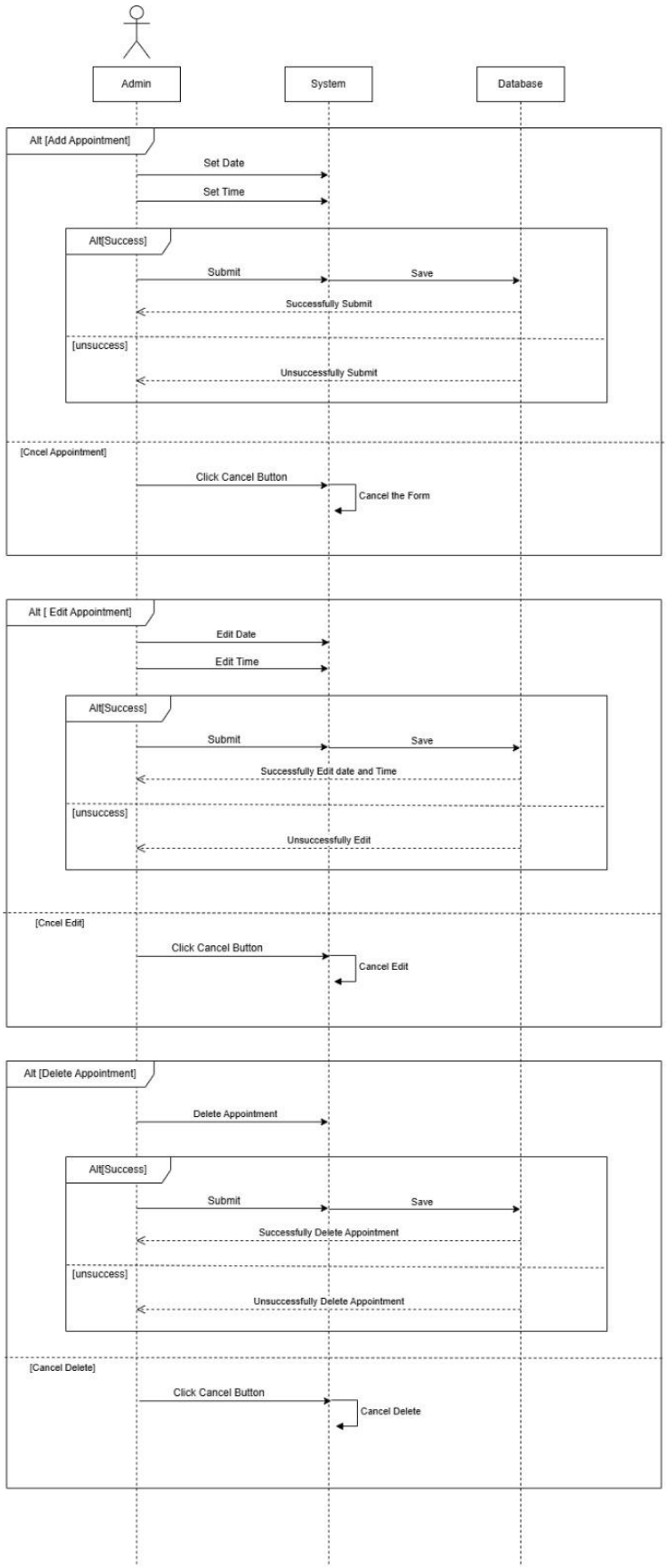

Figure 12: Sequence Diagram for Assigning Doctors for Appointments

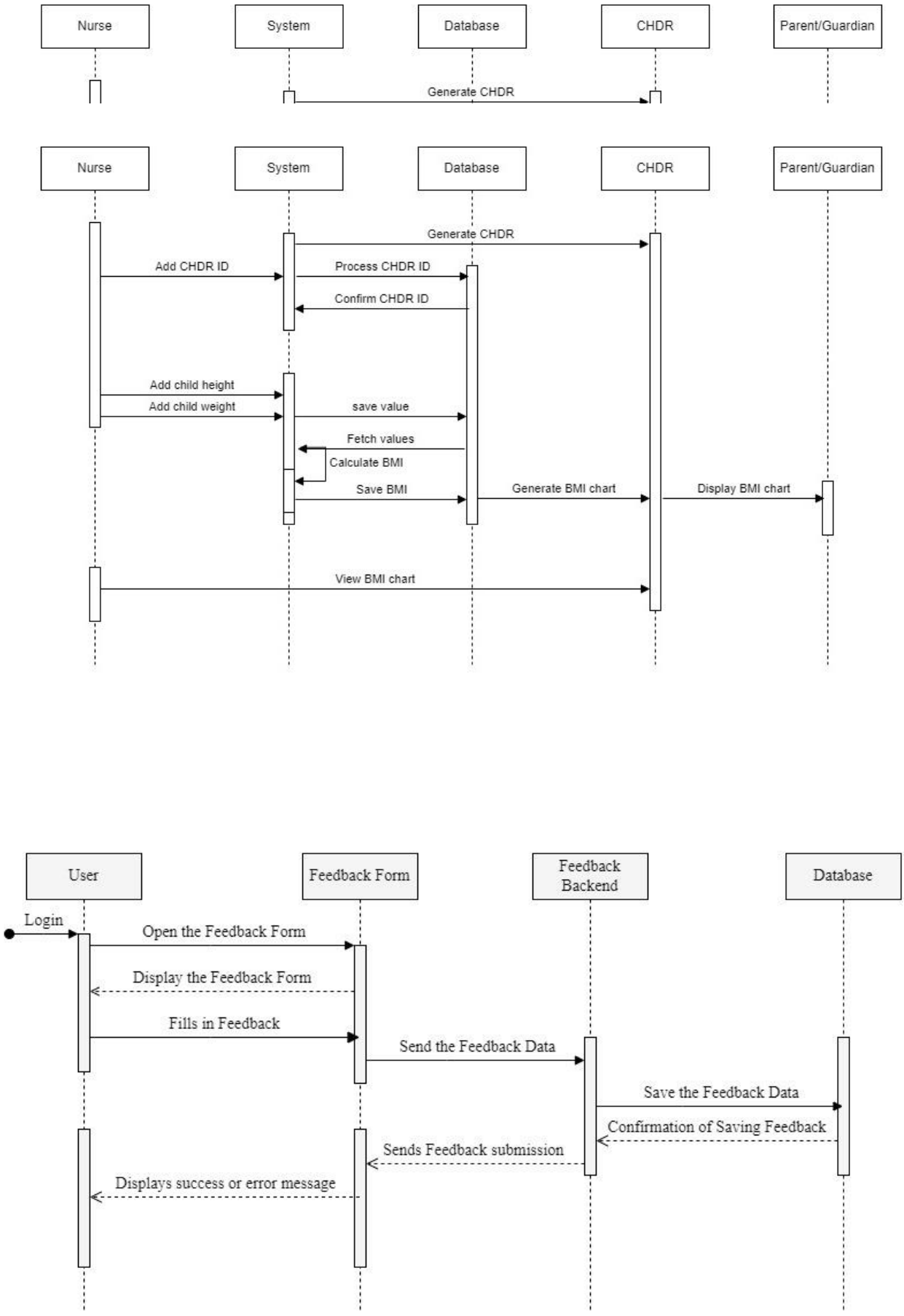

Figure 14: Sequence Diagram for Feedback

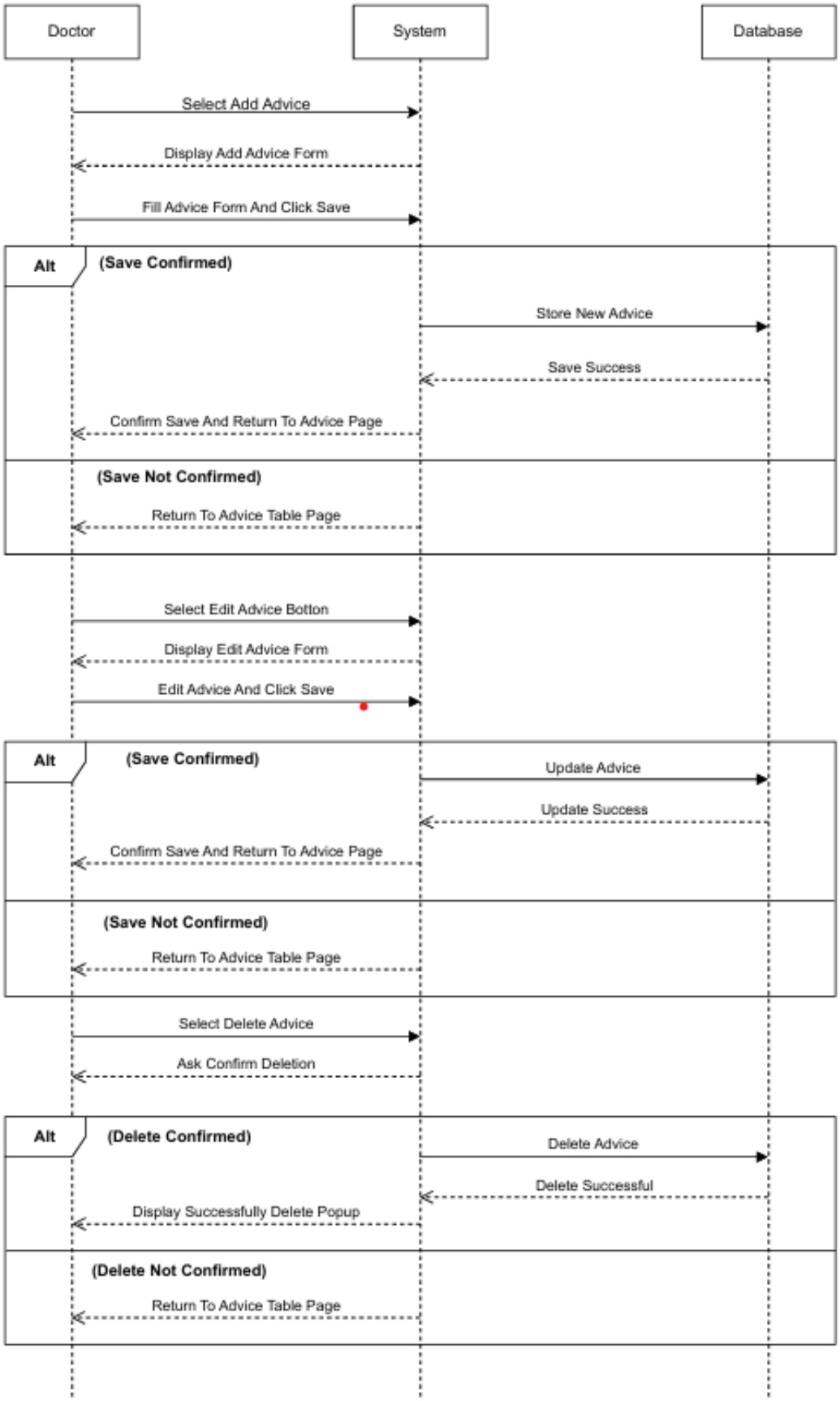

Figure 15: Sequence Diagram for Advises Based on BMI

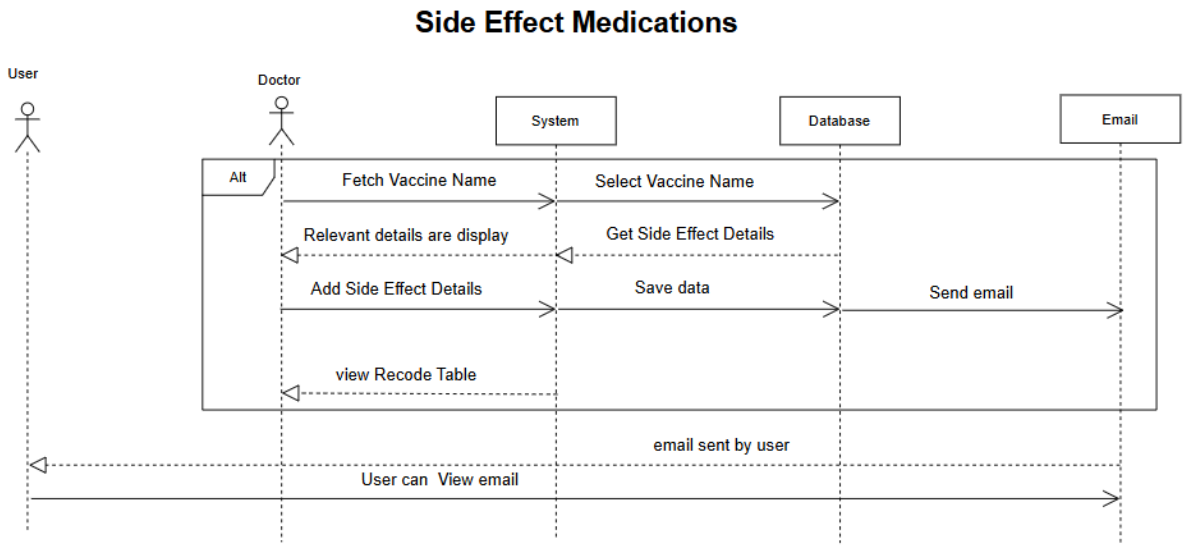

Figure 16: Sequence Diagram for Side Effect Medications

Figure 17: ER Diagram

2.3. Implementation

2.3.1. Module Structures

7Guard is built using following modules for each functionality.

