# Product Requirements Document (PRD) for User Management System

Contents

[Product Requirements Document (PRD) for User Management System 1](#_Toc205200972)

[1.Overview 1](#_Toc205200973)

[2.Functional Requirements 1](#_Toc205200974)

[3.Non-Functional Requirements 1](#_Toc205200975)

[4. Component Diagrams 2](#_Toc205200976)

[4.1. Sequence Diagram: 2](#_Toc205200977)

[4.2. Class Diagram 6](#_Toc205200978)

[5. Database Schema 7](#_Toc205200979)

[5.1 Database Design 8](#_Toc205200980)

[6. **API Endpoints (Sample)** 8](#_Toc205200981)

## 1.Overview

The User Management System will allow admin and service accounts to create users, assign roles, and manage access permissions, with secure login and auditing.

## 2.Functional Requirements

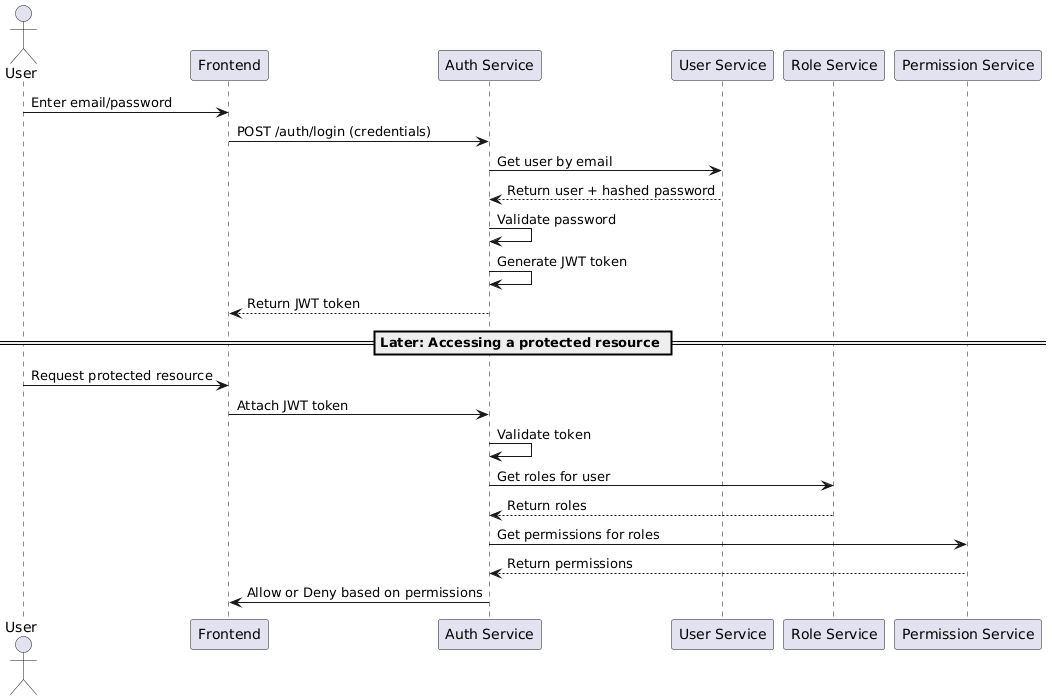
1. User Registration (Admin only or via Invite)
2. Role Creation and Assignment
3. Secure Login with JWT
4. Token Refresh & Expiry
5. Logout and Session Revocation
6. Two-Factor Authentication (Optional)
7. Audit Logging (User, Role, Permission Changes)
8. Forgot Password Workflow
9. Access Control Based on Roles

## 3.Non-Functional Requirements

* Scalability to support 1M+ users
* 99.99% availability
* Token expiration handling
* Encryption of sensitive data
* RBAC that is dynamic and configurable

## 4. Component Diagrams

### 4.1. Sequence Diagram:

****

**Fig 1:** Sequence Diagram For user

#### 4.1.1 Sequence Diagram: Login and Token Flow

A diagram of a server

AI-generated content may be incorrect.

**Fig 1.1:** Sequence Login and Token Flow

#### 4.1.2: Sequence Diagram: Token Validation and Refresh

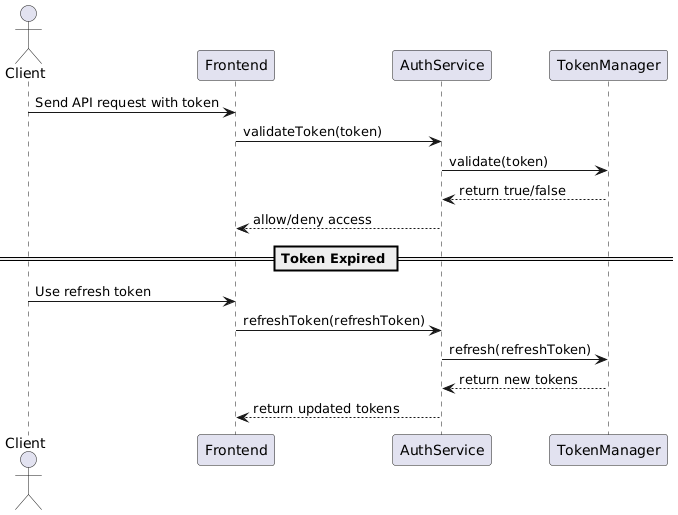


Fig 1.2: Token validation

#### 4.1.3: Expanded with 2FA and OAuth

A diagram of a computer

AI-generated content may be incorrect.

