

# API Specification Document for Classroom/Venue Management App

## 1 Introduction

This document outlines the API specifications for the Classroom/Venue Management App. The APIs facilitate the management of classroom and venue assignments, bookings, and maintenance, enhancing the efficiency of space utilization on campus.

## 2 APIs Overview

The following APIs are defined:

- **Schedule API:** Manage academic schedules to ensure classrooms and venues are available when needed.
- **Notification API:** Send notifications and reminders for upcoming bookings and maintenance updates.

## 3 Schedule API

### 3.1 Purpose

The Schedule API integrates with academic schedules and allows users (primarily lecturers) to create, update, and retrieve schedules.

### 3.2 Endpoints

#### 3.2.1 Create Schedule

- **Endpoint:** `POST /api/schedules`
- **Description:** Create a new schedule for a classroom or venue.
- **Request Body:**

```
{
  "roomId": "Room101",
  "courseId": "COMS3011A",
  "startTime": "10:00",
  "endTime": "12:00",
  "daysOfWeek": "Monday",
  "startDate": "2024-01-24",
  "endDate": "2024-06-08",
  "recurring": true,
  "userId": "Lecturer1923"
}
```

- **Response:**

```
{
  "scheduleId": "abc123",
  "message": "Schedule created successfully."
}
```

### 3.2.2 Get Schedule

- **Endpoint:** [GET](#) /api/schedules/{scheduleId}
- **Description:** Retrieve a specific schedule by its ID.
- **Response:**

```
{
  "scheduleId": "abc123",
  "roomId": "Room101",
  "courseId": "COMS3011A",
  "startTime": "10:00",
  "endTime": "12:00",
  "daysOfWeek": "Monday",
  "startDate": "2024-01-24",
  "endDate": "2024-06-08",
  "recurring": true,
  "userId": "Lecturer1923"
}
```

### 3.2.3 Update Schedule

- **Endpoint:** [PUT](#) /api/schedules/{scheduleId}
- **Description:** Update an existing schedule.
- **Request Body:**

```
{
  "startTime": "11:00",
  "endTime": "13:00",
  "daysOfWeek": "Wednesday"
}
```

- **Response:**

```
{
  "scheduleId": "abc123",
  "message": "Schedule updated successfully."
}
```

### 3.2.4 Delete Schedule

- **Endpoint:** [DELETE](#) /api/schedules/{scheduleId}
- **Description:** Delete a specific schedule by its ID.
- **Response:**

```
{
  "message": "Schedule deleted successfully."
}
```

### 3.2.5 Get Schedules by Room

- **Endpoint:** [GET](#) /api/schedules/room/{roomId}
- **Description:** Retrieve all schedules for a specific room.
- **Response:**

```
[
  {
    "scheduleId": "abc123",
    "courseId": "COMS3011A",
    "startTime": "10:00",
    "endTime": "12:00",
    "daysOfWeek": "Monday",
    "startDate": "2024-01-24",
    "endDate": "2024-06-08"
  },
  ...
]
```

## 4 Notification API

### 4.1 Purpose

The Notification API handles the scheduling and delivery of notifications to users, including reminders for upcoming bookings and updates on maintenance requests.

### 4.2 Endpoints

#### 4.2.1 Create Notification

- **Endpoint:** [POST](#) /api/notifications
- **Description:** Create a new notification to be sent at a specific time.
- **Request Body:**

```
{
  "userId": "Lecturer1923",
  "type": "booking_reminder",
  "message": "Reminder: Your booking for Room101 starts in 30 minutes.",
  "sendAt": "2024-08-19T09:30:00Z"
}
```

- **Response:**

```
{
  "notificationId": "notif123",
  "message": "Notification scheduled successfully."
}
```

#### 4.2.2 Get Notifications for User

- **Endpoint:** [GET](#) /api/notifications/user/{userId}
- **Description:** Retrieve all notifications scheduled for a specific user.
- **Response:**

```
[
  {
    "notificationId": "notif123",
    "type": "booking_reminder",
    "message": "Reminder: Your booking for Room101 starts in 30 minutes.",
    "sendAt": "2024-08-19T09:30:00Z"
  },
  ...
]
```

#### 4.2.3 Update Notification

- **Endpoint:** **PUT** /api/notifications/{notificationId}
- **Description:** Update an existing notification (e.g., change the message or send time).
- **Request Body:**

```
{
  "message": "Reminder: Your booking for Room101 starts in 15 minutes.",
  "sendAt": "2024-08-19T09:45:00Z"
}
```

- **Response:**

```
{
  "notificationId": "notif123",
  "message": "Notification updated successfully."
}
```

#### 4.2.4 Delete Notification

- **Endpoint:** **DELETE** /api/notifications/{notificationId}
- **Description:** Delete a scheduled notification.
- **Response:**

```
{
  "message": "Notification deleted successfully."
}
```

#### 4.2.5 Send Immediate Notification

- **Endpoint:** **POST** /api/notifications/send
- **Description:** Send a notification immediately without scheduling.
- **Request Body:**

```
{
  "userId": "Lecturer1923",
  "type": "booking_reminder",
  "message": "Immediate notification: Your booking for Room101 is starting now."
}
```

- **Response:**

```
{
  "notificationId": "notif124",
  "message": "Notification sent successfully."
}
```