



MAPD713 ENTERPRISE TECHNOLOGIES FOR MOBILE PLATFORMS

GROUP - 5

Team Members

Srijeet Sthapit - 301365217

Abi Chitrakar - 301369773

TABLE OF CONTENTS

PROJECT DESCRIPTION.....	3
PROJECT NAME: Patient Clinical Data Management API.....	3
OVERVIEW:.....	3
OBJECTIVES.....	4
TECHNOLOGY STACK.....	5
USE CASE DIAGRAM & FUNCTIONALITIES.....	6
LOGIN.....	7
REGISTER.....	9
ADD PATIENT.....	11
LIST PATIENT.....	13
VIEW PATIENT.....	15
UPDATE PATIENT.....	17
DELETE PATIENT.....	19
ADD PATIENT RECORD.....	21
DELETE PATIENT RECORD.....	23
LIST PATIENT RECORD.....	25
VIEW PATIENT RECORD.....	27
VIEW CRITICAL PATIENTS.....	29
CHALLENGES.....	31

PROJECT DESCRIPTION

Project Name: Patient Clinical Data Management API

Overview:

Specialized RESTful service tailored for healthcare institutions.

Primary goal is to offer seamless access to pertinent patient data leading to improved patient care.

OBJECTIVES

Centralized Data Access

Provide a unified platform where all patient clinical data is stored, updated, and retrieved.

Real-time Information:

Enable immediate access to the latest patient data, ensuring healthcare providers make decisions based on the most recent information.

Critical Condition Alerts

Implement a system to quickly identify patients in critical conditions, allowing for swift medical intervention.

TECHNOLOGY STACK

Programming Language: Node.js

Web Framework: Express.js

Database: MongoDB

USE CASE DIAGRAM & FUNCTIONALITIES



Figure 1: UseCase diagram of Patient Clinical Data Management System

LOGIN

Use Case	Login
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider must be able to log in to the system using the ID and password provided during registration. The login is verified and in case of wrong login details, an error message is displayed.
Steps	<ol style="list-style-type: none">1. The actor sends a POST request to /login with login credentials in the request body.2. The system validates the credentials.3. The system responds with a successful login message.

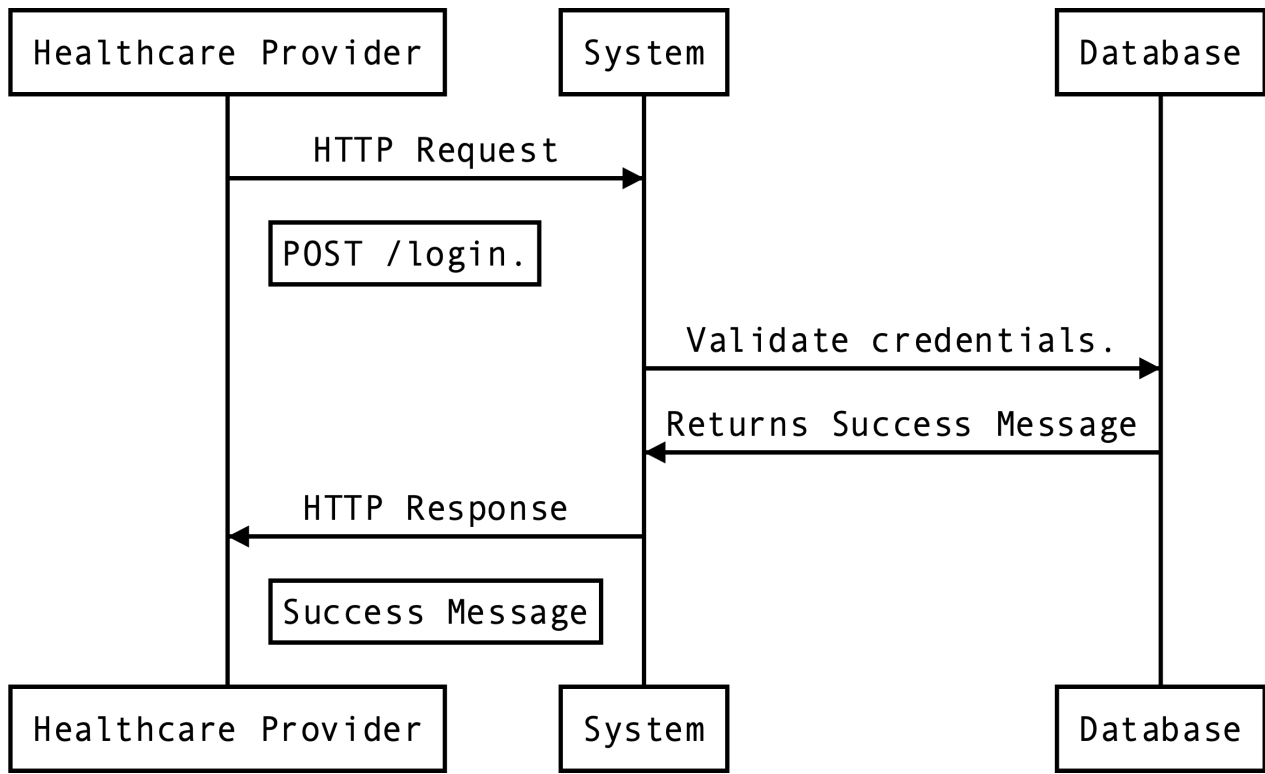


Figure 2: Sequence Diagram of Login functionality

REGISTER

Use Case	Register
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider fills in his/her details in the form to register into the system.
Steps	<ol style="list-style-type: none">1. The actor sends a POST request to /register with details in the request body.2. The system validates the credentials.3. The system responds with a successful registration message.

