⍰ Project Documentation: Hospital Management System

⍰ Objective

To create a simple Java-based Hospital management system that allows adding patients, viewing patients, viewing doctor, book Appointment .

⍰ Technologies Used

- Language: Java
- IDE: Eclipse
- Database: MySQL
- JDBC: For database connectivity
- Version Control: Git and GitHUB

⍰ Project Structure

```
JDBCDEMO/
├── src /
│   ├── Patient.java
│   ├── Doctor.java
│   └── Main.java
├── README.md
└── Documentation/
    └── ProjectDocumentation.docx
```

⍰ Modules/Features

1. Add Patients
2. View Patients
3. View Doctors
4. Book Appointment

⍰ User Roles

- Admin (add patients/view patients/add doctors)
- Member (book Appointment — limited actions)

⍰ Class Design

1. Patient Class
   - Attributes: id, name, age, gender
   - Methods: addPatient(), viewPatients(), getPatientByID()

2. Doctor Class

- Attributes: Id, name, specilization
- Methods: viewDoctors(), getDoctorByID()

3. Main Class
- Attributes: Patient ID, Doctor ID,Appointment date
- Methods: bookAppointment(), checkDoctorAvailability()

Sample Test Cases

Feature	Input	Expected Output
Add Patients	Enter name: sai, age:22, gender: male	Patient added successfully
View Patients	-	List of Patients
View Doctors	-	List of Doctors
Book Appointment	Doctor ID: 101, Patient ID: 1 Appointment date:2025-11-13	Book Appointment successfully

How to Run

1. create a "hospital" database with relevant tables.
2. Update database credentials in "HospitalManagementSystem.java".
3. Open project in your IDE(Eclipse).
4. Compile and run "HospitalManagementSystem.java".
5. Use the console menu to operate the system.

Future Enhancements

- Integration with MySQL database
- User authentication system
- Book Appointment feature
- Report generation

Author

- Name: KUNDEM PRANAV VENKAT
- Contact: pranavvenkat2005@gmail.com

-GitHub: https: https://github.com/pranav1-rock/Hospital-Management-System

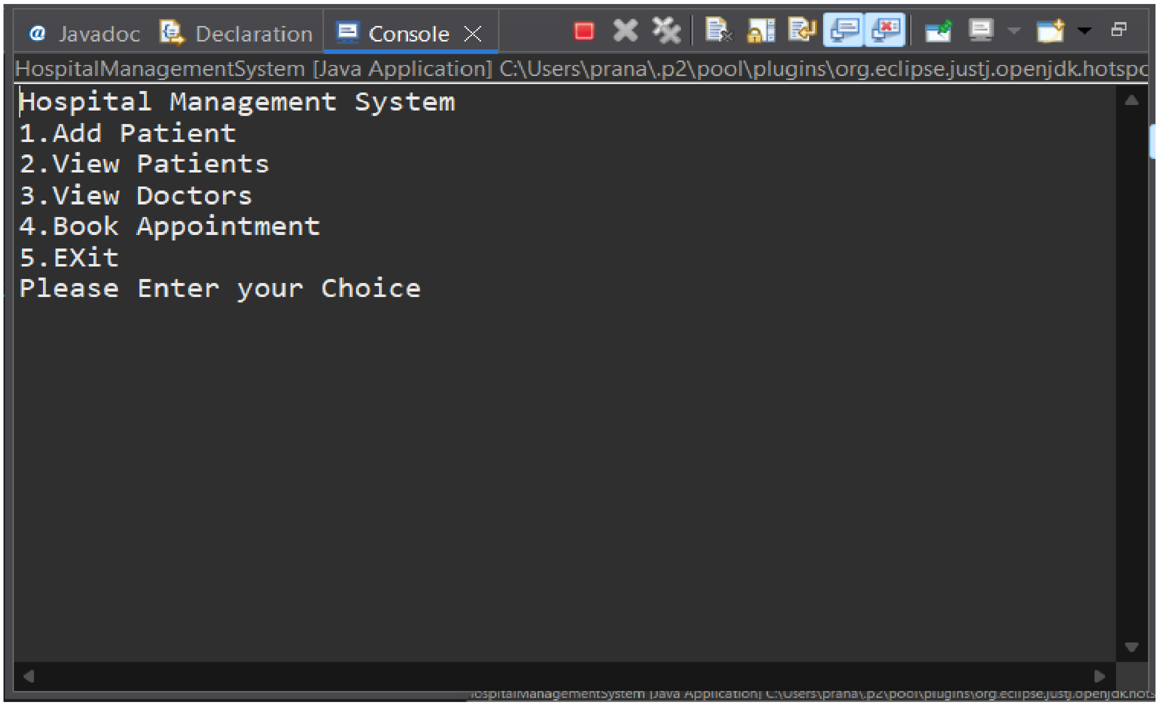
System Architecture Diagram

Below is a simple architecture diagram showing how different components interact:

```
[User Interface]
    ↓
[Hospital.java]
    ↓
[Patient.java] ↔ [Doctor.java]
    ↓
[Database (MySQL)]
```

Sample Screenshots

Screenshots showing the menu-driven interface in console output:



HospitalManagementSystem [Java Application] C:\Users\prana\.p2\pool\plugins\org.eclipse.justj.openjdk.hotspo

```
Hospital Management System
1.Add Patient
2.View Patients
3.View Doctors
4.Book Appointment
5.EXit
Please Enter your Choice
```

```
========= Hospital  Management System =========
1. Add Patients
2. View Patients
3. View Doctors
4. Book Appointment
5. Exit
 Please Enter your choice:
```

The system takes user input and performs operations based on the selected menu item.