📝 Software Requirement Specifications (SRS)

1. API for System Version

This endpoint is for basic system health and versioning information.

Specification	Details
Endpoint Name	/version
Method Type	GET
Input Parameters	None
Output Response Code (Success)	200 OK
Output Response Body (Success)	JSON object: {"version": "1.0.0", "service": "User-Management-Service"}
Output Response Code (Failure)	500 Internal Server Error (If service fails to respond)
Output Response Body (Failure)	JSON object: {"message": "Internal Server Error"}

---

2. API for User Registration (Pre-Confirmation)

This endpoint creates a pending user record and generates the confirmation link.

Specification	Details

Endpoint Name	/register
Method Type	POST
Input Parameters	**Request Body (JSON)**
	- email: **Mandatory** (String, 5 to 25 characters)
	- password: **Mandatory** (String, exactly 6 characters)
Output Response Code (Success)	201 Created
Output Response Body (Success)	JSON object: {"message": "Registration successful, please click the confirmation link.", "confirmation_link": "http://api-host:port/confirm_registration/standard_length_token_string"}
Output Response Code (Failure)	400 Bad Request (Invalid input/Email already exists)
Output Response Body (Failure)	JSON object: {"message": "Invalid input format (e.g., email length, password length)", "errors": {"field_name": "Error details"}} OR {"message": "User with this email already exists."}

---

3. API for Registration Confirmation

This endpoint completes the user's registration by validating the token.

Specification	Details
Endpoint Name	/confirm_registration/{token}
Method Type	GET
Input Parameters	**Path Parameter**
	- token: **Mandatory** (String, standard length token)
Output Response Code (Success)	200 OK
Output Response Body (Success)	JSON object: {"message": "Registration successfully confirmed. You can now login."}
Output Response Code (Failure)	404 Not Found (Invalid or expired token)
Output Response Body (Failure)	JSON object: {"message": "Invalid or expired confirmation token."}

---

4. API for User Login

This endpoint authenticates the user and provides a success message (no token required based on the simple spec).

Specification	Details

Endpoint Name	/login
Method Type	POST
Input Parameters	**Request Body (JSON)**
	- email: **Mandatory** (String, 5 to 25 characters)
	- password: **Mandatory** (String, exactly 6 characters)
Output Response Code (Success)	200 OK
Output Response Body (Success)	JSON object: {"message": "Login successful.", "user_id": 123}
Output Response Code (Failure)	401 Unauthorized (Invalid credentials or unconfirmed account)
Output Response Body (Failure)	JSON object: {"message": "Invalid email or password."} OR {"message": "Account not confirmed. Please check your email."}

5.📦 Database Schema and Queries (SQLite)

You will need a single table to manage users and their confirmation status.

User Table Schema

Field Name	Data Type	Constraint	Description

id	INTEGER	PRIMARY KEY AUTOINCREMENT	Unique user identifier.
email	TEXT	NOT NULL, UNIQUE	User's email (used as username).
password	TEXT	NOT NULL	The 6-letter password (no encryption required per spec).
confirmation_token	TEXT	UNIQUE	Temporary token for confirmation.
is_confirmed	INTEGER	NOT NULL, DEFAULT 0	0=Pending, 1=Confirmed.
created_at	TEXT	NOT NULL	Timestamp of registration.