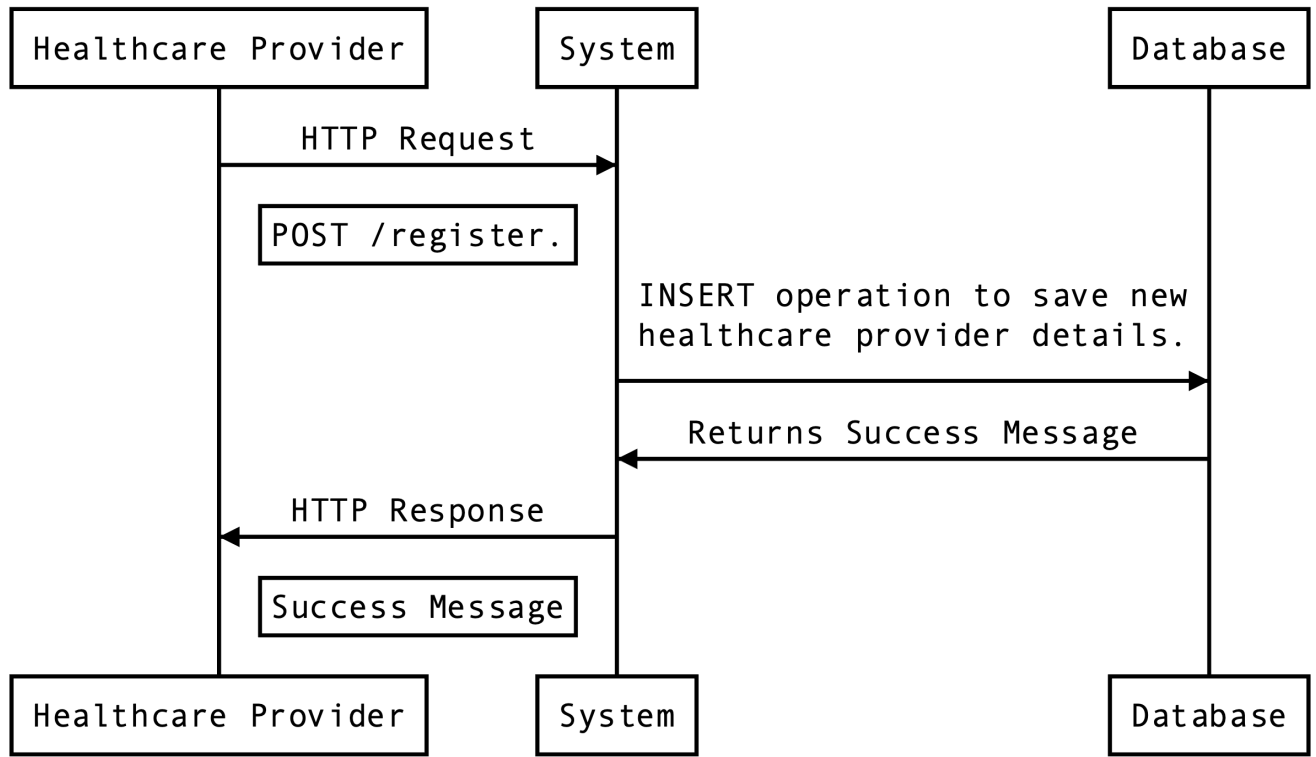


Figure 3: Sequence Diagram of Register functionality

ADD PATIENT

Use Case	Add patient
Actor	Healthcare Provider (Doctor/Nurse)
Description	This use case facilitates the addition of new patients into the system.
Steps	<ol style="list-style-type: none">1. The actor sends a POST request to /patients with patient details in the request body.2. The system saves the patient's information in the database.3. The system responds with a confirmation message.

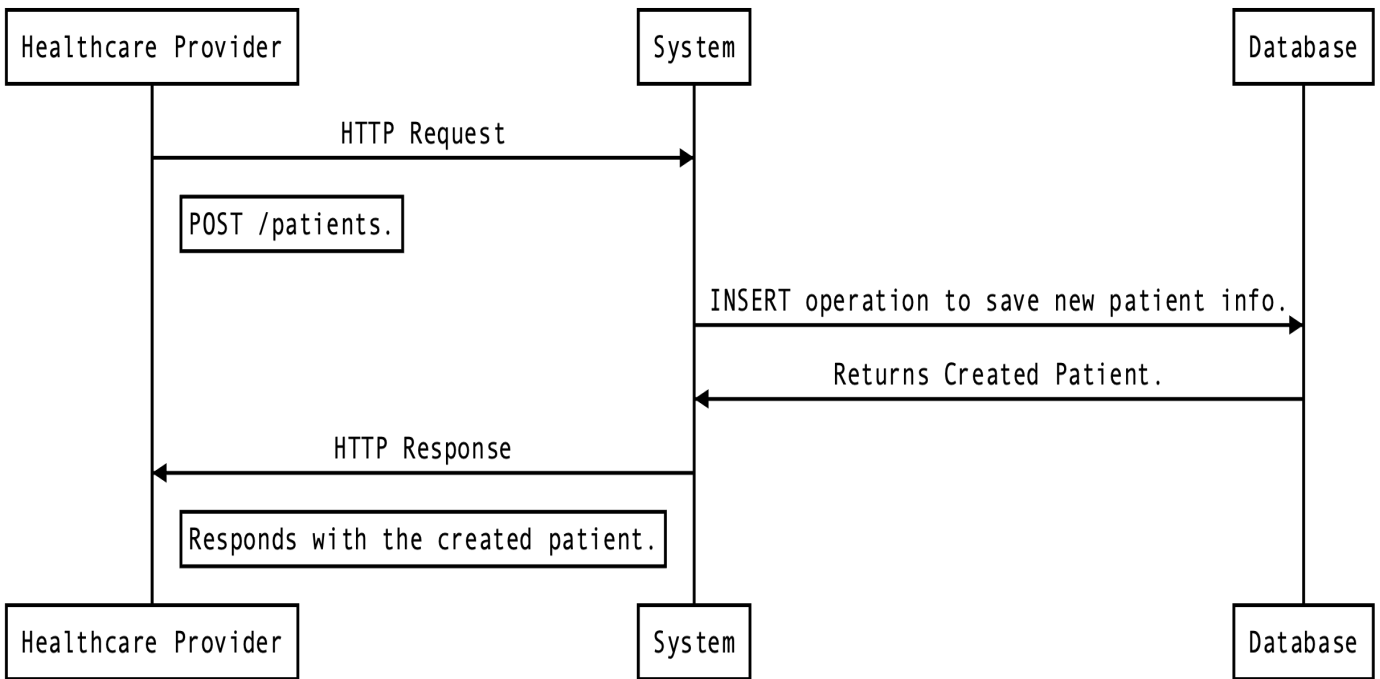


Figure 4: Sequence Diagram of Add Patient functionality

LIST PATIENT

Use Case	List patient
Actor	Healthcare Provider (Doctor/Nurse)
Description	This use case lists all patients registered in the system.
Steps	<ol style="list-style-type: none">1. The actor sends a GET request to /patients.2. The system retrieves the list of patients from the database.3. The system responds with the list of patients.