- User Registration:

  Staff is registered by the administrators. They are given log in credentials (username and password) through their emails. They can update the password if needed. According to the staff's job titles, they are assigned to their profiles. Admin can view staff list and update their details if the staff need to update their details. Nurse registers the parent by sending OTP to their emails to use it as the password. Nurse also registers the children based on the parent NICs registered to the system. This process creates a CHDR card inside the parent profile. If the parent is logging in for the first time, they must update their OTP into a new password and log in to the system again with it. They can view their profiles after that. That profile includes the CHDR card. If parents need to update the profiles, they can update it through the nurse.

- Assigning Doctors for Appointments:

  After selecting the doctor ID, the name is auto loaded to the form. Then the doctor's available time and date are added by the admin. The number of appointments that doctor has is also shown. After submitting a successful message is displayed as well. Admin can also delete the appointment or update doctor's available time and number of appointments per day. Parents can view those submitted details at last.

- Vaccine Appointments:

  The channeling feature in the system enables parents or guardians to schedule normal or vaccine-specific appointments. For vaccine appointments, users can choose to reserve a vaccine from the hospital or bring one from a third party. If ordering from the hospital, the system notifies the user of vaccine availability. Vaccine appointments are restricted to dates based on the doctor's schedule. After submitting the channeling form with basic details, users receive an email confirmation with a channeling number, fee, and doctor details. Payment of the installment is mandatory before the appointment, even if a

channel number is received. The hospital receptionist manages the admin side, storing all channeling records in the database. After payment, the system sends a bill via email confirming the transaction. The receptionist can update or cancel appointments as needed. Additionally, the system enforces restrictions like a daily limit on the number of patients per doctor, notifying users when limits are reached.

- Displaying Vaccine List, Stock, and Importance:

  This is done by assuming the hospital already has a main vaccine stock management system. Pharmacist can view vaccine stock list page that displays vaccine name and its quantity. Nurse can add importance of vaccine to a table view, update, and delete those details if needed. Parent can view these details through their interface. Parent can also view the vaccine list and importance of vaccine categorized by the age range based on weeks. CHDR card includes vaccine list as well.

- Vaccine Scan:

  Channel ID is entered first in the injected form. Then CHDR ID, vaccine purpose, vaccine name are automatically added to the form. If the vaccine purpose is outside vaccination, the name of that vaccine is shown. Then the vaccine barcode is scanned, and the details are saved to the database. Then the CHDR is updated as that specific vaccine was injected.

- Calculating BMI:

  This covers the calculation BMI part. When the CHDR ID, child weight, child height, is entered the BMI is automatically calculated. BMI status (Obese, Overweight, Underweight, Normal, Mildly underweight, Moderately underweight, Severely underweight) will also save based on the value and the range it matches. When the CHDR ID is entered to load the graph, a simple graph will be generated with the relevant data. The graph is also displayed through the CHDR card.

- Health Advises Based on BMI:
  Doctors adds advises based on BMI status of children. These advises are sent to the parents through their email.

- Health Advises Based on Vaccination Side Effects:
  To provide reassurance to parents following vaccination, an email is sent detailing the potential side effects that may occur based on the administered vaccine. Additionally, the email includes recommendations for appropriate medications to manage these side effects, ensuring that parents feel informed and supported during this time. The doctor can edit or delete details if needed.

- Vaccination Appointment Reminders:
  Sends reminders to the parent a week before the vaccination appointment is scheduled.

- Feedback:
  User can send feedback about the services they receive.

2.3.2. Development Tools

- IDE: IntelliJ and Visual Code Studio was used for coding and management.
- Frameworks: Backend was done using SpringBoot and frontend was built using React.js for interfaces.
- Libraries used: Chart.js was used to create the BMI graph. Emailjs is used to send emails from the frontend.

2.3.3. DMBS Choice

7Guard uses MySQL as the database management system due to its efficiency, reliability, and performance.

MySQL provides an easy and user-friendly environment for managing databases. It makes the designing and implementing the database easier. It is also reliable and can handle a growing number of data records. It supports SQL that works with complex queries for managing vaccine

records. It also has high performance which is suitable for a large database like 7Guard's. Since MySQL is open source and has a large developer community, it has access to resources, tutorials, and other troubleshooting support the developers need. MySQL's security features ensure that the sensitive data is secure. 7Guard database is designed to store, retrieve, and manage data records efficiently.

2.3.4. Implementation Languages

7Guard uses a combination of modern programming languages to create a strong and efficient platform. Each language was selected to fulfill the requirements of backend, frontend and the overall system.

- JAVA: Served as the primary language for the backend development. It is implemented using SpringBoot. It was chosen because of it is known for its robustness and scalability which are suitable for building a secure high-performance backend.

- JavaScript: Used for the dynamic interactions between the user interface and backend. This makes features like sending alerts and use of console logs for error detecting are much easier.

- CSS with Bootstrap: Used for the structuring and styling the interfaces. Combined with Bootstrap, CSS ensures the interface is visually appealing and responsive with devices. It ensures a uniform design for all pages.

- SQL: Used to write queries to manage data stored in the MySQL database.

All these programming languages ensure that 7Guard works efficiently to fulfill the requirements.

2.3.5. Special Algorithms

- The BMI is calculated by its standard formula.

```
public void calculateBmi() {  1 usage
    if (childHeight != null && childWeight != null && childHeight > 0) {
        double heightInMeters = childHeight / 100.0;
        this.bmiValue = childWeight / (childHeight * childHeight);
        this.date = LocalDate.now();
```

Figure 18: BMI calculation

- The age of children is separated into weeks to display in the importance of vaccine table shown in the home page.

```
useEffect(() => {

  if (child.dob) {

  let today = new Date();
  let birthday = new Date(child.dob);
  let age=today-birthday;

  let agetime=age/ (1000 * 60 * 60 * 24 * 7);
  let ageweek=agetime.toFixed(2);
  setChildAge(ageweek);}

}, [child]);
```

Figure 19: Age separation Code

- Separate time into fifteen minutes slots for assigning doctors.

```
const [startHours, startMinutes] = doctor.startTime.split(':').map(Number);
const [endHours, endMinutes] = doctor.endTime.split(':').map(Number);


if (isNaN(startHours) || isNaN(startMinutes) || isNaN(endHours) || isNaN(endMinutes)) {
  console.error('Invalid time format');
  return;
}

const startTimeInMinutes = startHours * 60 + startMinutes;
const endTimeInMinutes = endHours * 60 + endMinutes;

if (endTimeInMinutes < startTimeInMinutes) {
  console.error('End time must be greater than start time');
  return;
}


const totalAvailableTime = endTimeInMinutes - startTimeInMinutes;
console.log('Total Available Time (minutes):', totalAvailableTime);


const allocatedTime = 15
const userTime = channelings.channel_number * allocatedTime;


const channelTimes = startTimeInMinutes + userTime;
```

Figure 20: Time Slots Code Picture 1

```
if (channelTimes < 0 || channelTimes > 1440) {
  console.error('Calculated time is out of bounds:', channelTimes);
  return;
}
                              const channelTimes: any
const finalHours = Math.floor(channelTimes / 60);
const finalMinutes = channelTimes % 60;


const formattedTime = `${String(finalHours).padStart(2, '0')}:${String(finalMinutes).padStart(2, '0')}:00`;
const totalHours = finalHours + (finalMinutes / 60);

setTime(formattedTime)
  console.log('Formatted Time:', formattedTime);


  const finalTimeInDouble = channelTimes / 60;
  console.log('Final Time as Double (hours):', finalTimeInDouble);
}
}, [doctor, channelings, time]);

console.log("time",time)
```

Figure 21: Time Slots Code Picture 2

2.4. Testing

Testing was done through Postman. Below are some of the Postman response bodies that were used to test the functions.

- Loading data to the CHDR.

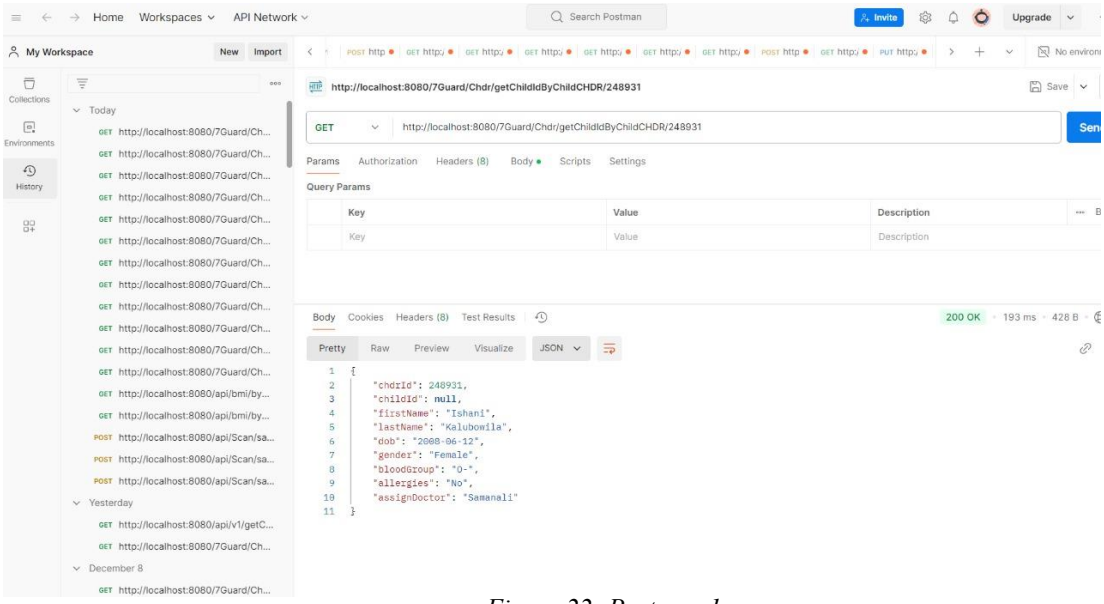

Figure 22: Postman 1

- After entering the channeling ID, the fields of injected vaccine form are auto filled.

