



Meeting Room Management

Presented by
Teja Parvathala

Problem statement

- Streamline meeting room reservations and usage in corporate offices.
- Ensure rooms are prepared and ready for meetings with real-time notifications.
- Sends reminder mails to the employee before the scheduled reservation.
- If the room remains unused notify the admin staff, and the Room Service Team to reset or clean the room for the next booking.



System Requirements

Meeting Room Reservation

- Admin staff can reserve rooms with real-time availability.
- Double bookings are prevented by room locking upon reservation.
- System admins manage roles, conflicts, and system settings.

Automated Notifications

- Booking Confirmation mail to employees with reservation details.
- Room Preparation Alerts Room Service Team for setup.
- Reminder Alert mail 15 minutes before the meeting.
- Release Notification for unused rooms after 15 minutes and notify relevant teams.

System Workflow

Reservation

- Admin staff reserves a room.
- The system checks availability and locks the room.

Notifications

- Booking confirmation sent to employees.
- Room preparation alerts sent to Room Service Team.
- Reminder sent to employees before the meeting.
- Auto-release unused rooms after a grace period and notify relevant teams.

Reservation



Confirmation



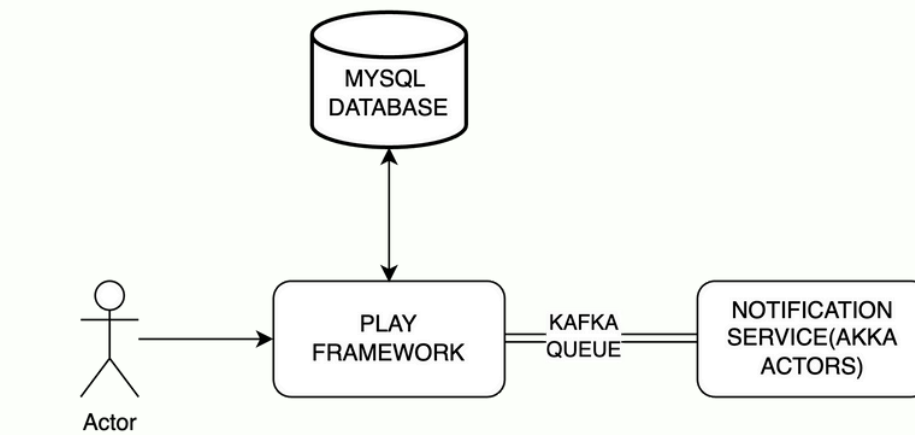
Preparation



Reminder



Release



Employee	
PK	<u>employeeId</u>
	employeeName
	role
	email
	contactNo

Room	
PK	<u>roomId</u>
	roomName
	capacity

Reservation	
PK	<u>reservationId</u>
FK	<u>employeeId</u>
FK	<u>createdBy</u>
FK	<u>roomId</u>
	startTime
	endTime
	purpose



Key Features

- API's for adding employees, fetching employee details, adding rooms, fetching room details, make reservations, check availability for particular time-frame.
- Actors created at notification server takes messages from kafka and notifies the respective teams based on messages.
- Sends reminder mails to the employee before the scheduled reservation.
- If the room remains unused notifying the admin staff, and the Room Service Team to reset or clean the room for the next booking through mail.



System Architecture

Components:

- Backend: REST APIs built with Play Framework for reservation management.
- Notification Microservice: Akka-based for automated notifications.
- Message Queues: Kafka for handling booking, preparation, and release notifications.
- Deployment: Docker containers for backend, and microservices.

The image features a light green background with a darker green border. The border is decorated with four dark green circles at the corners and four light green crescent shapes, each positioned between a corner circle and the nearest edge. The text "Thank You" is centered in a bold, dark green font.

Thank You