Fig 1.3: 2Fa and OAuth2

#### 4.1.4: OAuth Login (Google/GitHub)

A diagram of a software company

AI-generated content may be incorrect.

Fig 1.4: OAUTH Login

#### 4.1.5 Sequence Diagram: 2FA Login (TOTP)

A diagram of a server

AI-generated content may be incorrect.

Fig 1.5: 2FA Login Sequence Diagram

#### 4.1.6. Flow Breakdown:

* User logs in with email/password
* Auth Service checks credentials with User Service
* JWT token is issued
* On accessing a protected resource:
  + Token is verified
  + User’s roles are fetched
  + Permissions are checked
  + Access is granted or denied

### 4.2. Class Diagram

A diagram of a computer

AI-generated content may be incorrect.

Fig 1.3: Class Diagram User Service

#### 4.2.1 User & Authentication service class diagram

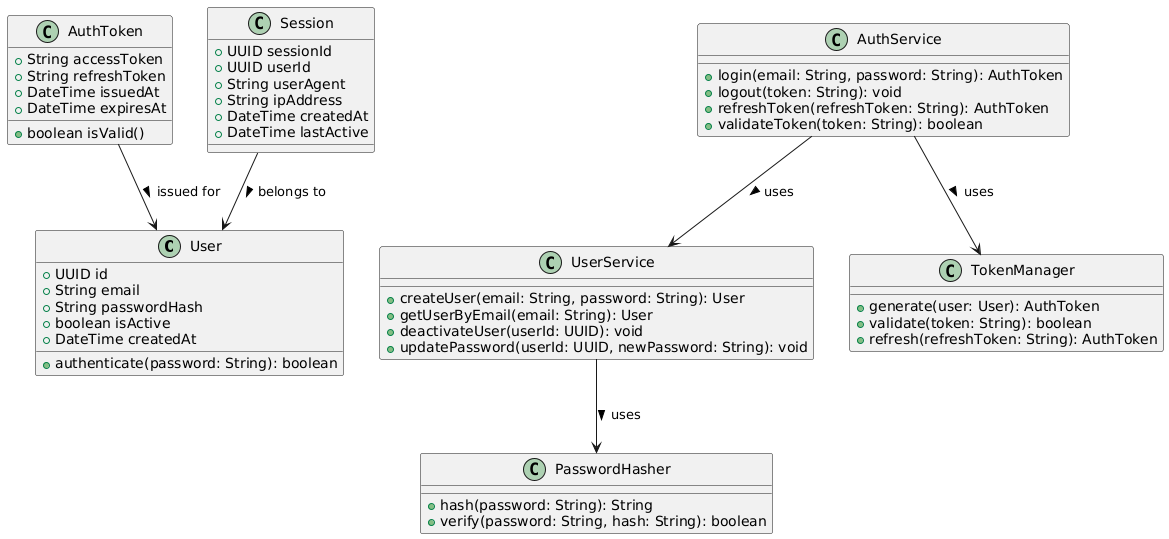


Fig 1.4: User and Authentication class diagram

## 5. Database Schema

CREATE TABLE users (

id Integer PRIMARY KEY,

first\_name Varchar(100),

last\_name Varchar(100),

email VARCHAR(255) UNIQUE NOT NULL,

hashed\_password TEXT NOT NULL,

created\_at TIMESTAMP DEFAULT CURRENT\_TIMESTAMP,

is\_active BOOLEAN DEFAULT TRUE

);

CREATE TABLE roles (

id Integer PRIMARY KEY,

name VARCHAR(50) UNIQUE NOT NULL,

description TEXT

);

CREATE TABLE permissions (

id Integer PRIMARY KEY,

name VARCHAR(100),

resource VARCHAR(100)

);

CREATE TABLE user\_roles (

user\_id Integer REFERENCES users(id),

role\_id Integer REFERENCES roles(id),

PRIMARY KEY(user\_id, role\_id)

);

CREATE TABLE role\_permissions (

role\_id Integer REFERENCES roles(id),

permission\_id Integer REFERENCES permissions(id),

PRIMARY KEY(role\_id, permission\_id)

);

### 5.1 Database Design

A screenshot of a computer

AI-generated content may be incorrect.

## 6. **API Endpoints (Sample)**

| **Endpoint** | **Method** | **Auth** | **Description** |
| --- | --- | --- | --- |
| /auth/login | POST | ❌ | Login with email/password |
| /auth/logout | POST | ✅ | Invalidate current token |
| /userall | GET | ✅ | List all users (Admin only) |
| /users | POST | ✅ | Create user (Admin only) |
| /roles | POST | ✅ | Create new role |
| /roles/assign | POST | ✅ | Assign role to user |
| /permissions | POST | ✅ | Define new permissions |
| /access-check | POST | ✅ | Check if user has permission |

|  |  |  |
| --- | --- | --- |
| Kaushik Nandy | 30-07-2025 | V 1.0 |
|  |  |  |
|  |  |  |