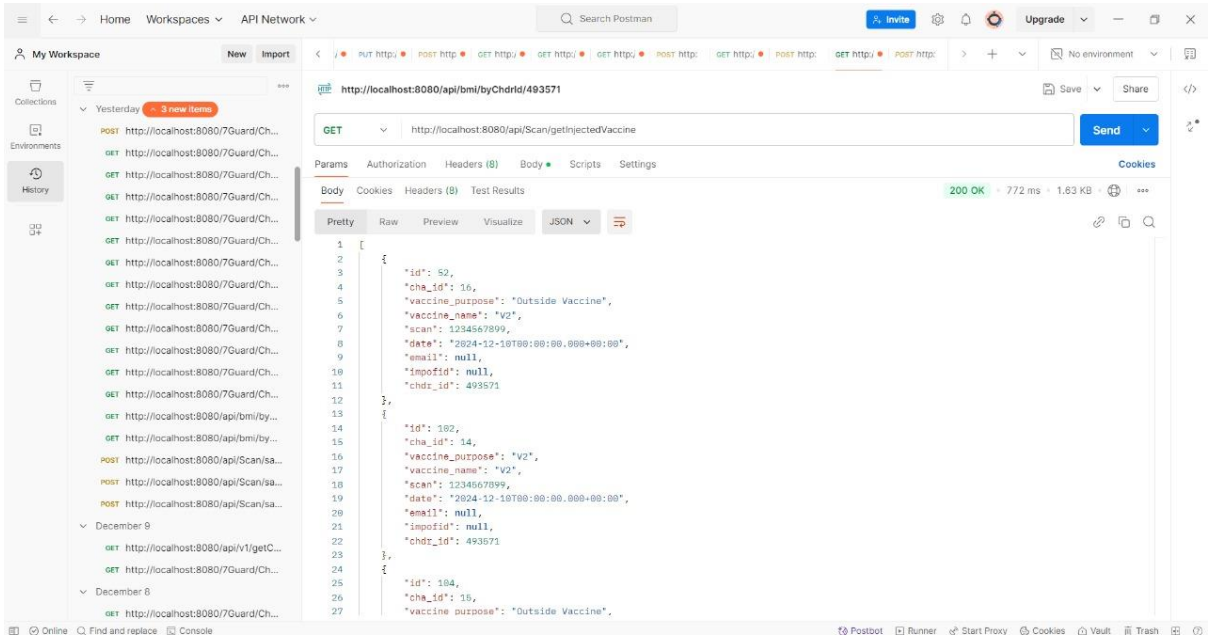

Figure 23: Postman 2

- Data to the vaccine stock table is loaded.

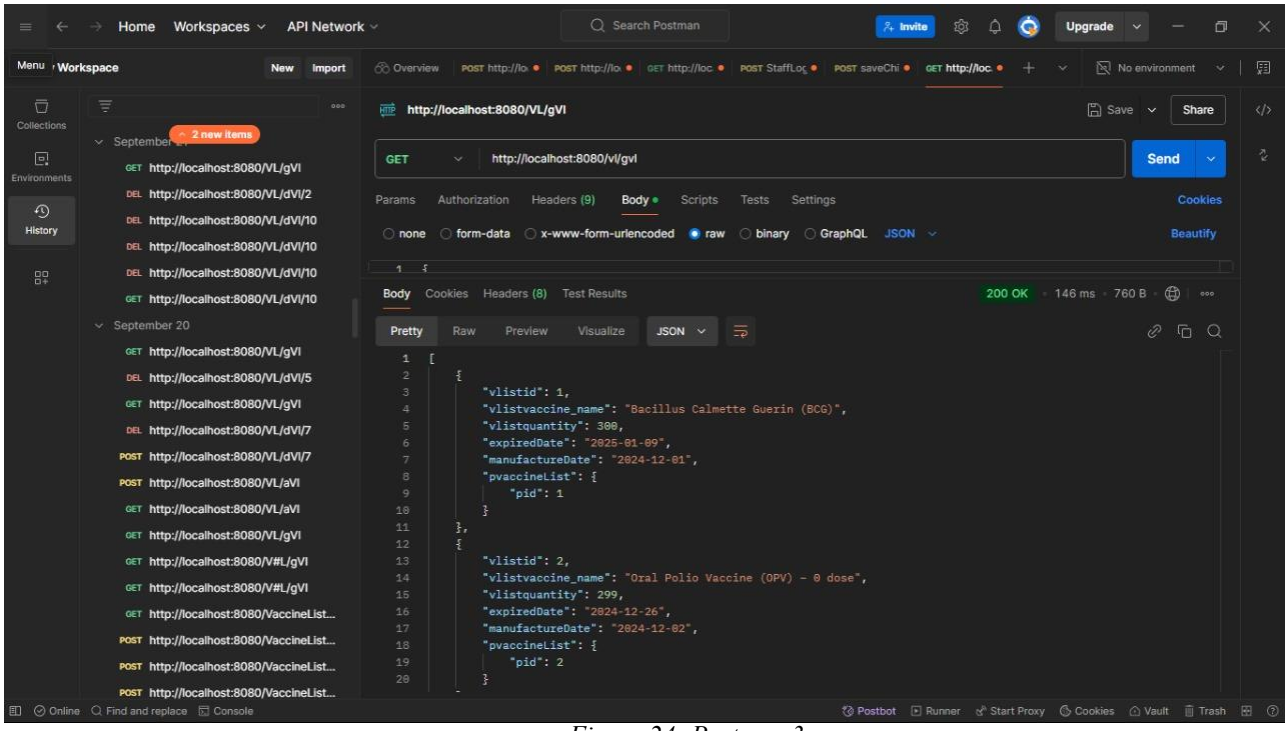

Figure 24: Postman 3

- Parent is registered with the form's data.

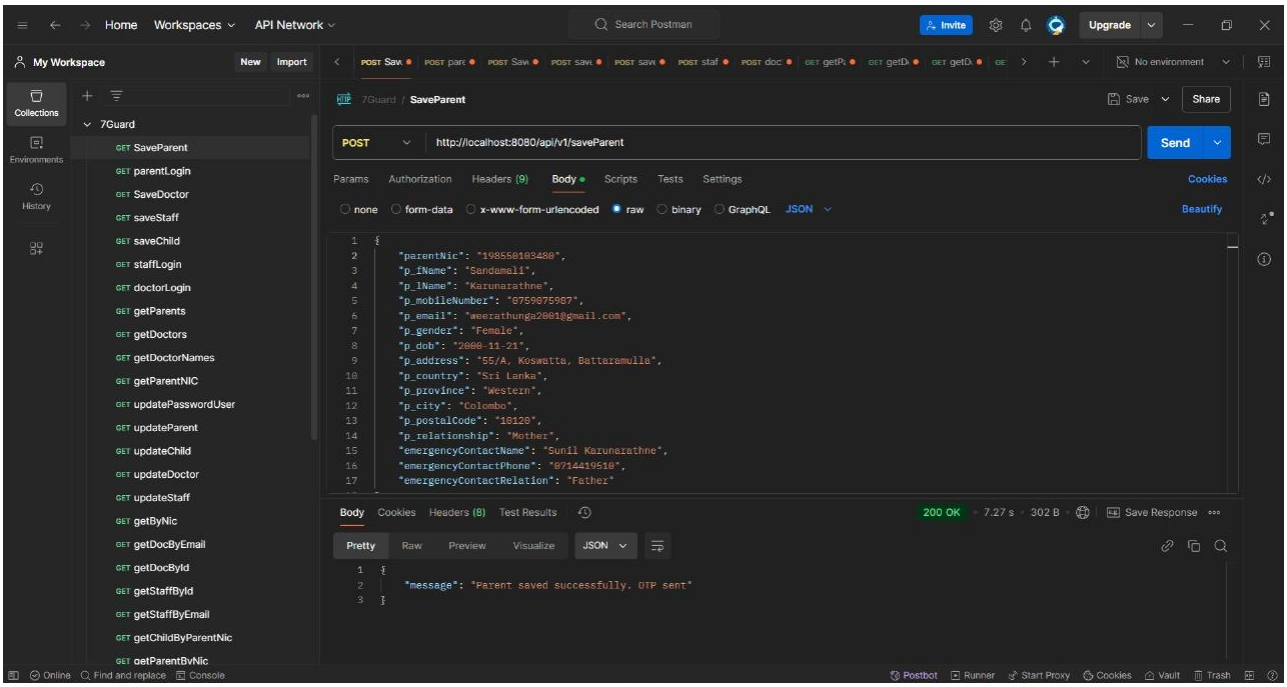

Figure 25: Postman 4

3. Evaluation

3.1. Assessment of the Project results

3.1.1. Techniques Used to Analyze Data

- Functional Testing:
  Conducted testing the functions individually and the system. Each component was tested by the developers individually before implementing the whole system. Then the system was tested with a trial round.
- Performance Testing:
  Observed the system response times and data retrieval speed to check the performance of the system.

3.1.2. Results of Analysis

The system successfully met the end goals of the functions. Even though the overall system works properly, the developers team aspire to add more testing and run the system through a thorough testing process.

3.1.3. Deficiencies in the Final Product

- The system only supports English language, restricting the usability for users lacking English language skills.
- Currently there is no barcode reader to test the vaccine codes, even though the system is designed to work seamlessly with scanning via a reader or manually typing the code.
- The system cannot be accessed offline.

3.1.4. Improvements to Make

- Consider modifying the system to be multilingual with translation features.
- Develop offline capabilities at least for the CHDR card.
- Have usability testing to improve the user interface.
- Having a barcode reader and do proper scanning.

3.2. Lessons Learned

- Need to have a better understanding and connection with the client to identify proper requirements. Having constant feedback from the client is much more beneficial.
- Should manage time properly to follow the exact project plan.
- The team should have neat collaboration and constant updated among each other to identify the changes and current state of the system.

3.3. Future Work

- Current system uses emails to send notifications to the users. The team aims to change that and add viewing notifications feature to a dashboard. That way, the users will be able to easily be notified.

4. Timeline

5. Conclusion

The purpose of 7Guard is to streamline and modernize the child vaccination process by providing an online digital platform. It facilitates the users with easy channeling or appointment scheduling, online accessible CHDR card that included the child's health records, secure vaccination record histories, personalized health advises, and ability to communicate with the hospital through user feedback. The project covers these services and meet its objectives by strengthening efficiency, accuracy, accessibility, and interactions between the parent or guardians and the healthcare providers. Parents or the guardians are given an opportunity for easy health care while the healthcare providers are benefit from an easily manageable system. However, there were limitations that restrict the perfection of the system. These difficulties are covered through the identified improvements to make and the future work. Despite these highs and lows, the system offers benefits to all the stakeholders of the system and encourages the society to engage with improved child health care delivery. Overall, 7Guard rises with a promising step towards efficient child health care management.

6. References

[1] Credible Organization. “All you need to know about childhood vaccination.” Unicef for every child. https://www.unicef.org/southafrica/unicef-parenting/health/parents-frequently-asked-questions-vaccines (Accessed Oct 17, 2024).

[2] OpenAI. "ChatGPT". OpenAI, San Francisco, CA. https://chat.openai.com/ (Accessed Dec 11, 2024).

