Hospital Management System

About the Document.

This document provides a complete technical overview of the Hospital Management System (HMS). It includes system workflows, microservices design, service-to-service communication using OpenFeign, event-driven messaging using Apache Kafka, and the role of API Gateway in managing requests. The documentation also contains a flowchart to illustrate the overall system architecture.

Table of Contents

1. Problem Statement

Hospitals require an integrated digital system to manage patients, doctors, appointments, billing, notifications, and medical records. The HMS must ensure efficiency, security, and scalability while handling large-scale concurrent operations.

2. System Workflows

1. Patient Service (Standalone)
- Handles patient registration, authentication, and profile management.
- Acts as the entry point for patient-related activities.

2. Appointment Service
- Manages doctor appointments and scheduling.
- Interacts with Patient Service and Doctor Service.

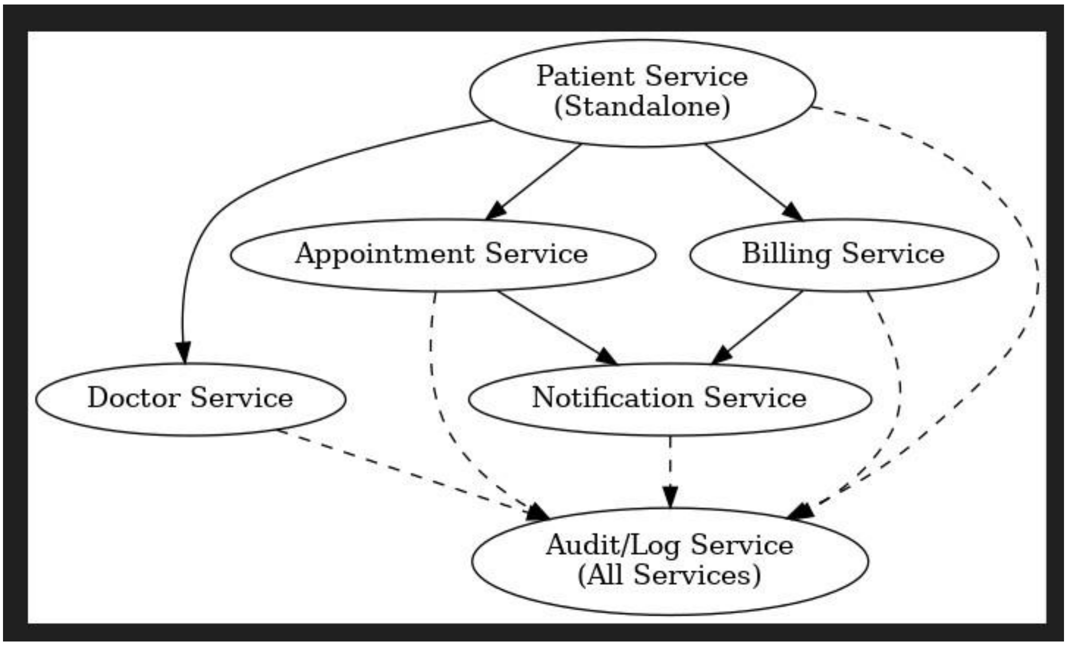
3. Doctor Service
- Provides doctor information, schedules, and medical history tracking.
- Linked to Appointment Service and Patient Service.

4. Billing Service
- Handles billing, payments, and insurance claims.
- Connects with Patient Service and Notification Service.

5. Notification Service
- Sends notifications (SMS/Email) for appointments, billing, and updates.
- Integrated with Appointment Service and Billing Service.

6. Audit/Log Service (All Services)
- Tracks all service activities for auditing and compliance.
- Works across all services to ensure accountability.

3. Workflow Flowchart



4. OpenFeign - Service-to-Service Communication

OpenFeign - Service-to-Service Communication

OpenFeign is a declarative REST client for Java, integrated with Spring Boot, that allows services to communicate with each other without writing boilerplate REST template code.

Usage in Hospital Management System:
- Patient Service calls Appointment Service for booking.
- Doctor Service fetches patient details from Patient Service.
- Billing Service communicates with Patient and Notification Services.

Example:

```
@FeignClient(name = "appointment-service")
public interface AppointmentClient {
@GetMapping("/appointments/{id}")
  Appointment getAppointment(@PathVariable Long id);
}
```

Benefits:
- Declarative REST calls.
- Easier integration with Ribbon/Eureka for load balancing.
- Cleaner code and reduced boilerplate.

5. Apache Kafka - Event Driven Messaging

Apache Kafka - Event Driven Messaging

Apache Kafka is a distributed event streaming platform used for building real-time pipelines and applications. In the Hospital Management System, Kafka enables asynchronous communication between services.