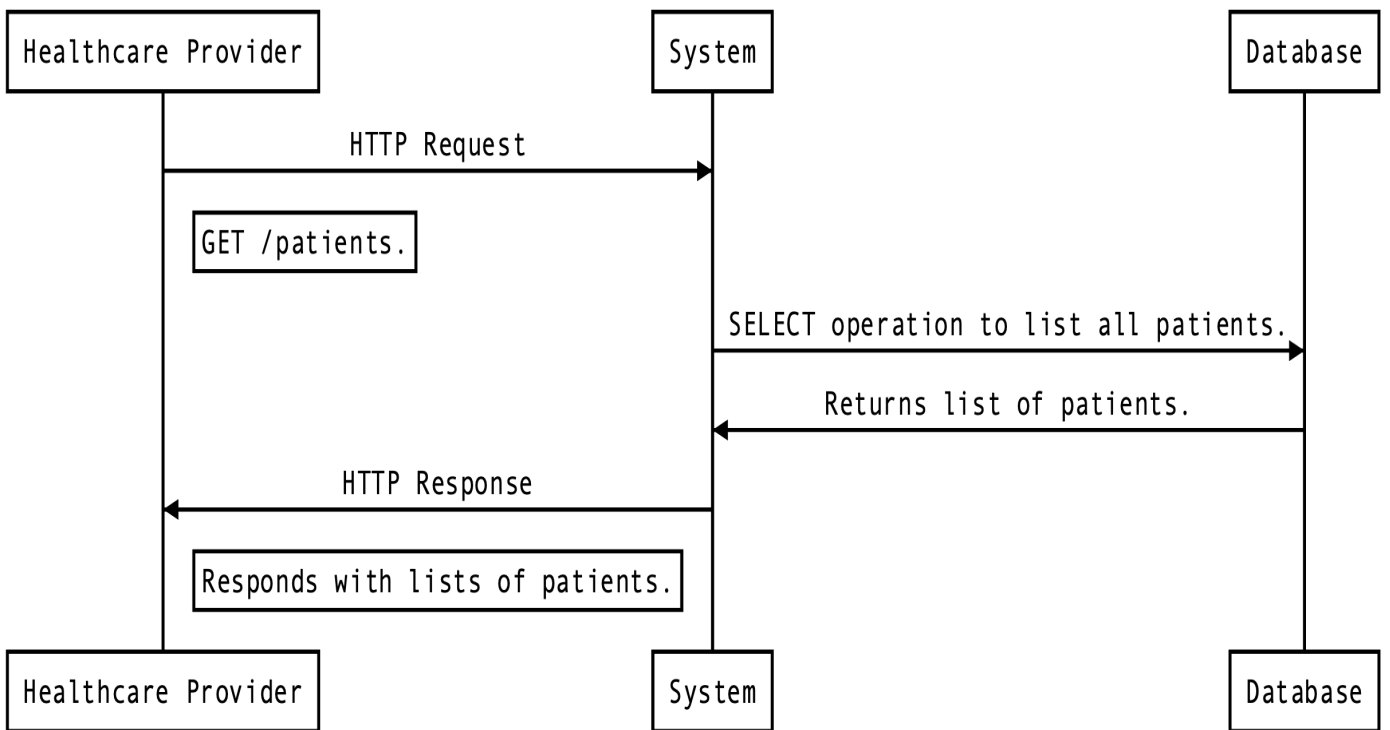


Figure 5: Sequence Diagram of List Patients functionality

VIEW PATIENT

Use Case	View Patient
Actor	Healthcare Provider (Doctor/Nurse)
Description	This use case facilitates retrieving and viewing detailed specifications of a specific patient.
Steps	<ol style="list-style-type: none">1. The actor sends a GET request to /patients/{patientId}.2. The system retrieves the patient's information from the database.3. The system responds with the requested patient's details.