Appendix A: Design Diagrams

Use case diagrams for individual functions:

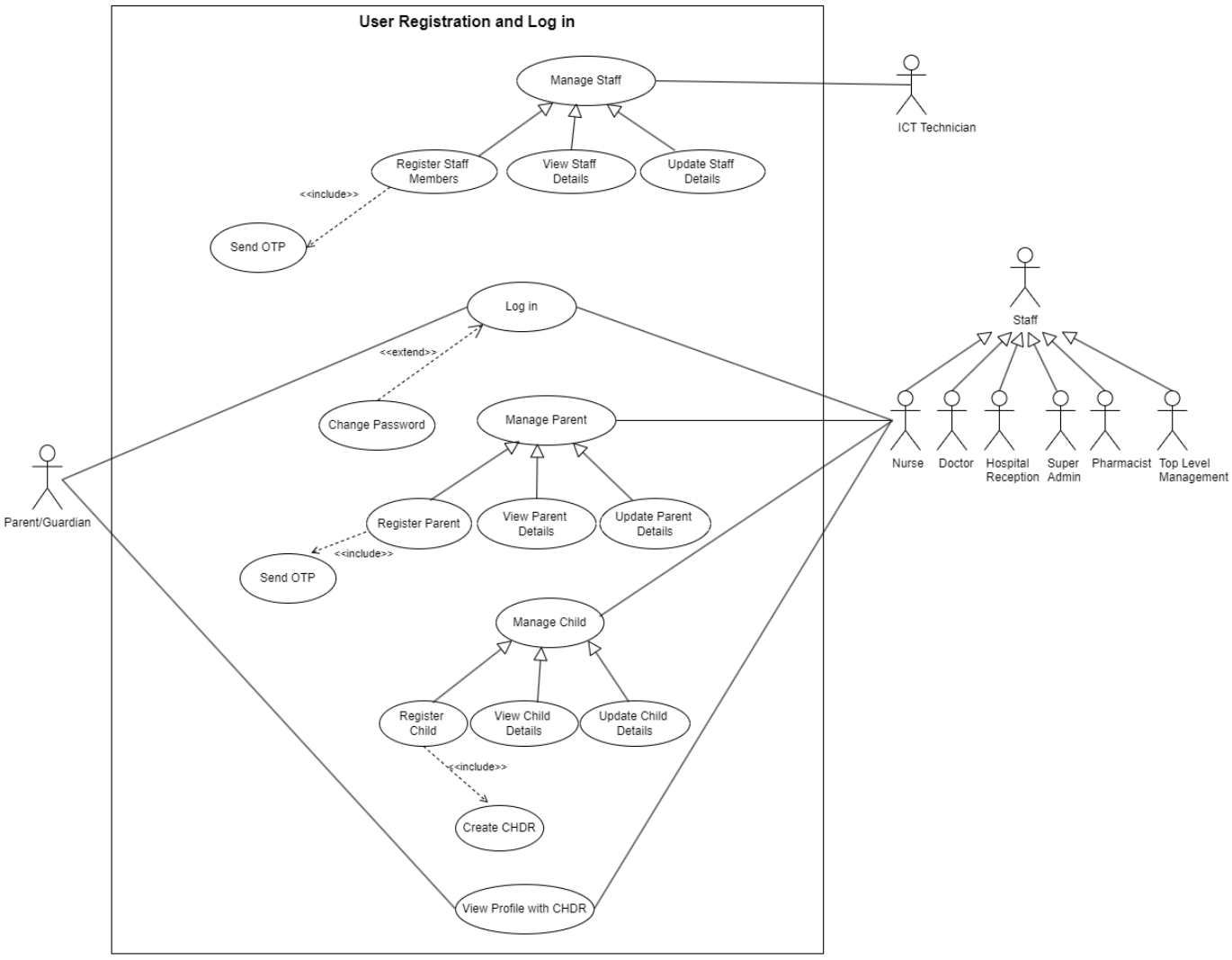

Figure 26: Use Case for User Registration

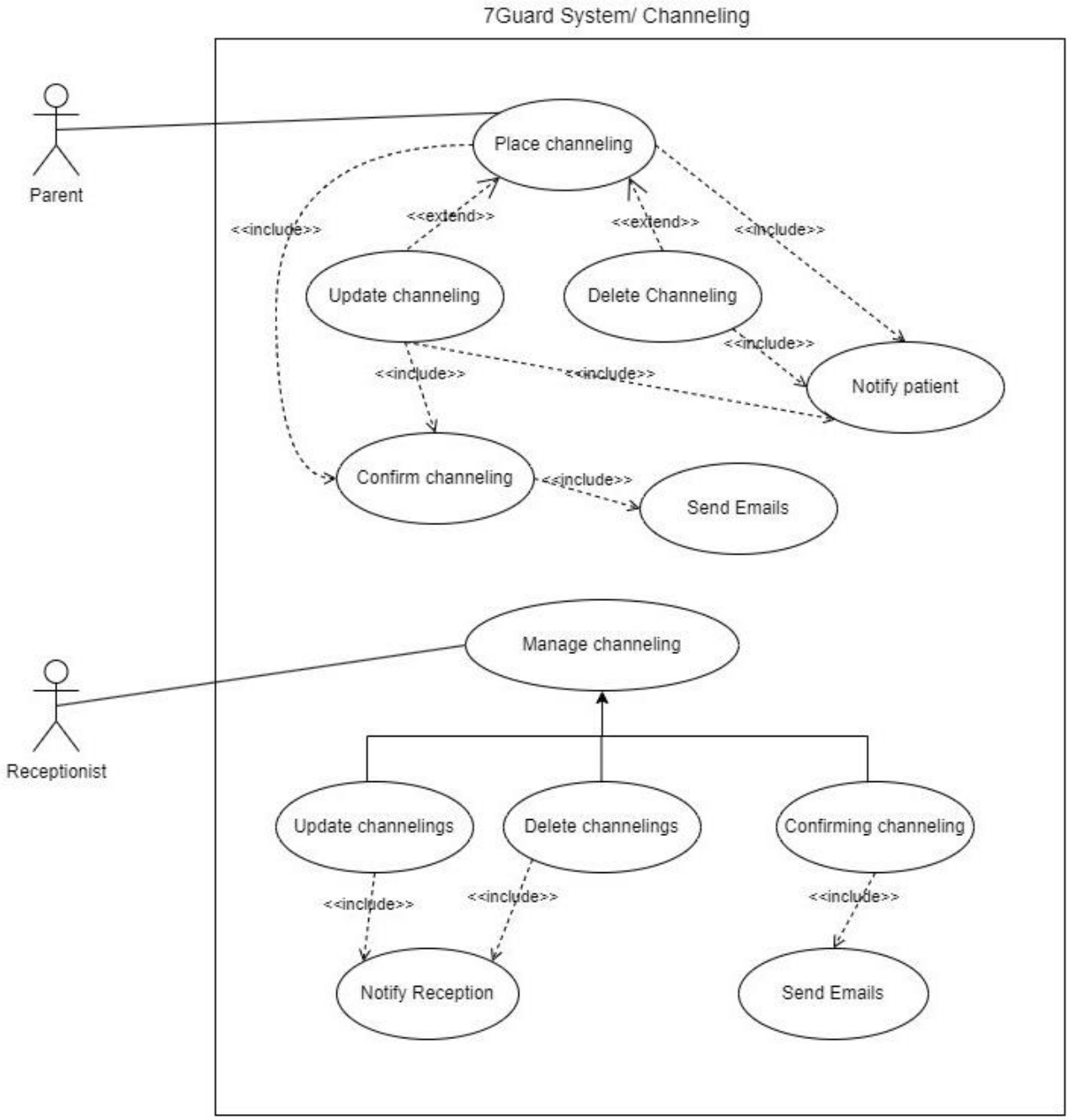

Figure 27: Use Case for Channeling

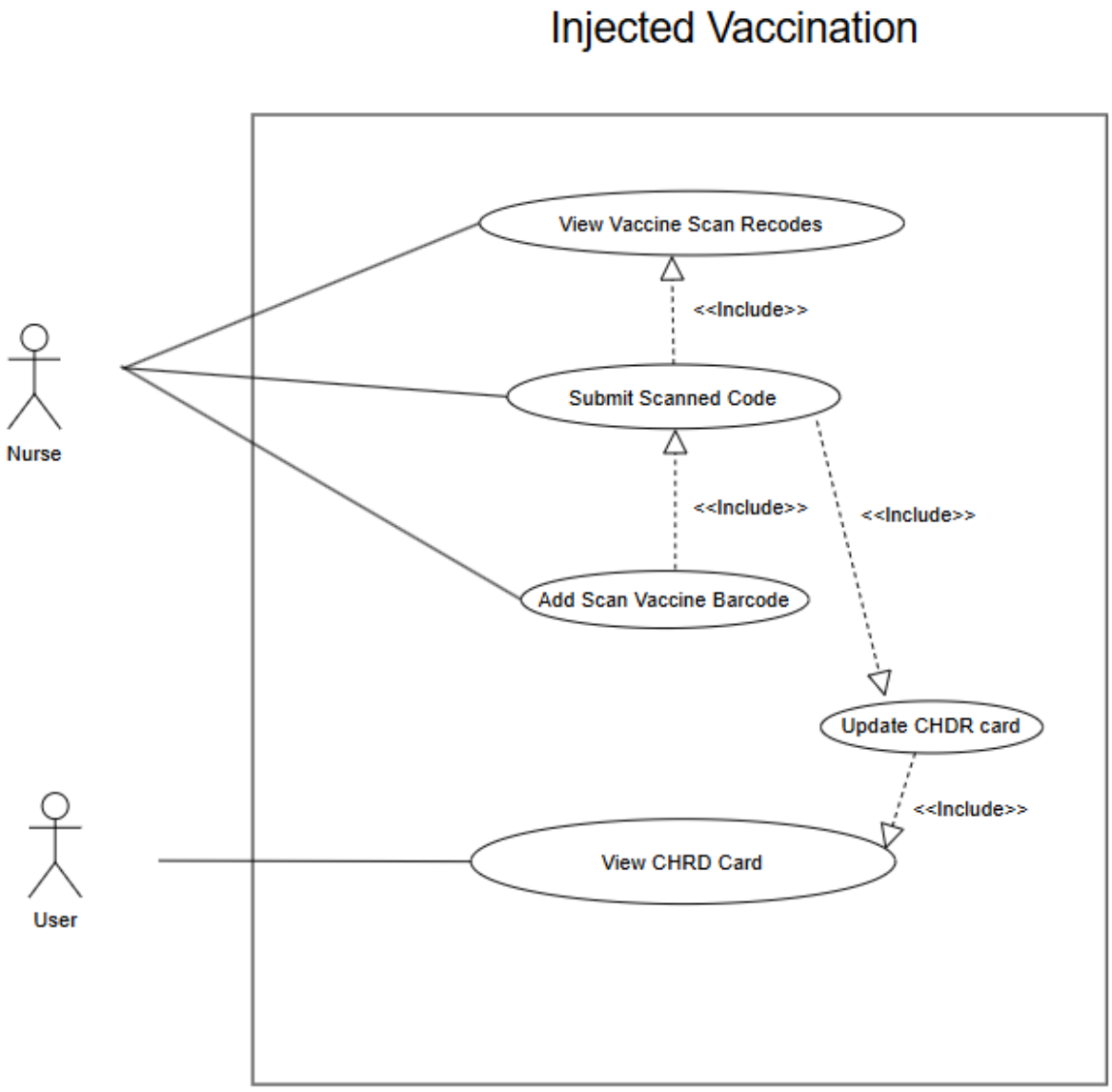

Figure 28: Use Case for Injected Vaccine

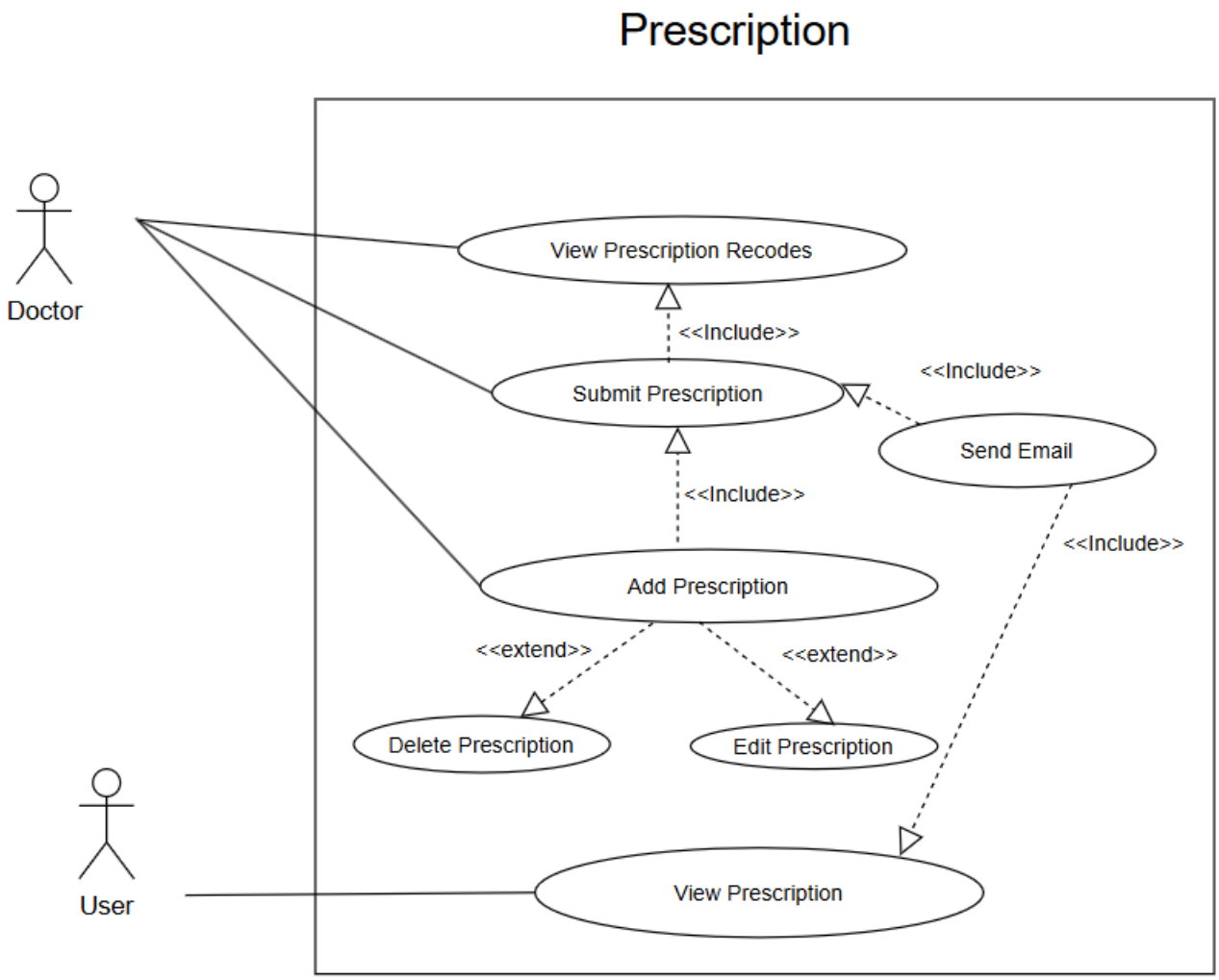

