**Gym Management System: Service Contract Document**

**Group**: Rome was built in a day

**Project** **Name**: Gym Manager

1. **Authentication Service**

|  |  |
| --- | --- |
| Service Name | Authentication Register |
| Purpose | Allows users to register on the system |
| Input | JSON Body: { "firstname": string, "lastname": string, “email”:string, “phone” number, “password: string, “roles”: array } |
| Output | JSON: { "token": string, "user": { "userId": number, "role": string } } |
| Authentication | Not required |
| Consumers | Mobile App |

|  |  |
| --- | --- |
| Service Name | Authentication Login |
| Purpose | Authenticates users and issues JWT tokens for secure access |
| Input | JSON Body: { "email": string, "password": string } |
| Output | JSON: { "token": string, "user": { "userId": number, "role": string } } |
| Authentication | Not required |
| Consumers | Mobile/ Web App |

1. **Coach Class Services**

a) View Assigned Classes

|  |  |
| --- | --- |
| Service Name | Coach Assigned Classes Service |
| Purpose | Retrieve all classes assigned to the logged-in coach |
| Input | None (coach ID derived from JWT token) |
| Output | Array of classes: [{ classId, scheduledDate, scheduledTime, workoutId, coachId, capacity }, ...] |
| Authentication | Required |
| Consumers | Mobile App (Coach view) |

b) Assign Workout to Class

|  |  |
| --- | --- |
| Service Name | Authentication Register |
| Purpose | Assigns a workout to a scheduled class |
| Input | JSON: { "classId": number, "workoutId": number } |
| Output | { "success": true } |
| Authentication | Required |
| Consumers | Mobile App (Coach view) |

1. **Member Booking Services**

a) View available classes

|  |  |
| --- | --- |
| Service Name | Member Available Classes Viewer |
| Purpose | Allows members to see what classes are available to book |
| Input | None (user ID derived from JWT) |
| Output | Array of bookings: [{ classId, scheduledDate, scheduledTime, workoutName }, ...] |
| Authentication | Required |
| Consumers | Mobile App (Member view) |

b) View Booked Classes

|  |  |
| --- | --- |
| Service Name | Member Booked Classes Viewer |
| Purpose | Allows members to see their class bookings |
| Input | None (user ID derived from JWT) |
| Output | Array of bookings: [{ bookingId, classId, scheduledDate, scheduledTime, workoutName }, ...] |
| Authentication | Required |
| Consumers | Mobile App (Member view) |

c) Book a Class

|  |  |
| --- | --- |
| Service Name | Class Booking Service |
| Purpose | Books a user into an available class |
| Input | JSON: { "classId": number } |
| Output | { "success": true } |
| Authentication | Required |
| Consumers | Mobile App (Member view) |

1. **Scheduling Service (Manager)**

a) Create a class

|  |  |
| --- | --- |
| Service Name | Class Booking Service |
| Purpose | Books a user into an available class |
| Input | JSON: { "classId": number } |
| Output | { "success": true } |
| Authentication | Required |
| Consumers | Mobile App (Member view) |

c) Assign a coach to a class

|  |  |
| --- | --- |
| Service Name | Class Booking Service |
| Purpose | Books a user into an available class |
| Input | JSON: { "classId": number } |
| Output | { "success": true } |
| Authentication | Required |
| Consumers | Mobile App (Member view) |