**User Registration and Authentication API Documentation**

This documentation provides information on the usage and implementation details of the User Registration and Authentication API. This API allows users to register, log in, by using secure JSON Web Token (JWT) mechanism.

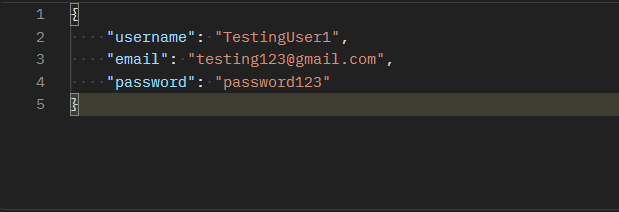
**Register User:**

Registers a new user by providing a username, email, and password.

Endpoint: http://localhost:3500/register

Method: POST

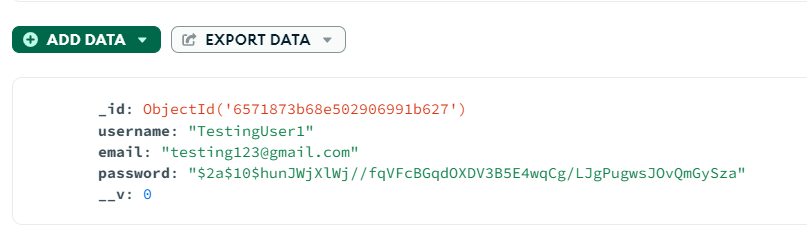
Request Body:



Response (Success - HTTP 201):



And the user has been saved in MongoDB compass:



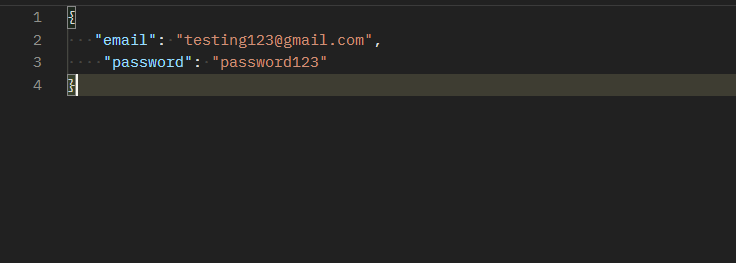
**Login User:**

Authenticates a user and provides a JWT token API request.

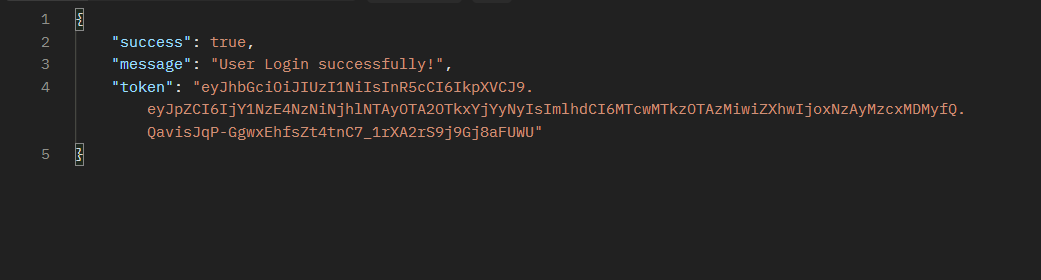
Endpoint: http://localhost:3500/login

Method: POST

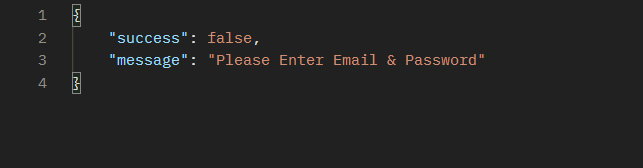
Request Body:



Response (Success - HTTP 200):



Response (Failure - HTTP 401):



**Error Handling:**

The API returns appropriate HTTP status codes and JSON responses for both success and failure scenarios.

Error messages provide meaningful information to assist developers in identifying and resolving issues.

**Security Considerations:**

Passwords are securely hashed before storing them in the database.

JSON Web Tokens (JWT) are used for authentication, providing a secure mechanism for user sessions.

**Implementation Details:**

The API is implemented using Node.js and Express.

MongoDB is used as the NoSQL database for storing user information.

Bcrypt.js library is employed for password hashing.

JSON Web Tokens (JWT) are generated and verified for user authentication.

**Conclusion:**

This documentation provides overview of the user authentication API, including registration, login.