Figure 29: Use Case for Prescription

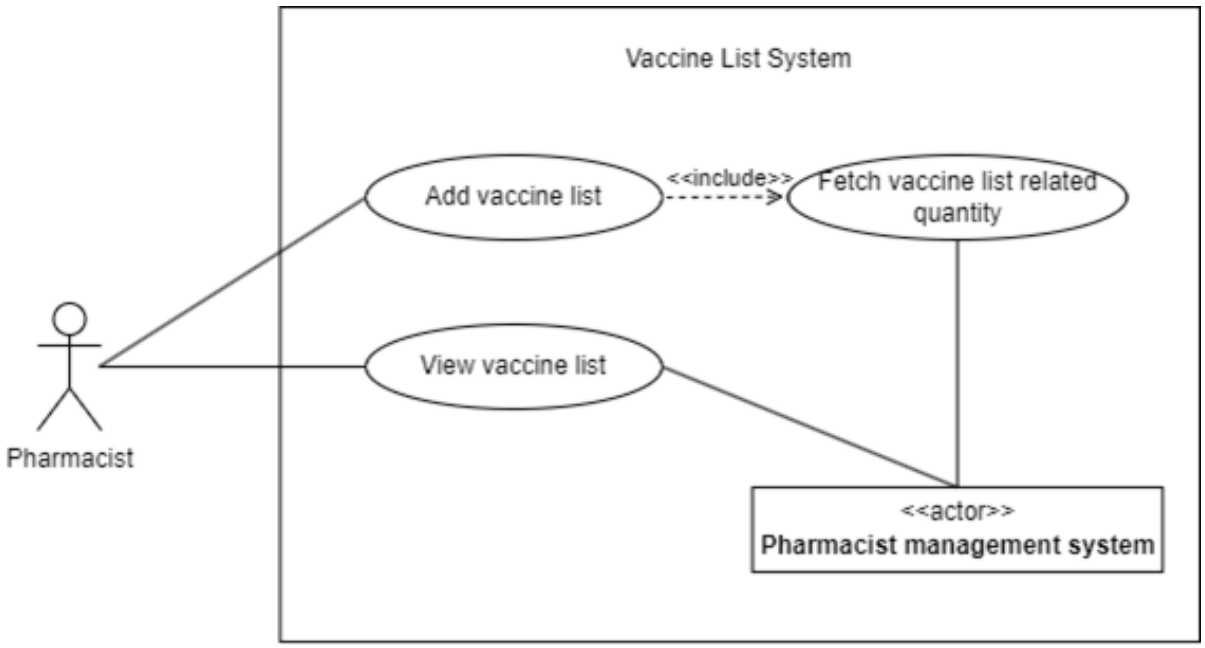

Figure 30: Use Case for Displaying Vaccine List

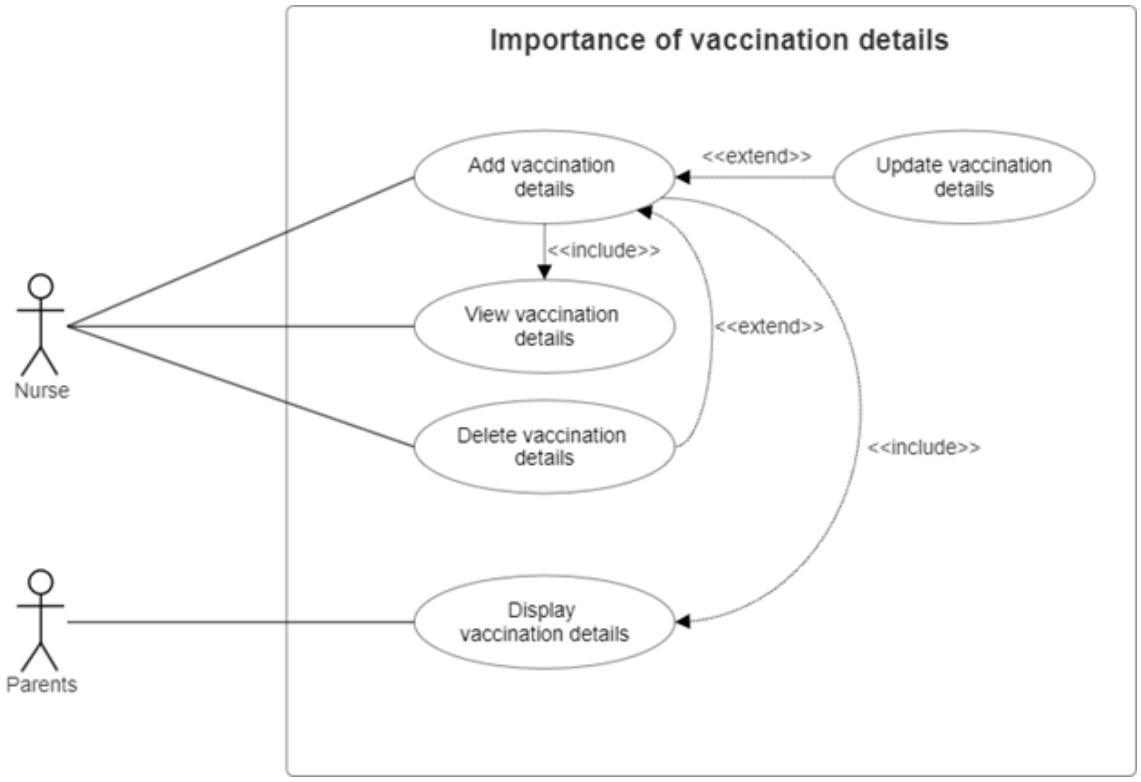

Figure 31: Use Case for Importance of Vaccine

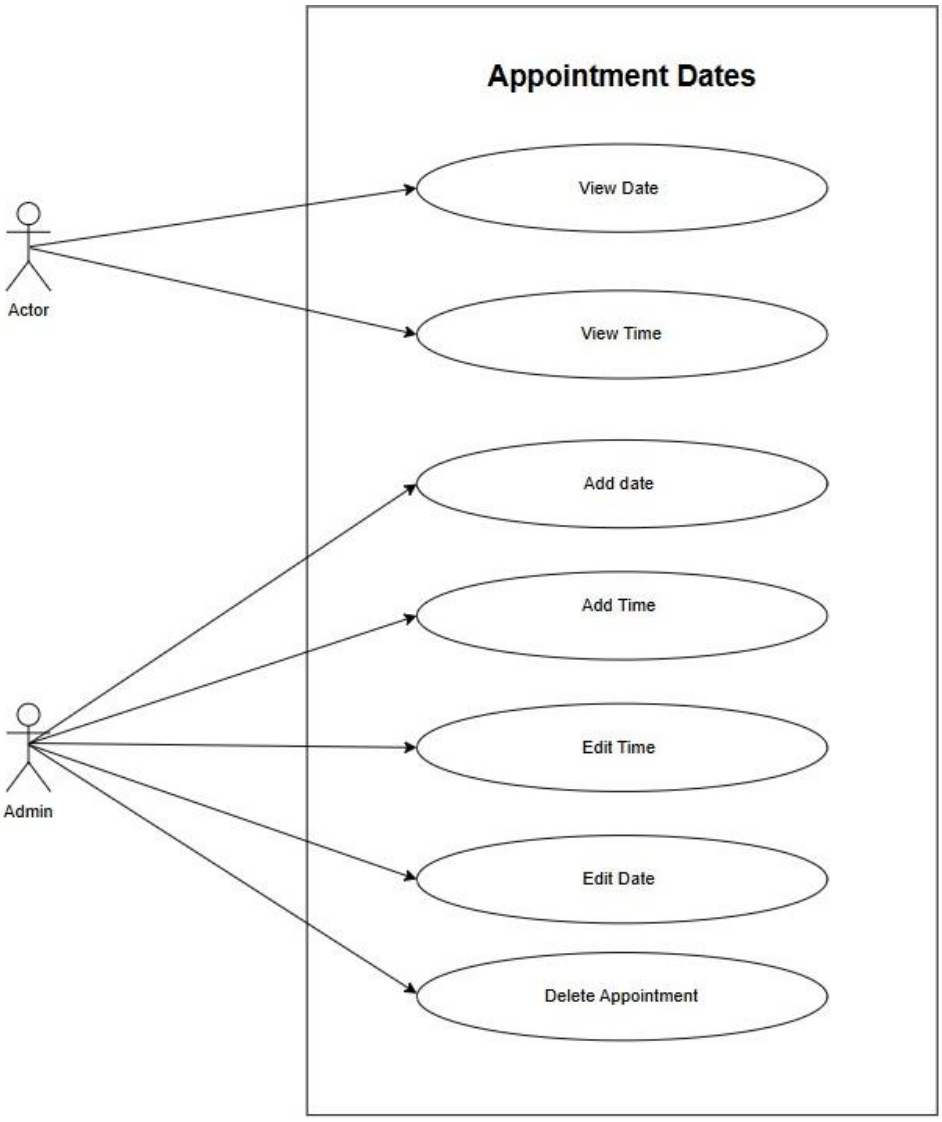

Figure 32: Use Case for Assigning Doctors for Appointments

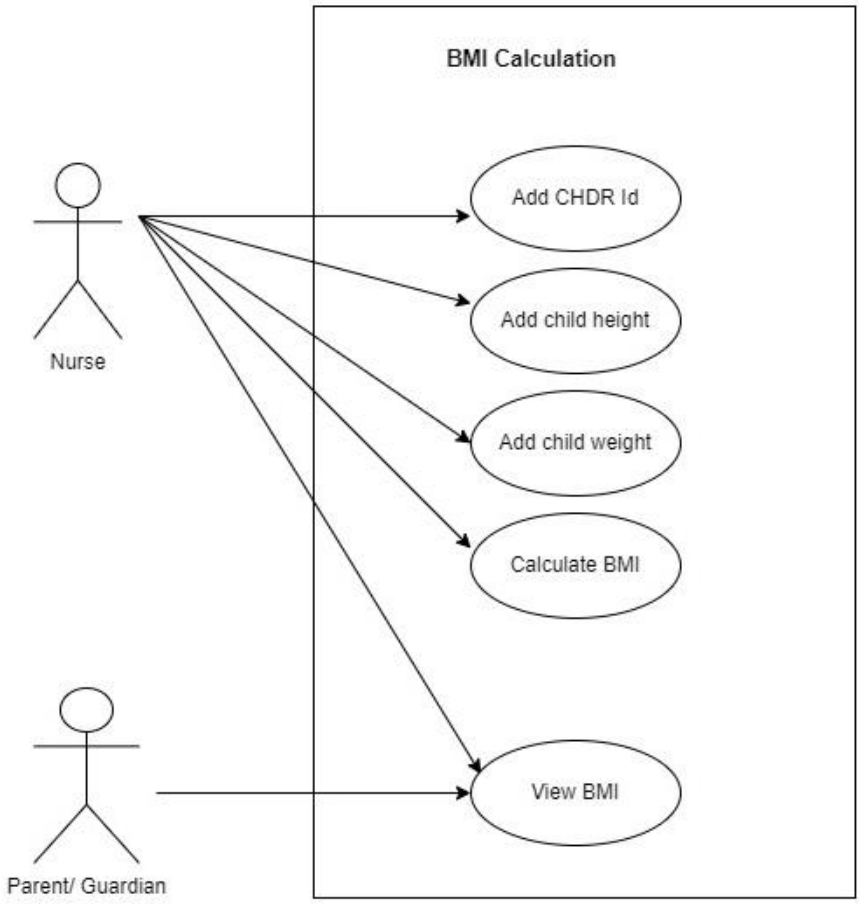