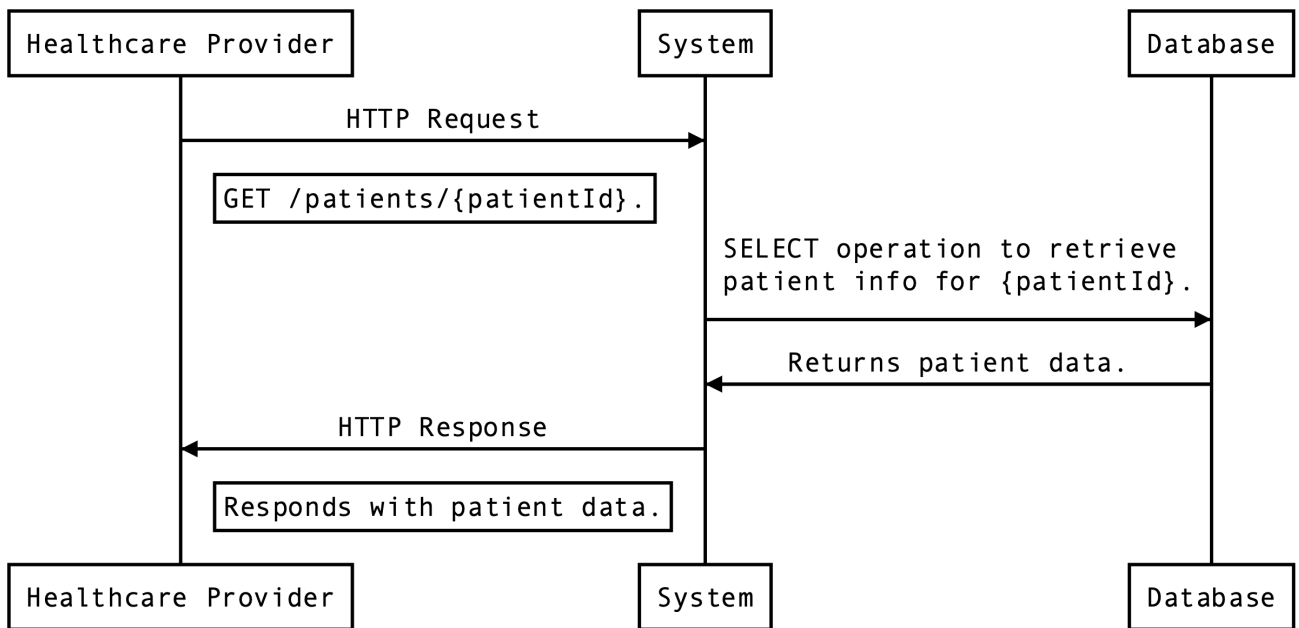


Figure 6: Sequence Diagram of View Patient functionality

UPDATE PATIENT

Use Case	Update Patient
Actor	Healthcare Provider (Doctor/Nurse)
Description	This use case facilitates updating details of a patient.
Steps	<ol style="list-style-type: none">1. The actor sends a PUT request to /patients/{patientId} with the updated data in the request body.2. The system updates the patient's information in the database.3. The system responds with a confirmation message.

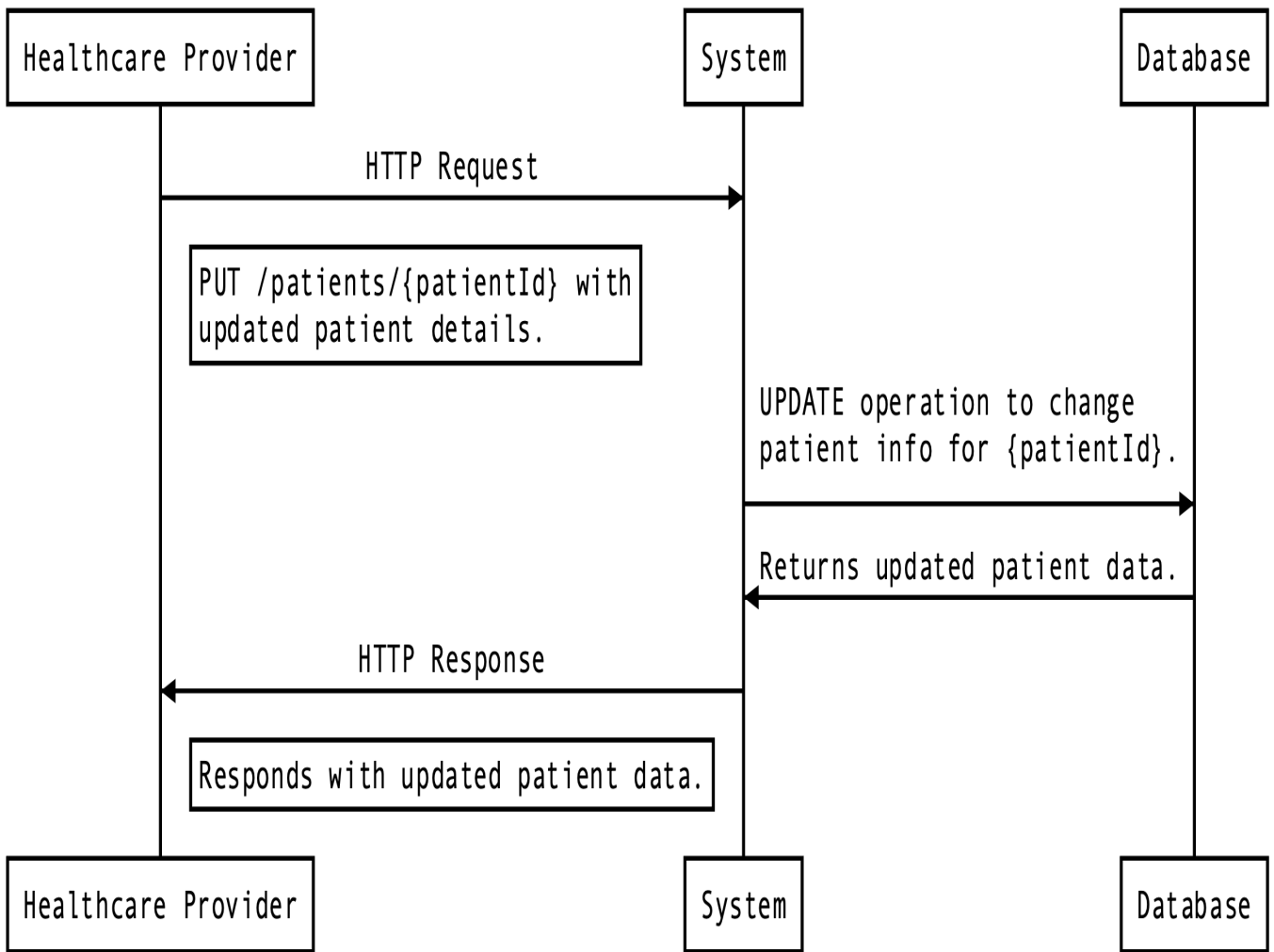


Figure 7: Sequence Diagram of Update Patient functionality

DELETE PATIENT

Use Case	Delete Patient
Actor	Healthcare Provider (Doctor/Nurse)
Description	This use case facilitates removal of patients along with records.
Steps	<ol style="list-style-type: none">1. The actor sends a DELETE request to /patients/{patientId}.2. The system deletes the patient's details and records in the database.3. The system responds with a successful deletion message.