Usage in Hospital Management System:
- Appointment events published to Kafka → consumed by Notification Service.
- Billing events published → consumed by Audit/Logs Service.
- Notification topics ensure real-time SMS/Email alerts.

Example Kafka Topics:
- appointment-events
- billing-events
- notification-events

Benefits:
- High throughput and fault-tolerant messaging.
- Decoupled services with asynchronous updates.
- Real-time processing of appointments, billing, and notifications.

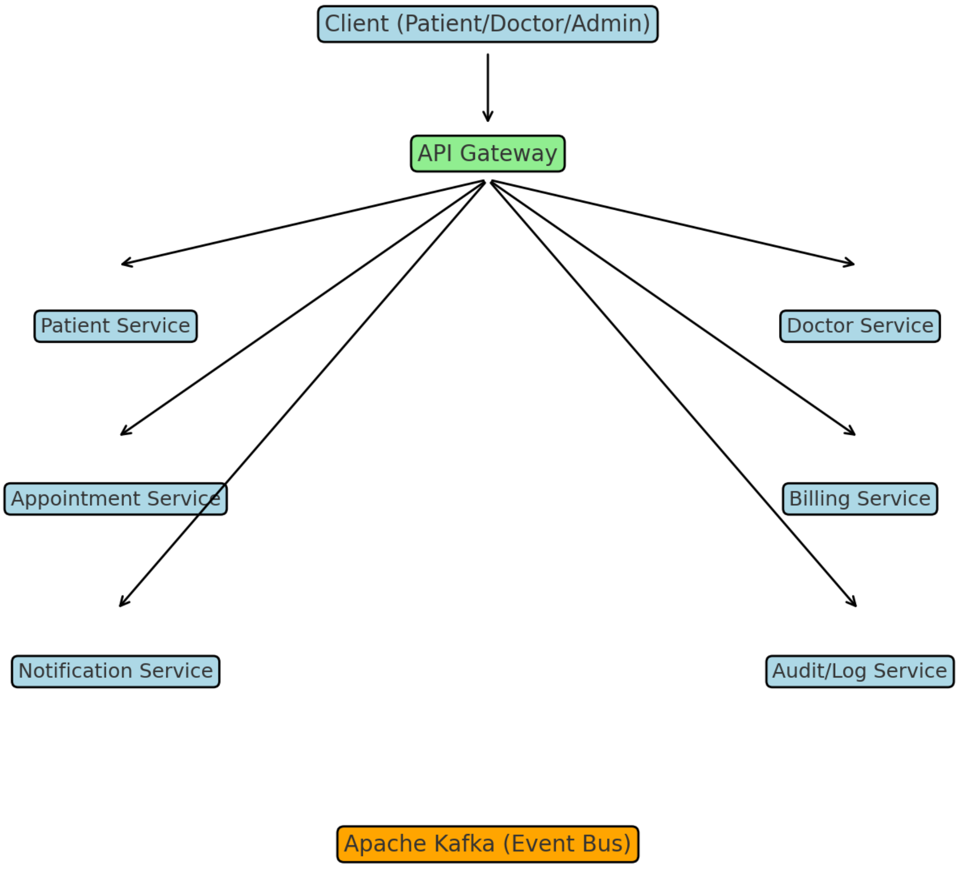
6. API Gateway Integration

The API Gateway serves as the single entry point for clients (patients, doctors, admins) to interact with the HMS microservices. Instead of calling services directly, all requests pass through the gateway.