Figure 33: Use Case for BMI Calculation

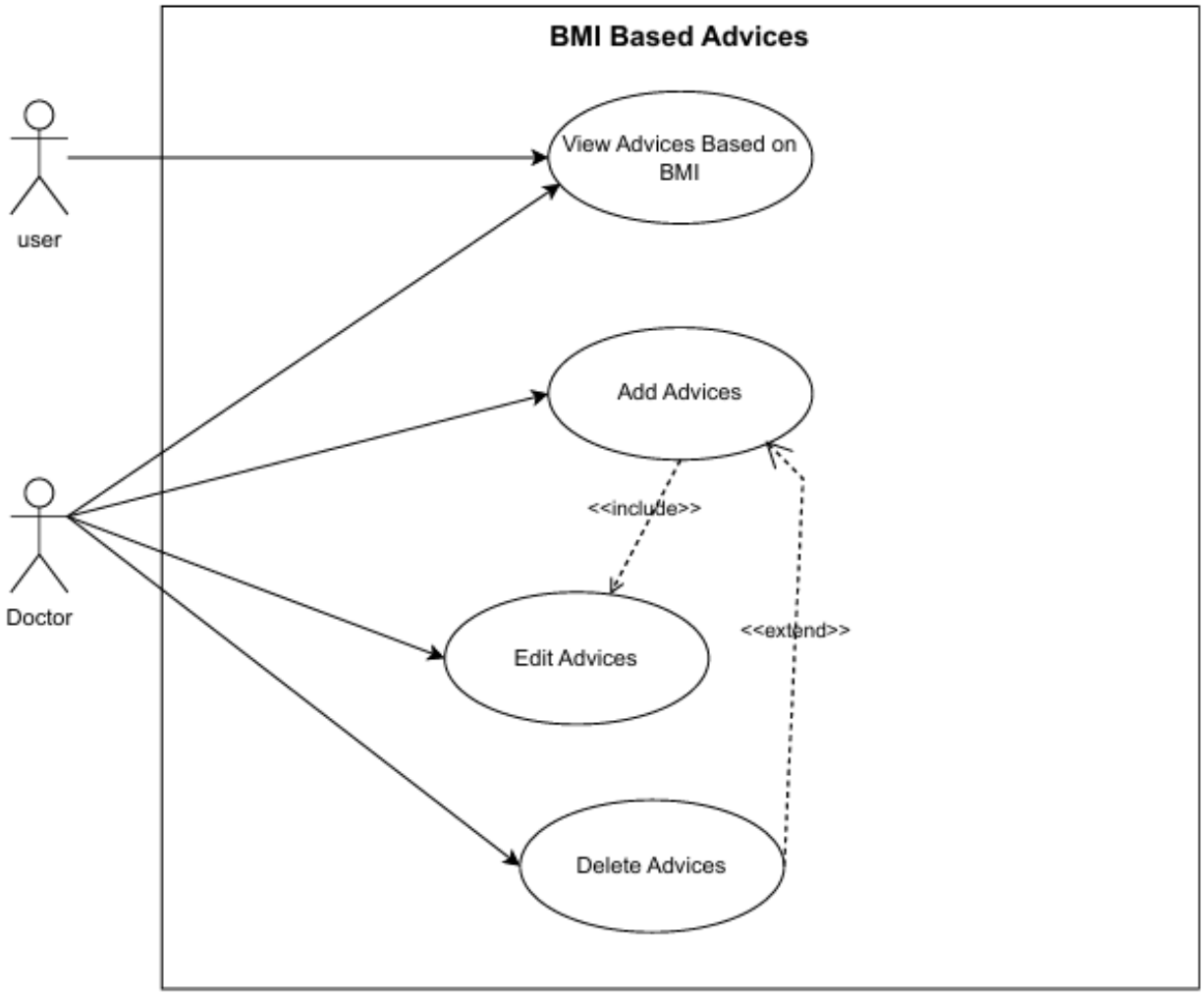

Figure 34: Use Case for BMI Based Advises

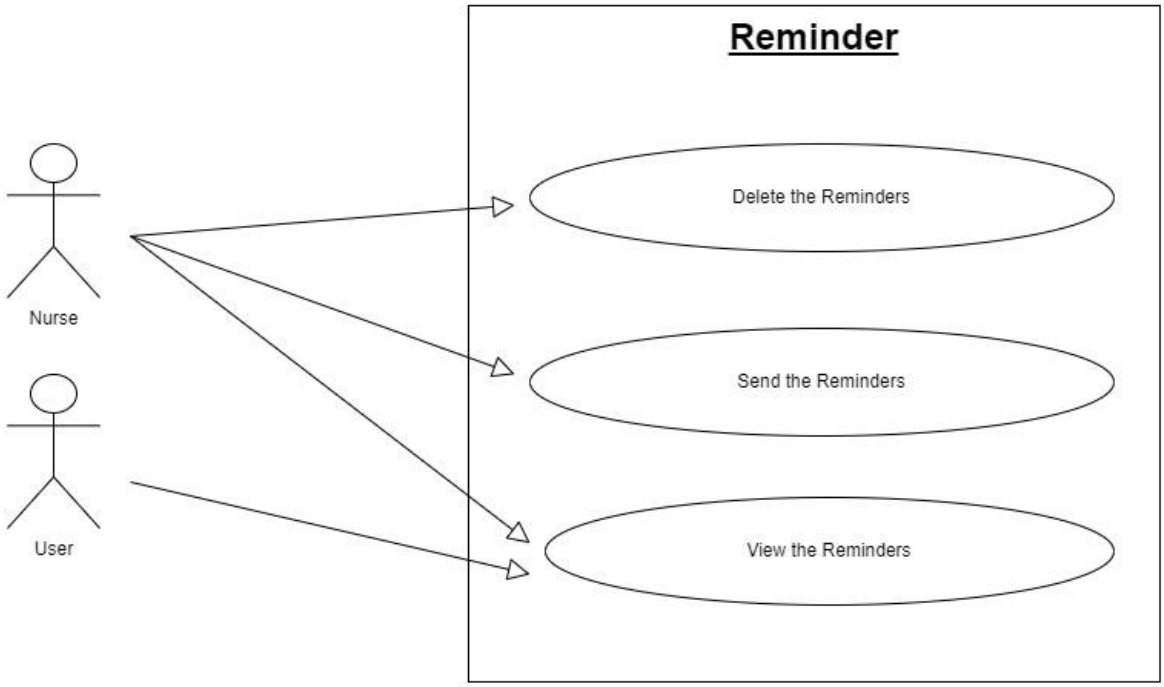

Figure 35: Use Case for Sending Reminders

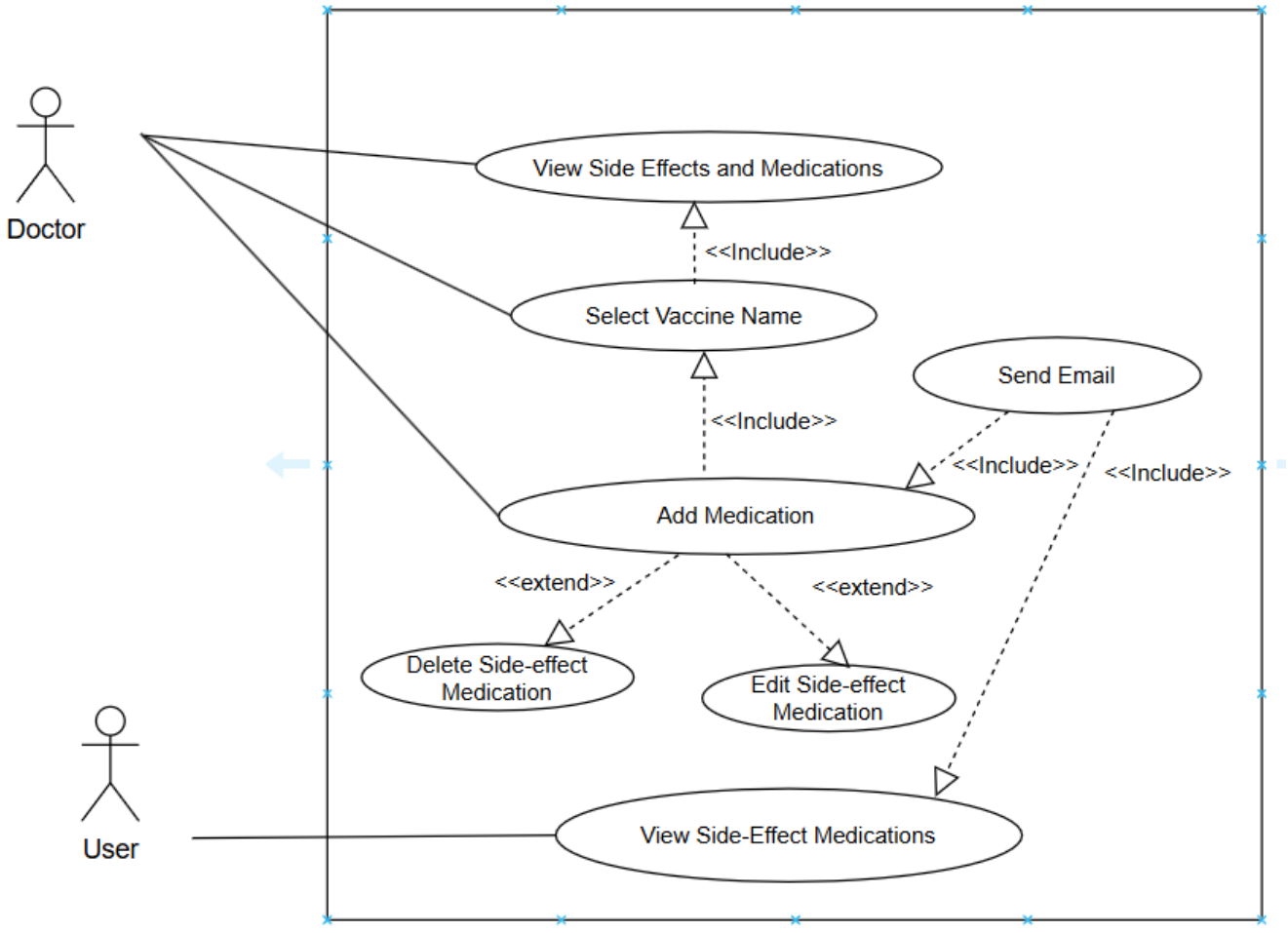

Figure 36: Use Case for Side Effect Medication Advises

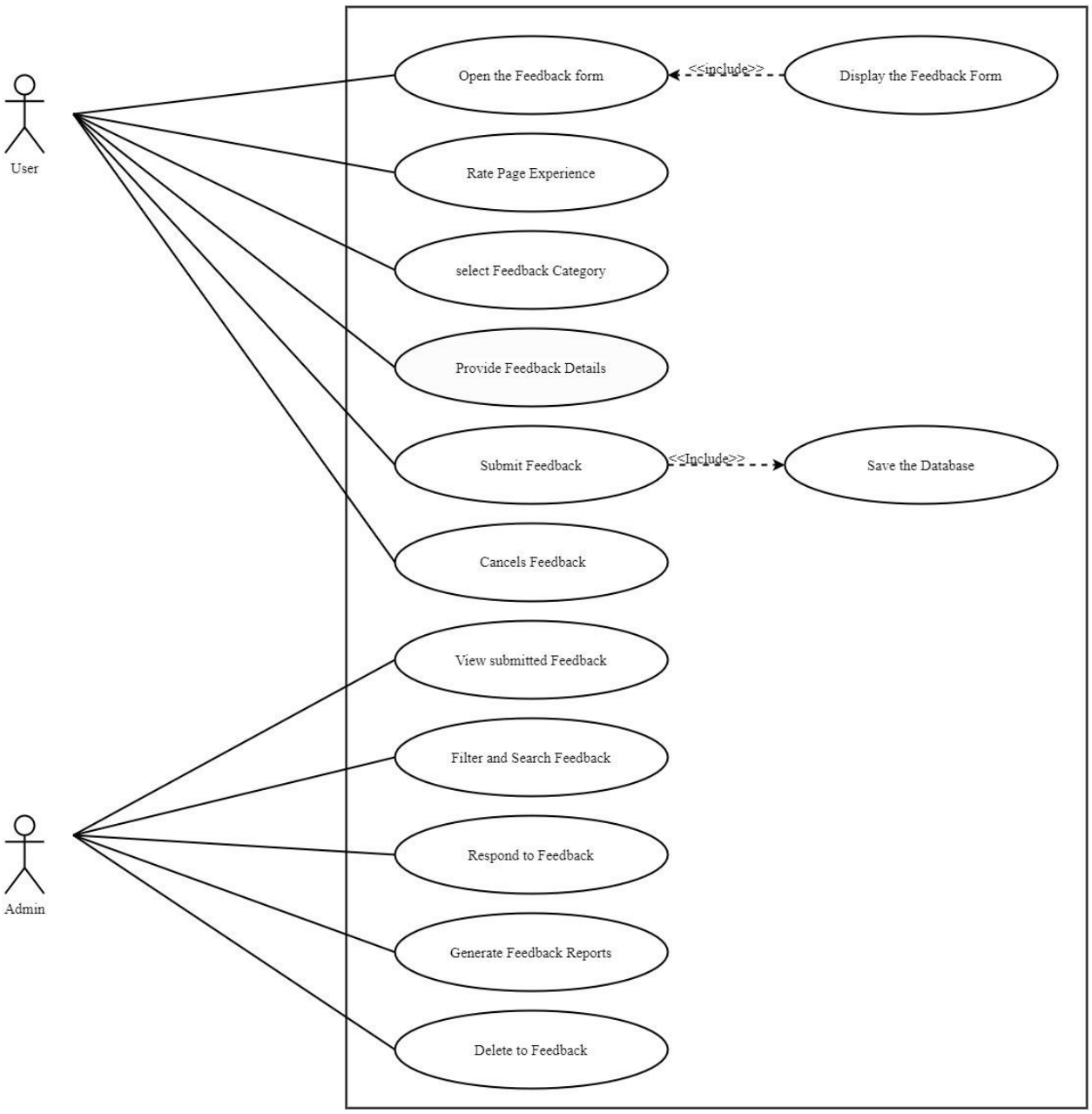