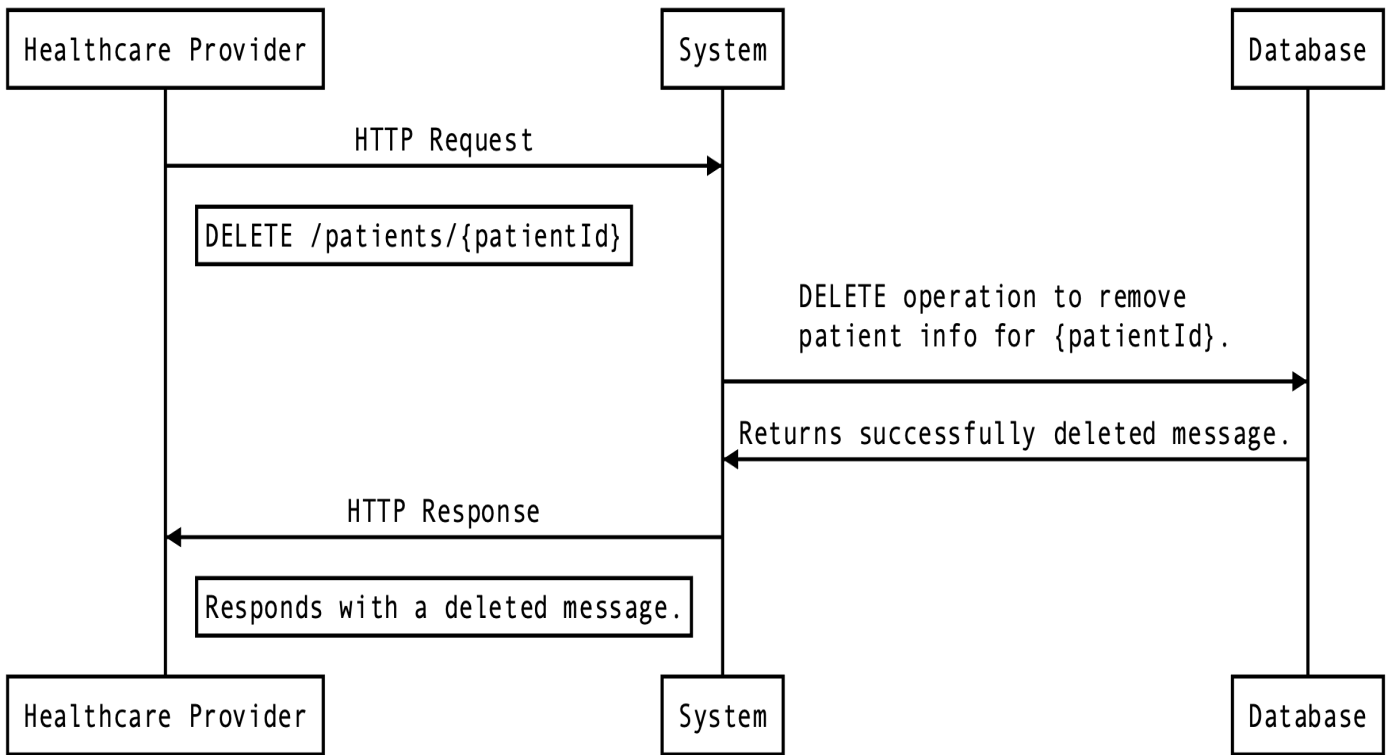


Figure 8: Sequence Diagram of Delete Patient functionality

ADD PATIENT RECORD

Use Case	Add Patient Record
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider selects a patient and enters the medical record information, including Blood Pressure (X/Y mmHg), Respiratory Rate (X/min), Blood Oxygen Level (X%) & Heartbeat Rate (X bpm/min).
Steps	<ol style="list-style-type: none">1. The actor sends a POST request to /patients/{patientId}/records with patient details in the request body.2. The system saves the patient's health record in the database.3. The system responds with a confirmation message.

