Hotel Management System

Project by: Ashutosh Pandey

Employee ID: 30433

Purpose

To streamline and automate hotel operations by providing an integrated platform for managing room bookings, reservations, and user authentication.

Enhanced Efficiency

Streamlined operations and automated processes

Customer Experience

Improved booking experience and service delivery

Centralized Platform

Real-time management of room availability and user profiles

Target Audience

This project is designed for hotel guests so that they can browse room availabilities, make reservations, and manage their bookings online.

Browse Room Availability

Guests can easily view and filter available rooms and their amenities

Make Reservations

Simple and intuitive booking process with immediate confirmation

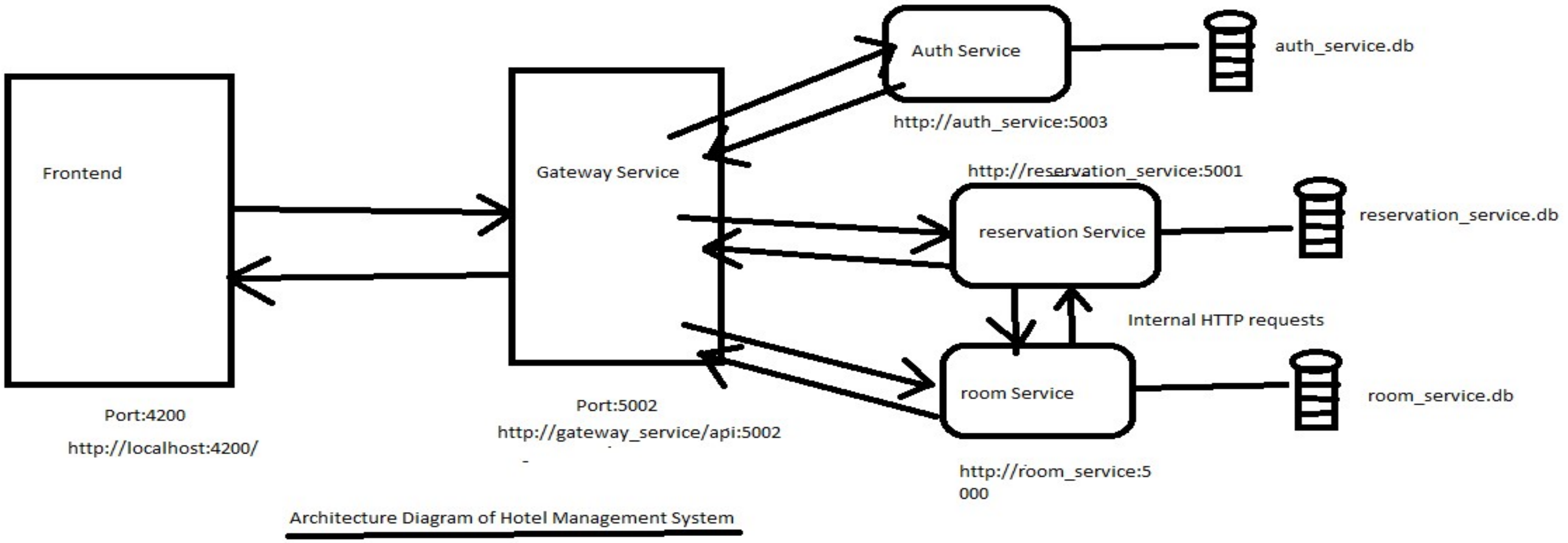
Manage Bookings

View, modify or cancel existing reservations with ease

Architecture Overview

Microservices Architecture

A software architecture pattern that structures the application as a collection of loosely coupled, independently deployable services.



Architecture Diagram of Hotel Management System

- Modular Services
- Scalability
- Technology Flexibility

Room Service Microservice

Room Service – Manages hotel room availability and details.

API Routes

GET /rooms

Returns all rooms with room number, category name, and unavailable dates

GET /rooms/available-categories

Returns rooms category, price per night, image URLs, description, amenities

GET /rooms/available

Returns room numbers, categories, and unavailable dates

PUT /rooms/update_booking

Returns success message either cancelled or updated

Hotel Management System

Reservation Service Microservice

Handles customer reservations and cancellations. Communicates with Room Service to validate availability before confirming reservation.

API Routes

POST **/reservations/**

Creates a new reservation after validating room availability.

Returns: Reservation ID, room number, category, and success message

PUT **/reservations/update_booking**

Updates an existing reservation with new details.

