

Pulse Gym Management System SRS

1.Introduction

1.1 Purpose of the document

The document provides the requirements specification for Pulse Gym System. It outlines the functional and non-functional requirements that allow receptionists to manage members, handle subscriptions, record attendance, and manage private sessions while enabling admins to control bundles and user management.

1.2 Scope of Product

The Gym management System will provide tools for managing the daily operations of the gym. The system allows the receptionist to add member information, manage subscriptions, terminate subscriptions, handle private training sessions, and record member attendance. Admins will be able to manage subscription bundles and receptionist accounts.

The system focuses on simplifying member management, improving accuracy of subscription tracking, organized control of gym service .it will serve as a centralized platform that supports both administrative tasks and member-related operations to enhance the gym's efficiency and overflow.

1.3 Definitions, acronyms, abbreviations

User: Someone that actively uses the system, includes Admin and Receptionist.

Member: A customer who is registered in the gym and utilizes its services (a member is not a user in the system).

GMS: Gym Management System.

SRS: Software Requirement Specification.

CRUD: Create, Update, Delete.

DB: Database.

REST: Representational State Transfer.

API: Application Programming Interface.

1.4 Overview of Remainder of Document

It describes the complete requirements of the gym management system .it details the system overall functions ,the specific features provided to each user role and the constraints that apply during development .this structure ensures that all stakeholders have a clear understanding of how the system should operate and what is expected during implementation.

2. General Description

2.1 Product Perspective

The gym management system is a standalone software platform designed to support the operational activities of a gym. It replaces the traditional manual processes of managing members, subscriptions, private sessions and attendance. The system interacts primarily with its internal database where all members and the subscription data are stored and maintained.

2.2 Product Functions

The system provides the following core functions:

Member Management: add new members, update and view member details.

Subscription Management: add new subscription to a member, terminate a member's subscription. Admin can manage subscription bundles.

Private Session Management: add private training sessions to a member, admin can manage session bundles.

User Management: admin can manage receptionists' registration.

System Access: secure login for admins and receptionists.

2.3 User Characteristics

The system will be used by two main types of users:

1- Admin Users:

Add, remove and manage receptionist accounts.

View all system users.

Create, update and remove subscription bundles.

Create, update and remove private session bundles.

2- Receptionist Users:

Add new members.

View member profiles.

Add or terminate member subscriptions.

Add private training sessions for members.

Record daily attendance.

View recent attendance records.

2.4 General constraints

The system can only be accessed by authenticated users (admin or receptionist).

Receptionists have limited permissions and cannot modify system wide bundles or user accounts.

Attendance can only be recorded for members with an active subscription.

All data must be stored in the system's database, no manual edits outside the system are allowed.

The system must follow basic data privacy and security practices to protect member information.

2.5 Assumptions and Dependencies

2.5.1 Assumptions:

The gym's internal systems will support member management, subscription processing and attendance tracking.

2.5.2 Dependencies:

The gym's IT team is responsible for deploying, configuring and maintaining the system infrastructure.

3. Specific Requirements

3.1 Functional Requirements

The system contains four functional modules:

3.1.1 User Management

The system will implement different user roles, utilizing the two primary staff roles: administrator and receptionist.

The system will provide secure login authentication for all authorized staff.

Receptionists:

The system will allow receptionists to do the following:

- Manage member profiles.
- Manage member subscriptions.
- Manage private sessions for members.
- Track attendance.

Administrator:

The administrator role will include the following:

- Manage internal user accounts.
- Manage subscription bundles.
- Manage private session bundles.

3.1.2 Subscription Membership Management

- The system will allow the receptionist to manage subscription activation by assigning a configured bundle to a member and the expiration date will be set automatically.
- The system will track the real-time status (Active, Expired) for all member subscriptions.

3.1.3 Private Session management

- The system will allow the receptionist to add private sessions to the members' subscriptions.

3.1.4 Attendance Management

- The system will allow the user to manage member check-in by recording the time of entry.

- The system will validate access by checking if the member has an active subscription before confirming check-in.

3.2 Non-functional Requirements

3.2.1 Security Requirements

- Any attempt by a receptionist to access admin-only functions will be blocked.
- User passwords must meet a minimum complexity standard (minimum 6 characters).
- The system will automatically record all critical actions in an immutable audit log, capturing the timestamp, and action details.

3.2.2 Usability Requirements

- The user interface must be consistent across all modules to minimize staff training time.
- The system will provide clear and constructive error messages in plain language when an action fails.

3.3 Interface Requirements

- The system will present role-specific navigation menus where admin-exclusive functions are hidden or inaccessible to the receptionist role and vice versa.
- The system shall provide front-end validation for all data entry fields.
- The system's application layer shall be implemented using a RESTful API to manage all client-server communication.

3.4 Logical DB Requirements

The system must store unique member identifiers, personal information (Name, phone number), user login credentials, role identifier, amount of private sessions left, check-in time.