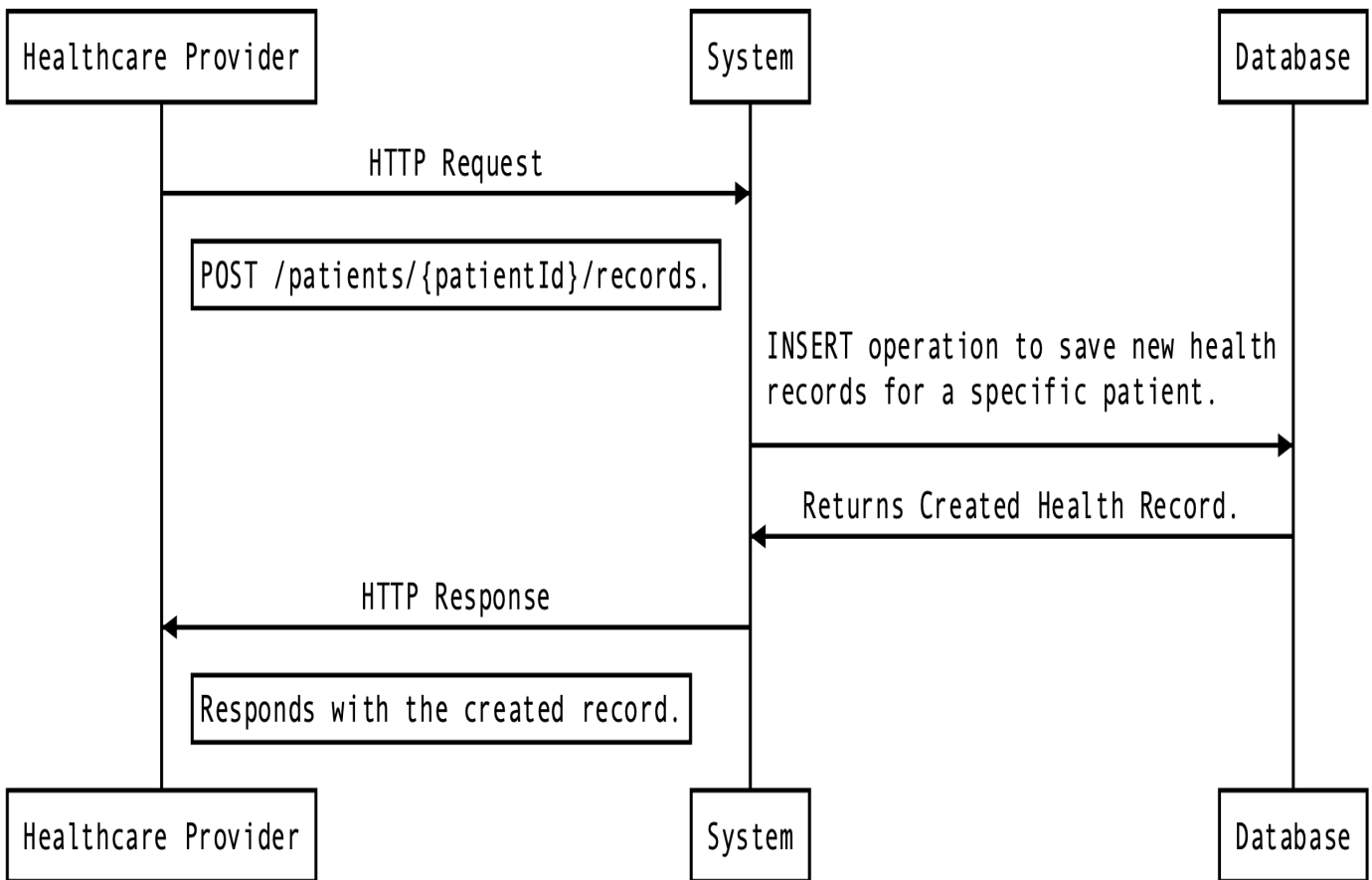


Figure 9: Sequence Diagram of Add Patient functionality

DELETE PATIENT RECORD

Use Case	Delete Patient Record
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider deletes a record of a patient in case it is not necessary.
Steps	<ol style="list-style-type: none">1. The actor sends a DELETE request to /patients/{patientId}/records/{recordId}.2. The system deletes the patient's details and records in the database.3. The system responds with a successful deletion message.

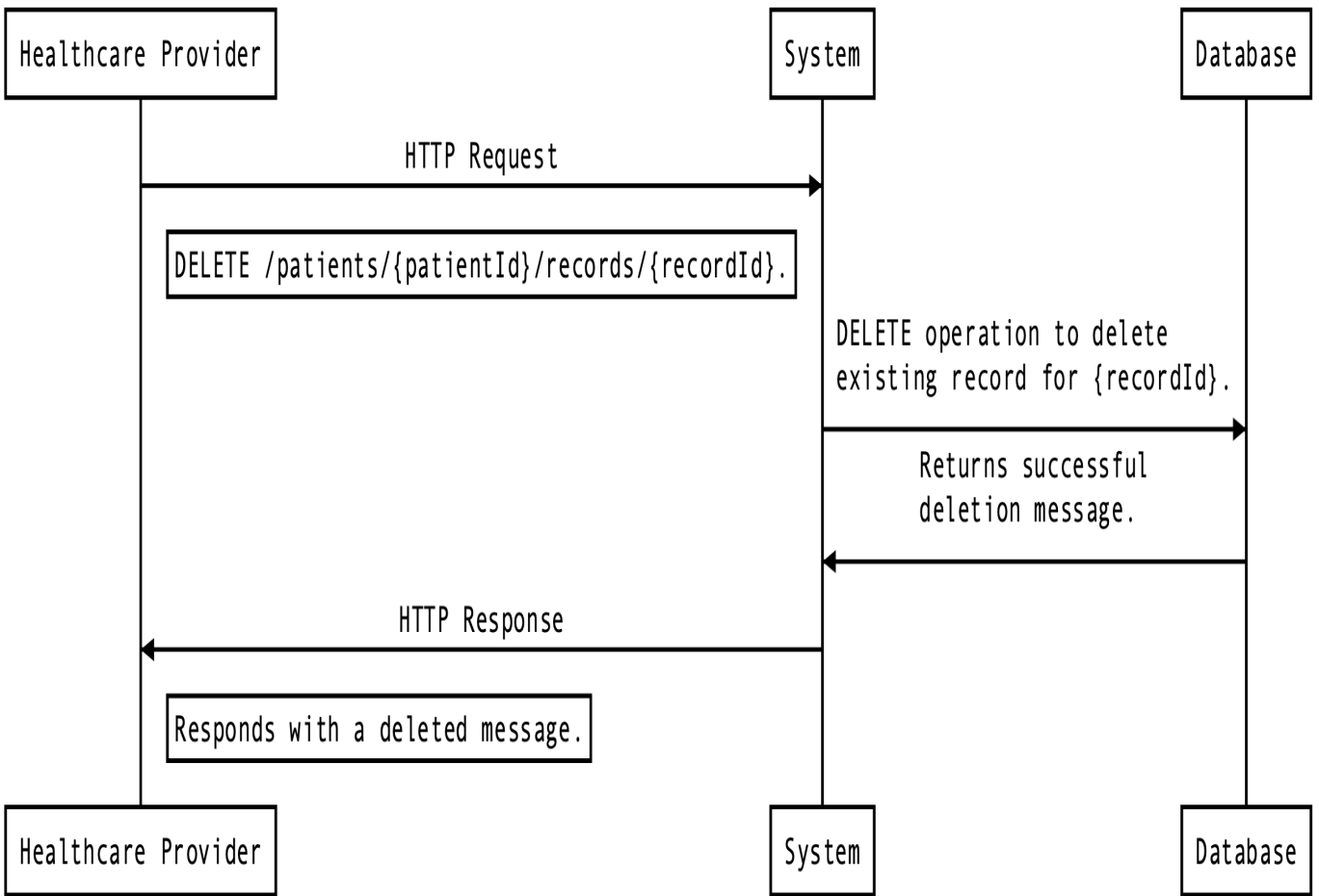


Figure 10: Sequence Diagram of Delete Patient Record functionality

LIST PATIENT RECORD

Use Case	List Patient Record
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider lists all the health record of a single patient.
Steps	<ol style="list-style-type: none">1. The actor sends a GET request to /patients/{patientId}/records.2. The system retrieves the list of patients from the database.3. The system responds with the list of patients.

