Authentication & JWT Token Specification (MVP)

Overview

This document defines the authentication structure for the MVP. Authentication is based on JWT with access and refresh tokens. The scope is limited to email/password authentication with optional account verification.

User Model (Auth-Relevant Fields)

id (UUID)
email (unique)
passwordHash
firstName
lastName
avatarId (string, provided by frontend)
isVerified (boolean)
createdAt
updatedAt

Token Strategy

Access Token:
- JWT
- Expiry: 5 days (configurable up to 7 days)
- Used for all authenticated API requests

Refresh Token:
- Long-lived token
- Used to issue new access tokens
- Rotated on each refresh

JWT Payload (Access Token)

{
sub: userId,
email: userEmail,
iat: issuedAt,
exp: expiry
}

Auth Endpoints

POST /auth/register
Request: { email, password, firstName, lastName, avatarId }
Response: { success: true, userId }

POST /auth/login
Request: { email, password }
Response: { accessToken, refreshToken, user }

POST /auth/refresh
Request: { refreshToken }
Response: { accessToken, refreshToken }

POST /auth/verify (optional)
Request: { userId, code }
Response: { success: true }

Security Notes

- Passwords must be hashed (bcrypt or argon2)
- Refresh tokens stored hashed in DB
- Refresh tokens invalidated on logout
- Access tokens are stateless
- Backend only stores avatarId

Out of Scope for MVP

- Social login
- Password reset
- Roles and permissions
- Admin tools