Figure 37: Use Case for Feedback

Appendix B: Test Results

Postman evidence for testing:

- Deleting medication advises sent to the parent.

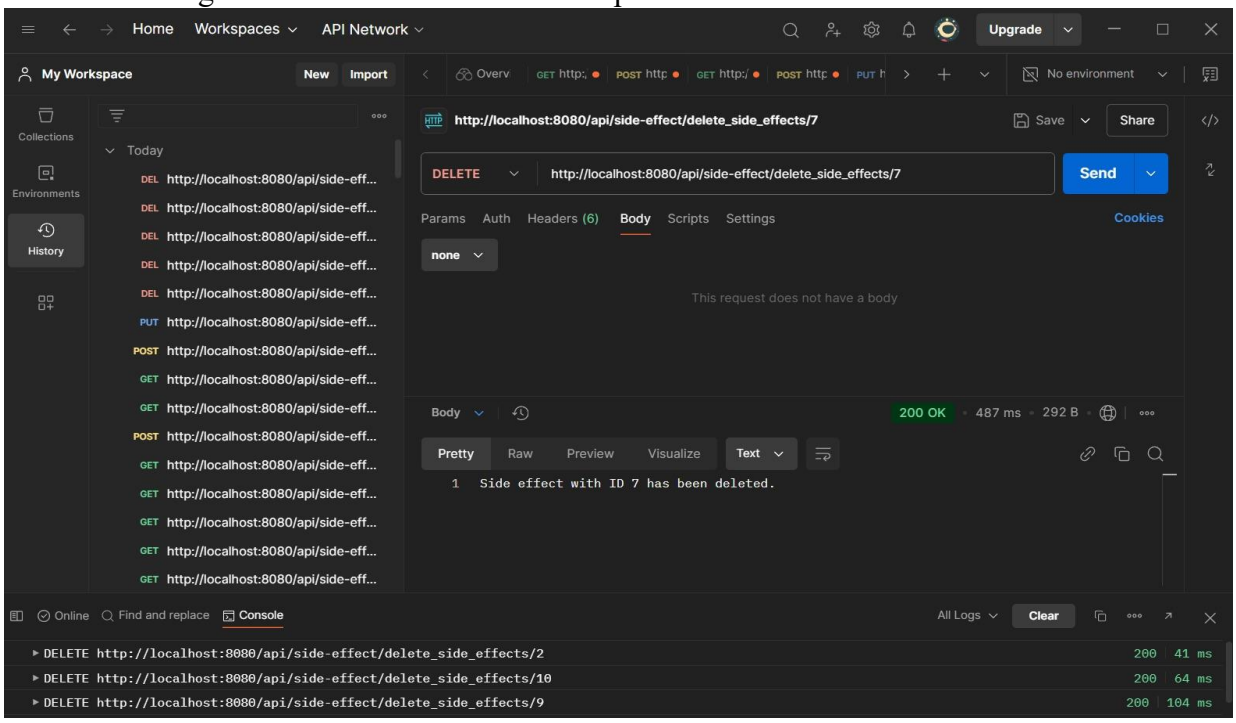

Figure 38: Postman 5

- Adding BMI values to the database.

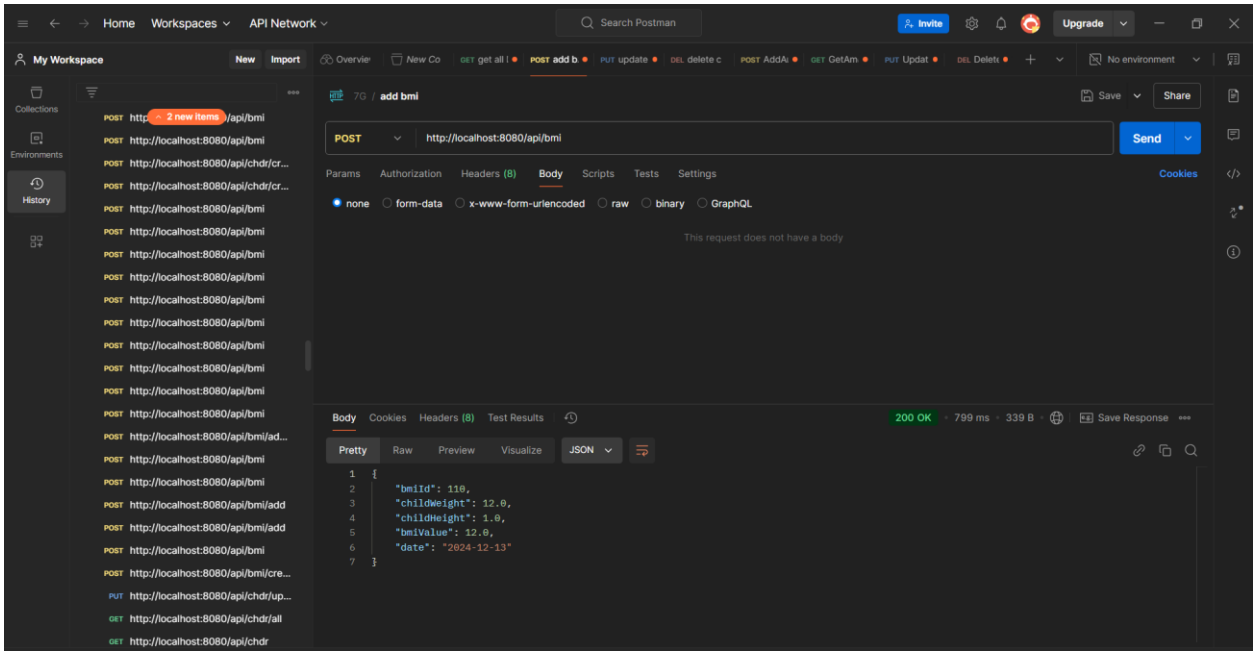

Figure 39: Postman 6

- Retrieving the BMI based advises.

