Q1- Doctor-Referral SpringBoot Application

Objective & Scope:

The project aimed to create a user-friendly web application that would help patients find suitable doctors based on their symptoms and location. The main objectives were to allow patients to input their symptoms and receive recommendations for relevant doctors. The system also needed to provide built-in validations to ensure data accuracy. Additionally, if no specialists were available locally, the system would verify so, thereby improving healthcare access and outcomes for patients. The scope of my work was primarily on the backend side, where I was responsible for designing the data models, implementing business logic, handling database interactions, and creating APIs for the frontend to consume

Technology Used: The project used the following technologies and tools-

- Java –The primary language for backend development.
- SpringBoot Framework - to create stand-alone bootstrap app.
- MySQL database: To store and manage data, which is crucial for this healthcare-related application.
- Postman: Used for testing and debugging APIs.

Challenges & Impact:

- Data Modelling: Designing the database schema and establishing relationships between various entities (Doctors, Patients, Specialities, and Symptoms) presented a significant challenge. The accuracy of the data model was critical to providing relevant doctor recommendations to patients.

- Complex Business Logic: Implementing the logic to match symptoms to doctors and handling cases where no local specialists were available was complex. This required a thoughtful algorithm and efficient database queries to ensure optimal performance.

- Validation: Ensuring the input data from patients was valid and secure was a constant concern. I had to implement both annotation-based validation and custom validation checks to maintain data integrity.

- Error Handling: Building a comprehensive GlobalExceptionHandler to handle various exceptions, including custom ones like AlreadyExistsException, NotFoundException, and NotValid, was vital to provide informative error responses to the frontend.

Outcomes:

The end result was a functional web application that met its objectives. Patients could input their symptoms and receive relevant doctor recommendations. The system handled validation effectively and suggested doctors. The project provided valuable insights into data modelling, complex business logic, and error handling in a real-world application. It also highlighted the importance of validation, security, and performance in healthcare-related systems. Overall, this project significantly challenged me and allowed me to develop valuable skills in backend development, database management, and healthcare domain knowledge.