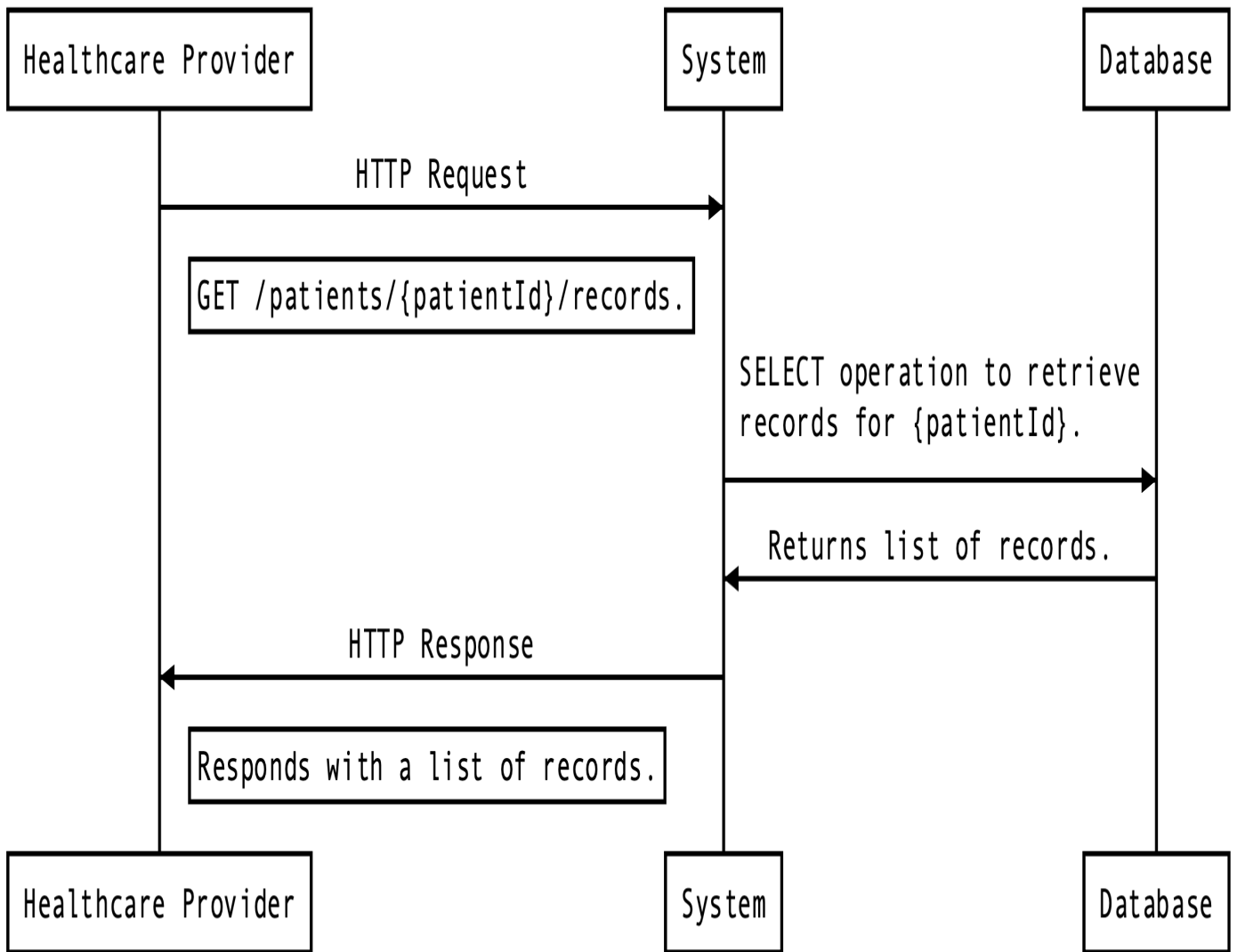


Figure 11: Sequence Diagram of List Patient Record functionality

VIEW PATIENT RECORD

Use Case	View Patient Record
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider views the details of a health record related to a single patient.
Steps	<ol style="list-style-type: none">1. The actor sends a GET request to /patients/{patientId}/records/{recordId}.2. The system retrieves the patient's information from the database.3. The system responds with the requested patient's details.

