PureGym Management System

Functional Specification Document

Version 1.0

Bese Barnabás

February 26, 2026

Abstract

This document outlines the functional requirements for the PureGym autonomous gym management system. It details the core logic for authentication, membership management, physical access control via QR codes, locker management, and administrative operations. The system is designed for high availability and low latency access control.

Contents

1 Document Control

Date	Version	Change Description
2026-02-26	1.0	Initial Draft based on architectural decisions (ADRs) and feature scope.

2 Actors and Roles

The following actors interact with the system boundaries:

Actor	Description
Guest	An unauthenticated user browsing the public website/app.
Member	An authenticated user (Standard Role). May or may not have an active membership.
Admin	A privileged user with access to back-office configuration, logs, and user management.
System (Worker)	Background processes handling time-based events (notifications) and asynchronous tasks (logging, outbox processing).
Hardware Mock	Simulates physical devices (Revolving Doors, Locker Cabinets) and consumes hardware events.

3 Functional Requirements

3.1 Authentication & Identity (AUTH)

The system relies on stateless authentication to support scalability.

AUTH-01 **Login/Registration.** Users must be able to create an account or log in using:

- Local credentials (Email/Password).
- Google OAuth 2.0 provider.

AUTH-02 **Session Management.** The system uses stateless JWT (JSON Web Tokens). The token is issued upon login and must be included in the `Authorization` header of subsequent requests.

AUTH-03 **Role-Based Access.** The JWT contains claims distinguishing between `Member` and `Admin` roles to enforce API authorization policies.

3.2 Membership & Payments (MEM)

MEM-01 **Purchase Initiation.** Upon selecting a membership type:

1. System creates an `Order` record in "Pending" state.
2. System pushes an `OrderCreated` event.

3. Client is redirected to the external Payment Service.

MEM-02 **Payment Processing.** The Payment Service processes the transaction. On success, a webhook is sent back to the API.

MEM-03 **Activation.** Upon receiving the `PaymentSuccess` event:

- The Order state updates to "Completed".
- The Member's status is set to Active (or tickets are incremented).

Figure 1: Payment and Membership Activation Flow

3.3 Physical Access Control (ACS)

Performance is prioritized over immediate database persistence to ensure rapid entry.

ACS-01 **QR Generation.** A Member can request an entry QR code.

- System generates a secure OTP (One Time Password).
- OTP is stored in **In-Memory Cache** for low latency.

ACS-02 **Entry Validation.** When the QR is scanned at the 'EnterGym' endpoint:

- System validates OTP against in-memory state.
- **Rules:** Member must have active subscription OR > 0 tickets. Member must not be banned.
- **Success:** Returns immediate HTTP 200 OK to hardware mock to open door.

ACS-03 **Async Processing.** After granting access, the system asynchronously:

- Publishes a `UserEntered` event via the Outbox pattern.
- Deducts a ticket (if applicable).
- Persists entry in `AccessLog` table.

3.4 Locker Management (LCK)

LCK-01 **Cabinet Action.** Members can request to Lock/Unlock a specific cabinet ID.

LCK-02 **Authorization.** System verifies ownership/rental status of the cabinet.

LCK-03 **Real-time Feedback.** System uses **Server-Sent Events (SSE)** to push status changes (`LockerLocked`, `LockerUnlocked`) to the client UI immediately after hardware confirmation.

3.5 Notifications (NOT)

NOT-01 **Expiration Warning.** A background worker scans for memberships expiring in < 3 days and triggers an email notification.

NOT-02 **Low Ticket Warning.** A background worker scans for users with < 3 tickets remaining and triggers an email notification.

NOT-03 **Frequency Cap.** Notifications are capped to prevent spam (e.g., max 1 per 24h).

3.6 Administration (ADM)

ADM-01 **Membership Types.** Admins can Create, Update (price/benefits), or Invalidate (soft-delete) membership tiers.

ADM-02 **User Management.** Admins can list users, view profiles, and force-add memberships to users (e.g., for support/cash payments).

ADM-03 **Audit Logs.** Admins have read-only access to `GymEntered` and `CabinetAccessed` logs.

4 Data Flow & State Machines

4.1 Membership Lifecycle

The membership status transitions through the following states:

1. **None:** User has no active plan.
2. **Pending:** Purchase initiated, awaiting payment provider callback.
3. **Active:** Payment confirmed, access rights granted.
4. **Expired:** Duration elapsed or ticket count reached zero.
5. **Cancelled:** Access revoked by Administrator.

4.2 Door Entry Sequence (Happy Path)

1. **Client:** Request QR Code.
2. **API:** Validate Member and Membership $\rightarrow$ Generate OTP $\rightarrow$ Store in RAM $\rightarrow$ Return QR.
3. **Hardware:** Scan QR $\rightarrow$ Send to API.
4. **API:** Validate OTP (RAM) $\rightarrow$ Send out GymEntryRequestEvent $\rightarrow$ Return OK $\rightarrow$ **Hardware Opens Door**.
5. **API (Async):** Publish Event $\rightarrow$ Worker consumes $\rightarrow$ Write DB Log / Decrement Ticket.