Returns: Reservation ID with new room number and success message

DELETE **/reservation/cancel_booking**

Cancels an existing reservation.

Returns: Success message confirming cancellation

Auth Service Microservice

3. Auth Service

Handles user authentication and authorization, ensuring secure access to the system.

API Routes:

POST /auth/register

Purpose: Register new users in the system

Returns: User registered successfully message

Request body: User credentials (username, password, email)

POST /auth/login

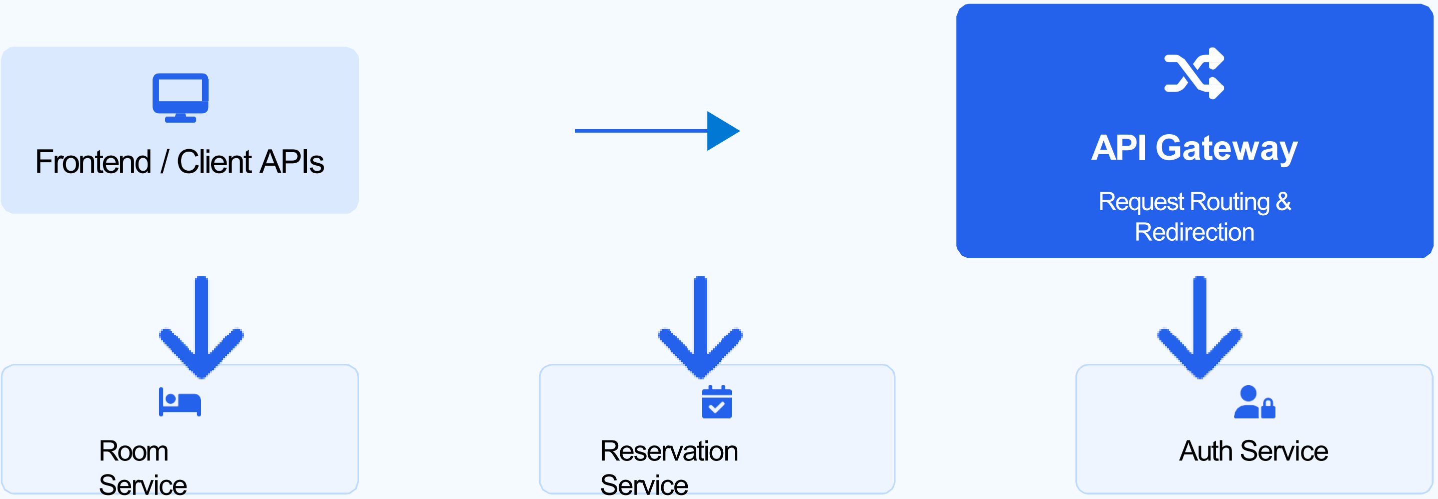
Purpose: Authenticate existing users

Returns: Authentication token and login status

Request body: User credentials (username, password)

Gateway Service Microservice

Common entry point for requests coming from frontend or any API



The Gateway service acts as a single entry point that receives all client requests and intelligently redirects them to the appropriate microservices. This approach simplifies the client interface while enabling independent scaling and deployment of backend services.

Hotel Management System

Frontend Components (Part 1)

1. Auth Component

Handles user authorization and authentication processes

- Manages user login and registration forms
- Validates user credentials through Auth
- Service

  Routes authenticated users to the Room Catalogue component Maintains session state throughout the application

2. Room Catalogue Component

Displays different available categories of rooms

- Fetches available room categories from Room Service
- Provides filtering and sorting options for room selection
- Organizes room categories in a grid layout for easy browsing
- Displays each category using Room Card components

Frontend Components (Part 2)

Checkout Component

- Displays information about the selected room category
- Shows detailed pricing and amount information
- Includes optional "Include breakfast?" feature

Routes to Confirmation component upon successful booking

Confirmation Component

- Displays assigned room number and category
- Shows check-in and check-out date details
- Provides booking summary information
- Routes back to Room Catalogue component

Deployment

The Hotel Management System is deployed and orchestrated using Docker Compose, allowing seamless integration and management of all microservices.

Docker Compose Orchestration

Key Benefits

- Container orchestration for all microservices
- Easy horizontal scaling of services
- Environment consistency across development and production
- Simplified service dependency management

Containerized Services

- Room Service
- Reservation Service
- Auth Service
- API Gateway