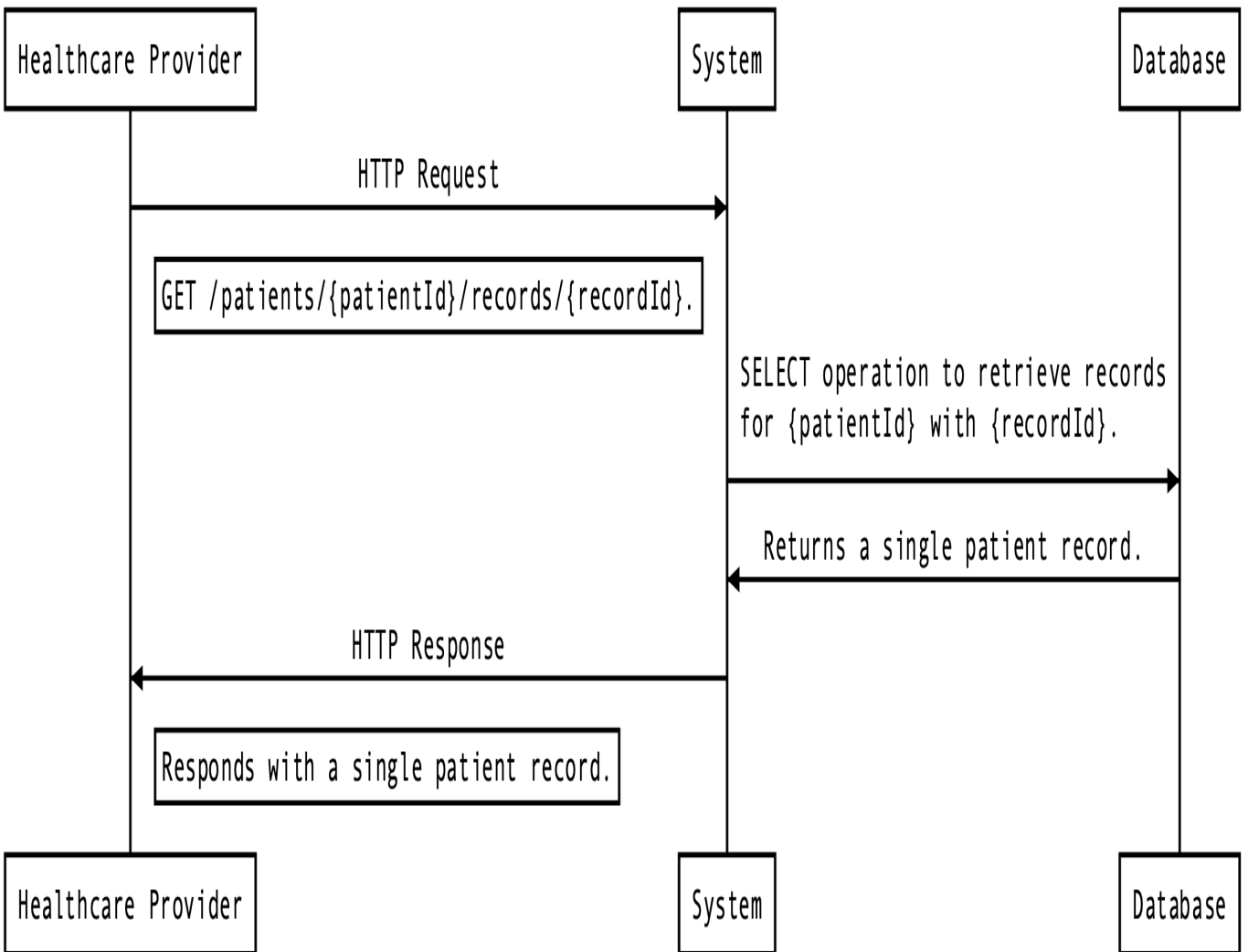


Figure 12: Sequence Diagram of View Patient functionality

View Critical Patients

Use Case	View Critical Patients
Actor	Healthcare Provider (Doctor/Nurse)
Description	A healthcare provider can view a list of critically ill patients.
Steps	<ol style="list-style-type: none">1. The actor sends a GET request to /patients/critical.2. The system retrieves the patient's information from the database.3. The system responds with the requested patient's details.

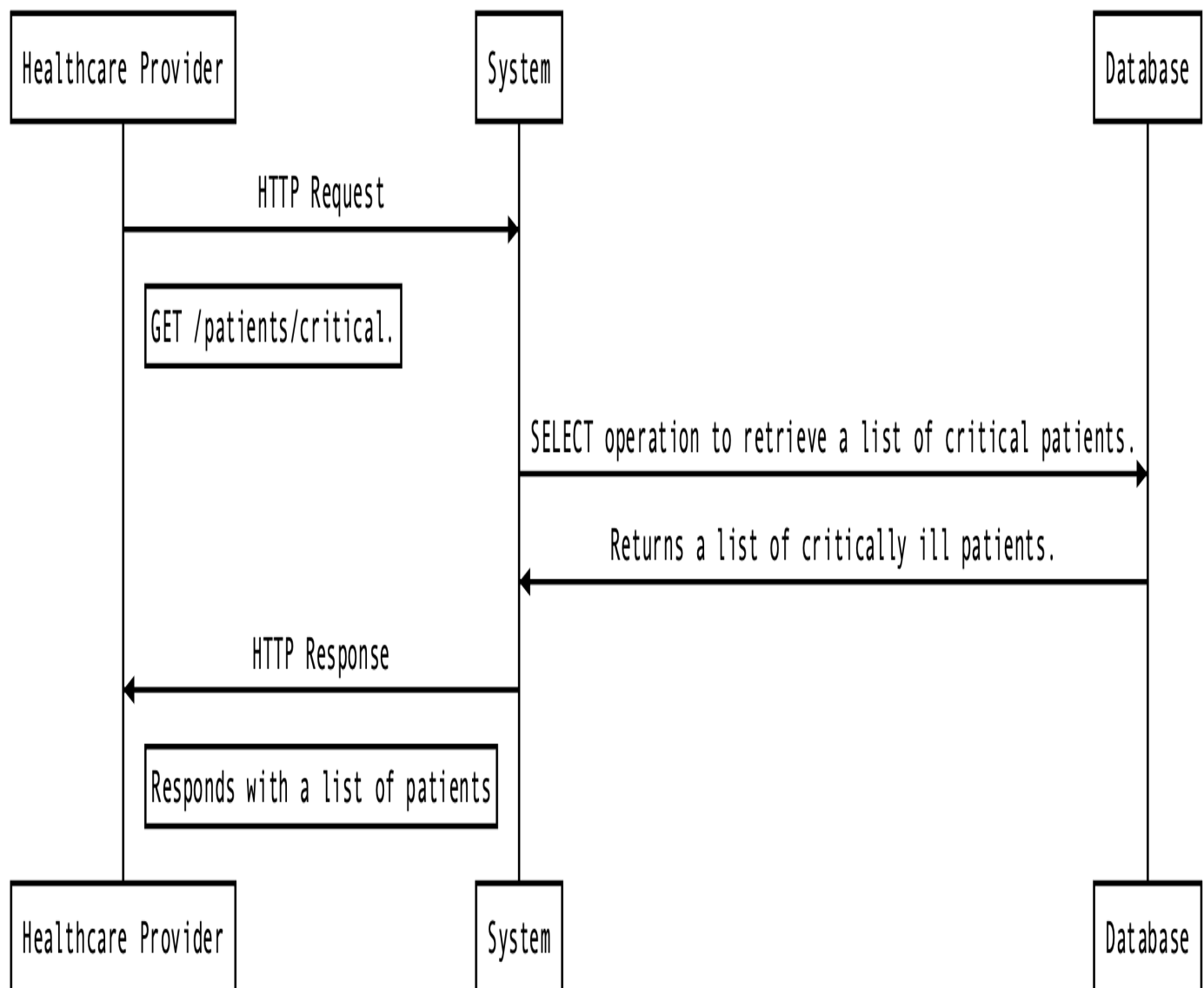


Figure 13: Sequence Diagram of View Critical Patients functionality

CHALLENGES

Optimizing database queries and ensuring efficient data retrieval, especially for functionalities like viewing patient records and critical patient alerts.

Implementing robust error-handling mechanisms, especially during data input and API interactions.

Validating user inputs to prevent data inconsistencies and ensuring data integrity.

THANK YOU 😊

ANY QUERIES ?