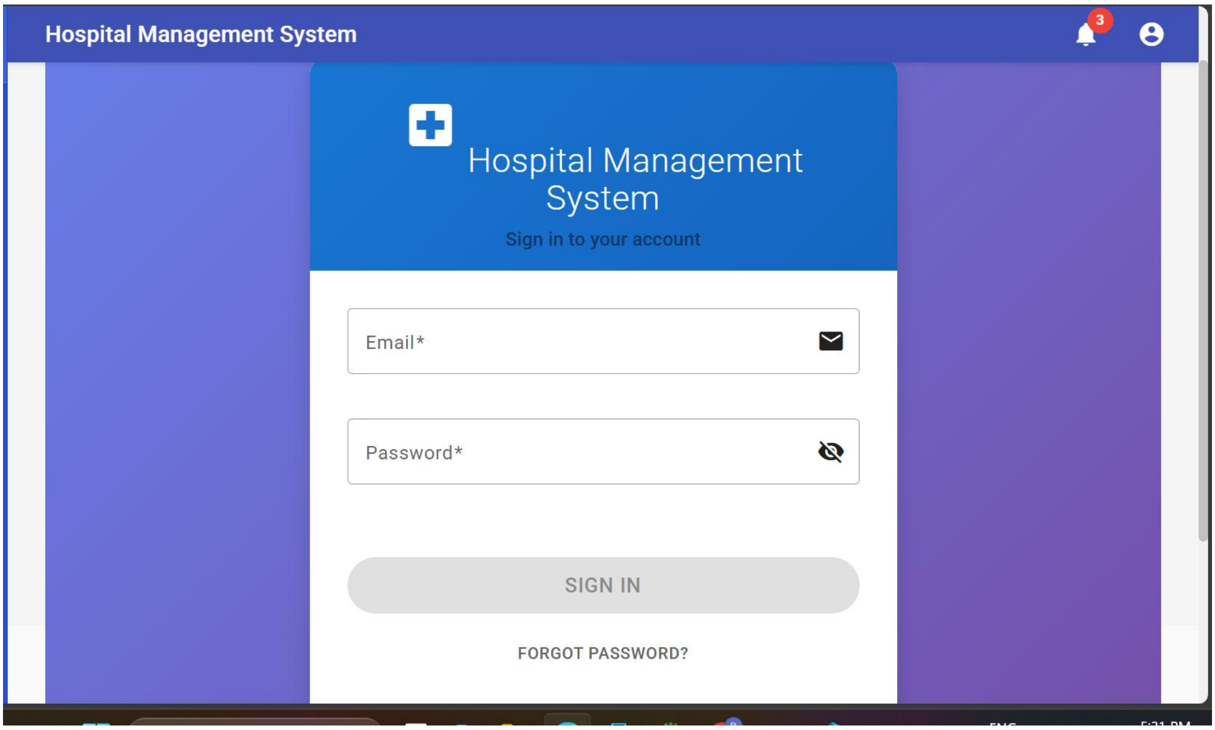
Responsibilities of API Gateway:
- Routing requests to appropriate microservices (Patient, Doctor, Appointment, Billing, Notification).
- Authentication & Authorization using JWT/OAuth2.
- Load balancing and failover.
- Request/response transformation.
- Logging and monitoring.

Flow:
Client → API Gateway → Service Routing → Microservices (Patient, Doctor, Appointment, Billing, Notification, Audit).

