High-Level Design (HLD) - Login & JWT Authentication

# Revision History

|  |  |  |
| --- | --- | --- |
| Version | Date | Description |
| 1.0 | 2025-07-21 | Changed coding language from TypeScript to HTML + Python (Streamlit) for frontend; Changed backend from MongoDB to MySQL. |

# 1. Overview

This document describes the high-level design for a Login and JWT Authentication system, with frontend implemented in HTML + Python (Streamlit) and backend using MySQL.

# 2. Actors

• User – Registers, logs in, and accesses application features.  
• System – Validates credentials, generates JWT tokens, and authorizes requests.  
• Database (MySQL) – Stores user credentials and session data.

# 3. Use Cases

• Sign-up: User registers with username, password, and email.  
• Sign-in: User logs in with credentials; system returns JWT token.  
• Token Validation: API endpoints verify token for authentication.  
• Logout: Client removes stored JWT.

# 4. Sequence Flow

Sign-up & Sign-in sequence:  
```mermaid  
sequenceDiagram  
 participant U as User  
 participant F as Frontend (HTML+Streamlit)  
 participant B as Backend (Python)  
 participant DB as MySQL  
 U->>F: Enter details  
 F->>B: POST /signup or /signin  
 B->>DB: Store/Retrieve User  
 DB-->>B: Success/Failure  
 B->>B: Generate JWT (if signin)  
 B-->>F: Return success + token  
 F-->>U: Show success or error  
```