SQLite Schema Creation Query

SQL

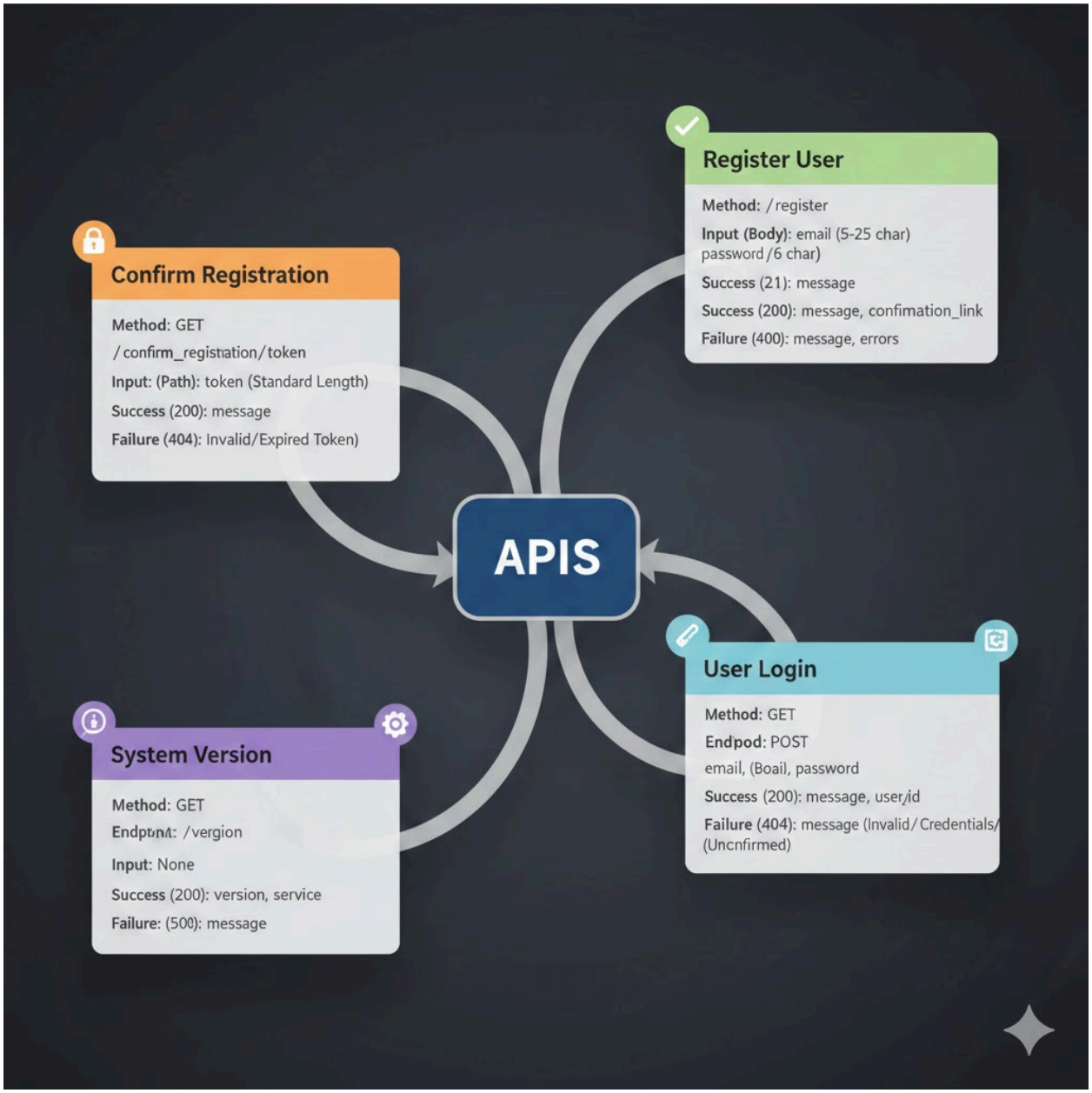
```
CREATE TABLE users (
    id INTEGER PRIMARY KEY AUTOINCREMENT,
    email TEXT NOT NULL UNIQUE CHECK(LENGTH(email) BETWEEN 5 AND 25),
    password TEXT NOT NULL CHECK(LENGTH(password) = 6),
    confirmation_token TEXT UNIQUE,
    is_confirmed INTEGER NOT NULL DEFAULT 0,
    created_at TEXT NOT NULL
);
```

Required Database Queries (CRUD)

Operation	Purpose	Example Query Template

CREATE (Register)	Insert new pending user	INSERT INTO users (email, password, confirmation_token, created_at) VALUES (?, ?, ?, DATETIME('now'))
READ (Login)	Fetch user for authentication	SELECT id, password, is_confirmed FROM users WHERE email = ?
UPDATE (Confirm)	Mark user as confirmed	UPDATE users SET is_confirmed = 1, confirmation_token = NULL WHERE confirmation_token = ? AND is_confirmed = 0
READ (Token Check)	Check if a token is valid	SELECT id FROM users WHERE confirmation_token = ?

6. Mind Map


APIS
Register User
Method: /register
Input (Body): email (5-25 char) password /6 char)
Success (21): message
Success (200): message, confimation_link
Failure (400): message, errors
Confirm Registration
Method: GET
/ confirm_registration/token
Input: (Path): token (Standard Length)
Success (200): message
Failure (404): Invalid/Expired Token)
User Login
Method: GET
Endpod: POST
email, (Boail, password
Success (200): message, user/id
Failure (404): message (Invalid/ Credentials/ (Uncnfirmed)
System Version
Method: GET
Endpunt: /vergion
Input: None
Success (200): version, service
Failure: (500): message