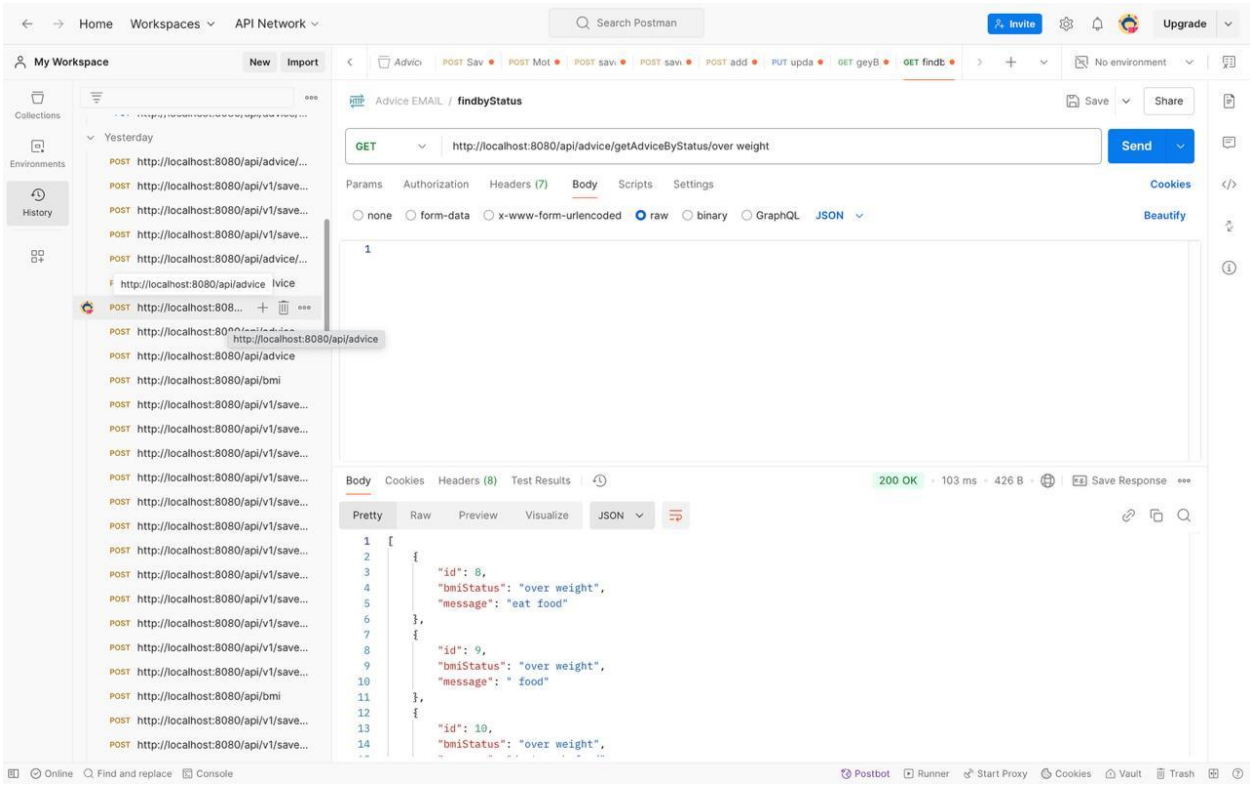

Figure 40: Postman 7

- Adding feedback.

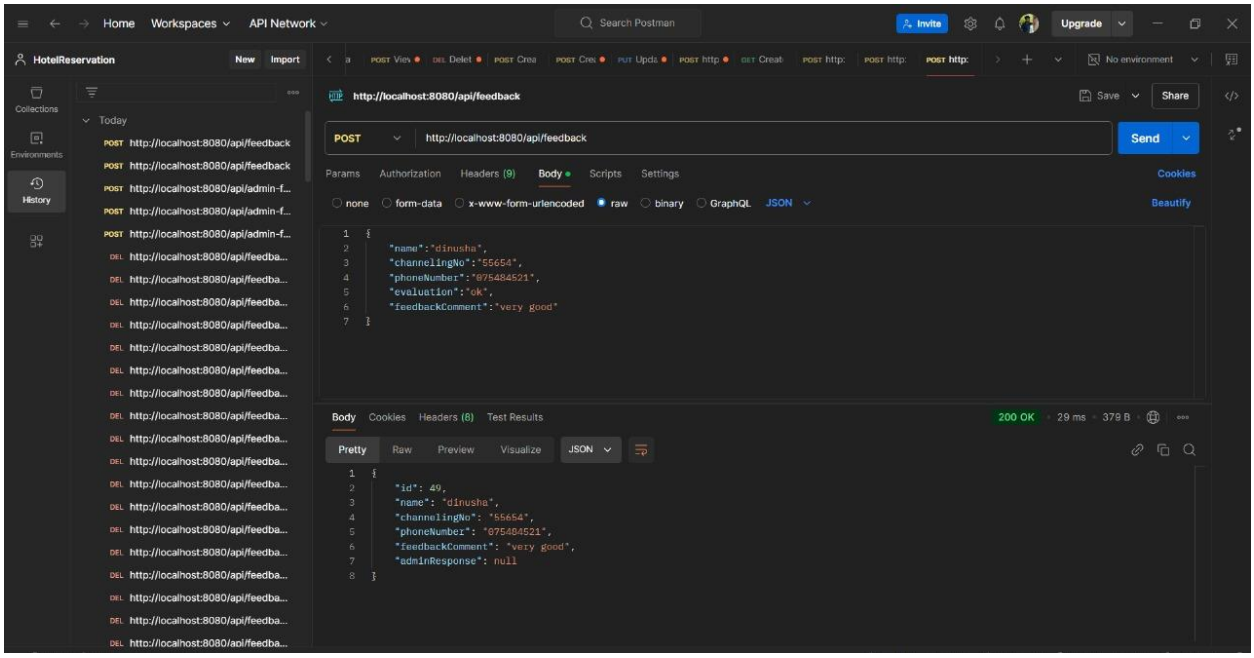

Figure 41: Postman 8

- Retrieving doctor availability details.

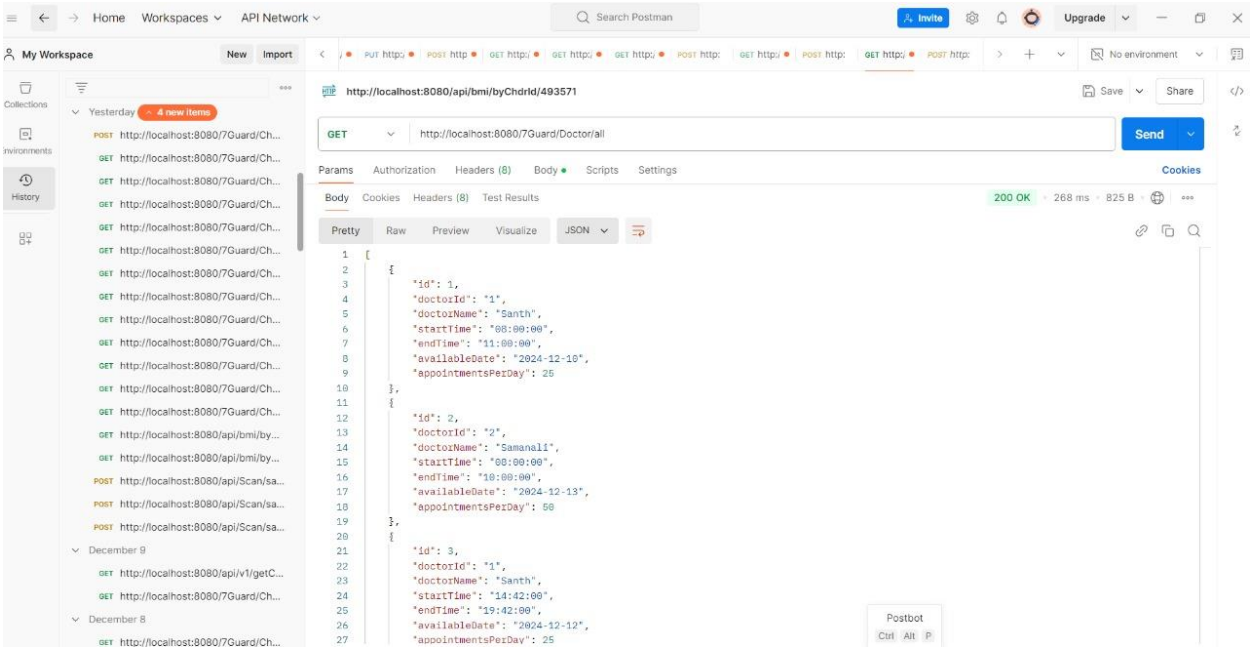

Figure 42: Postman 9

- Creating a view and saving that data into another table (Reminder).

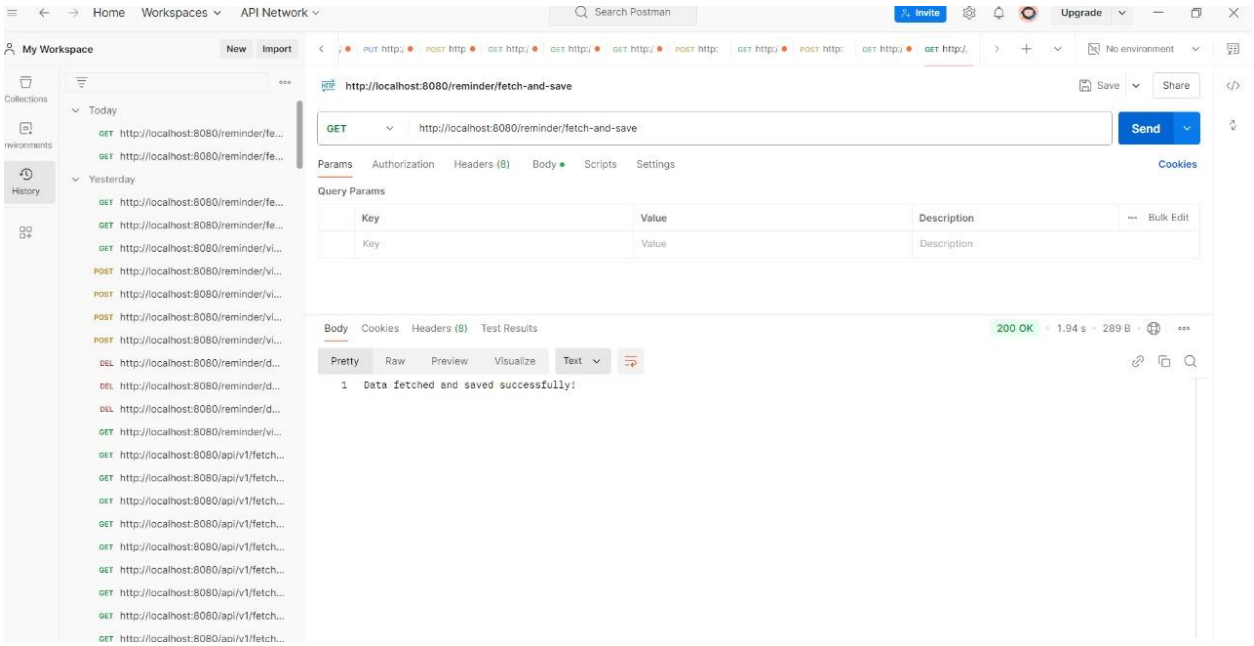

Figure 43: Postman 10