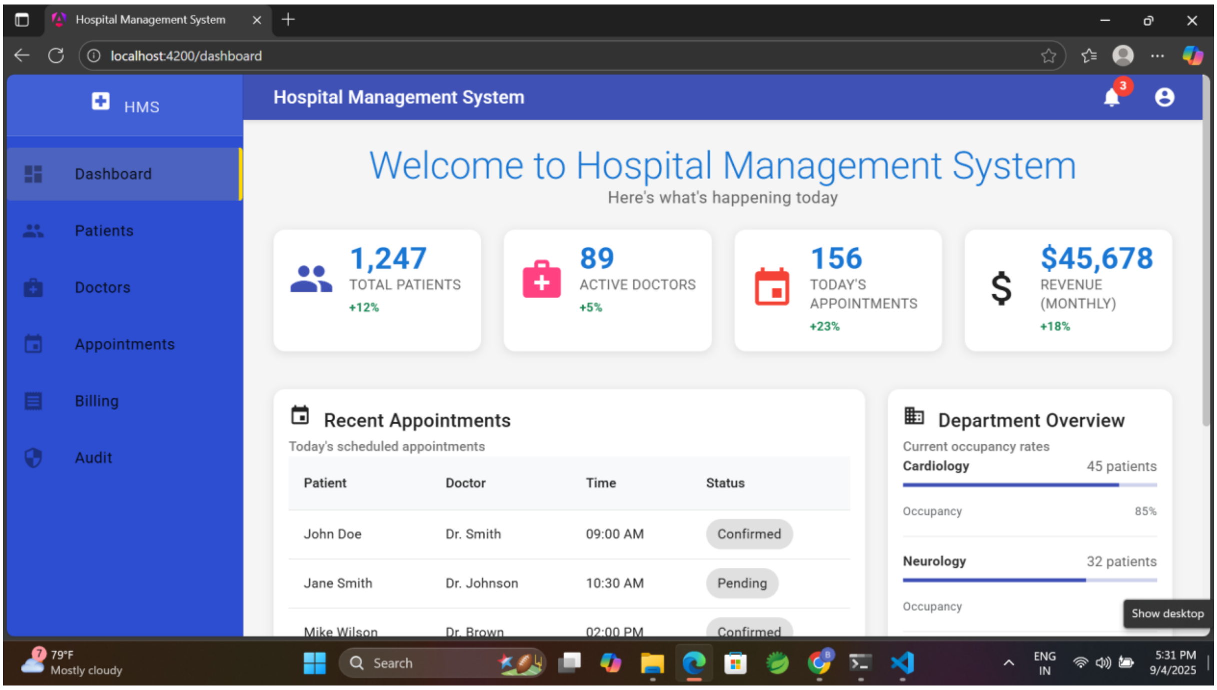
API Gateway Architecture Flowchart



Sign In page



Dashboard



Welcome to Hospital Management System

Here's what's happening today

1,247 TOTAL PATIENTS +12%

89 ACTIVE DOCTORS +5%

156 TODAY'S APPOINTMENTS +23%

$45,678 REVENUE (MONTHLY) +18%

Recent Appointments

Today's scheduled appointments

Patient	Doctor	Time	Status
John Doe	Dr. Smith	09:00 AM	Confirmed
Jane Smith	Dr. Johnson	10:30 AM	Pending
Mike Wilson	Dr. Brown	02:00 PM	Confirmed

Department Overview

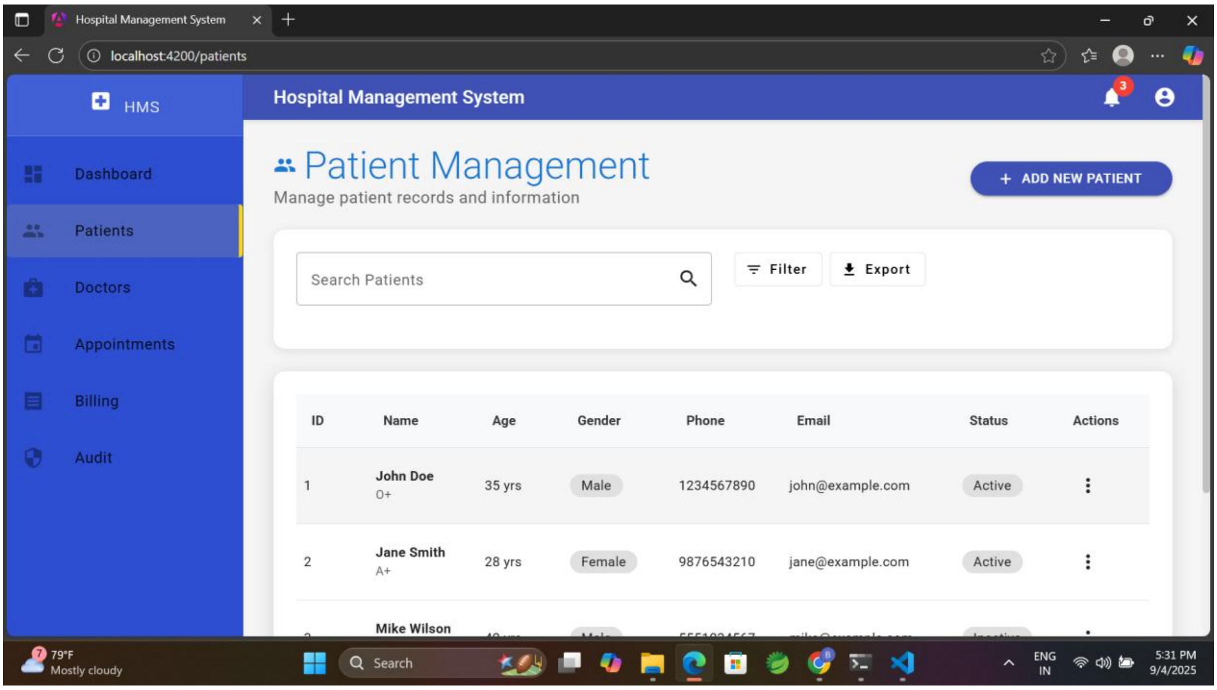
Current occupancy rates

Cardiology 45 patients

Occupancy 85%

Neurology 32 patients

Occupancy



Patient Management

Manage patient records and information

ID	Name	Age	Gender	Phone	Email	Status	Actions
1	John Doe O+	35 yrs	Male	1234567890	john@example.com	Active	⋮
2	Jane Smith A+	28 yrs	Female	9876543210	jane@example.com	Active	⋮
	Mike Wilson						

Conclusion

The Hospital Management System (HMS) is built using a robust microservices architecture with Spring Boot. Services interact via OpenFeign for synchronous communication and Apache Kafka for event-driven messaging. The API Gateway centralizes access, ensuring security, scalability, and efficient routing of requests. This architecture makes the HMS modular, maintainable, and future-ready for healthcare digitalization.