API Test Cases: Authorisation Module (v2)

Introduction

This document outlines the suite of API test cases for the core authentication module (auth.py). The purpose of these tests is to verify the correctness, security, and robustness of the endpoints responsible for user login, logout, and password reset.

Testing Environment

- **Tool**: Postman
- **Base URL**: http://127.0.0.1:5000
- **Prerequisites**: The application is running, and the database has been seeded.

Feature: User Login

- **Feature ID**: AUTH-001
- **Endpoint**: POST /auth/login
- **Description**: Verifies the functionality of the user login endpoint.

Test Case ID	Description	Test Steps	Expected Result
AUTH-001-TC-001	Successful Login	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login. 3. Set Body to raw JSON with valid credentials. 4. Send request.	**ER**: Status code is **200 OK**. The response body contains a non-empty access_token.
AUTH-001-TC-002	Invalid Password	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login.	**ER**: Status code is **401 Unauthorized**. The response

		3. Set Body to raw JSON with a correct username but incorrect password. 4. Send request.	body contains the message: "Invalid credentials".
AUTH-001-TC-003	Invalid Username	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login. 3. Set Body to raw JSON with a non-existent username. 4. Send request.	**ER**: Status code is **401 Unauthorized**. The response body contains the message: "Invalid credentials".
AUTH-001-TC-004	Missing Credentials	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login. 3. Set Body to an empty raw JSON object: {}. 4. Send request.	**ER**: Status code is **400 Bad Request**. The response contains a validation error message.
AUTH-001-TC-005	Empty Credentials	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login. 3. Set Body to raw JSON with empty strings for credentials. 4. Send request.	**ER**: Status code is **400 Bad Reques**t. The response contains the message: "Username and password are required".
AUTH-001-TC-006	Non-JSON Request	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/login.	**ER**: Status code is **400 Bad Request**. The response contains a

		3. Do not set the Content-Type header to application/json. 4. Send request.	message about invalid JSON format.

- **Results**

Test Case ID	Result	Status	Notes
AUTH-001-TC-001	Similar to ER	PASSED	N/A
AUTH-001-TC-002	Similar to ER	PASSED	N/A
AUTH-001-TC-003	Similar to ER	PASSED	N/A
AUTH-001-TC-004	Similar to ER	PASSED	N/A
AUTH-001-TC-005	Similar to ER	PASSED	N/A
AUTH-001-TC-006	Similar to ER	PASSED	N/A

Feature: User Logout

- **Feature ID**: AUTH-002
- **Endpoint**: DELETE /auth/logout
- **Description**: Verifies that a user can securely log out.

Test Case ID	Description	Test Steps	Expected Result
AUTH-002-TC-001	Successful Logout	1. Perform a successful login to get a valid access_token.	**ER**: Status code is **200 OK**. The response body contains the

		2. Set method to DELETE. 3. Set URL to {{baseURL}}/auth/logout. 4. Set Authorisation header to Bearer {{accessToken}}. 5. Send request.	message: "Successfully logged out!".
AUTH-002-TC-002	Verify Token is Blocklisted	1. Complete all steps for AUTH-002-TC-001. 2. Using the same access_token, immediately attempt to call the /auth/logout endpoint again. 3. Send request.	**ER**: Status code is **401 Unauthorized**. The response body contains the message: "Token has been revoked".
AUTH-002-TC-003	Logout with Invalid Token	1. Set method to DELETE. 2. Set URL to {{baseURL}}/auth/logout. 3. Set Authorisation header to Bearer an-invalid-token. 4. Send request.	**ER**: Status code is **401 Unauthorized**. The response body contains a message like "Token is invalid or expired".
AUTH-002-TC-004	Logout without Token	1. Set method to DELETE. 2. Set URL to {{baseURL}}/auth/logout.	**ER**: Status code is **401 Unauthorized**. The response body contains a message like

		3. Do not include an Authorisation header. 4. Send request.	"Missing Authorization Header".

- **Results:**

Test Case ID	Result	Status	Notes
AUTH-002-TC-001	Similar to ER	PASSED	N/A
AUTH-002-TC-002	Similar to ER	PASSED	N/A
AUTH-002-TC-003	Similar to ER	PASSED	N/A
AUTH-002-TC-004	Similar to ER	PASSED	N/A

Feature: Password Reset

- **Feature ID**: AUTH-003
- **Endpoints**: POST /auth/request_reset, PUT /auth/reset
- **Description**: Verifies the password reset request and confirmation process.

Test Case ID	Description	Test Steps	Expected Result
AUTH-003-TC-001	Successful Password Reset Request	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/request_reset. 3. Set Body to raw JSON with an existing employee's email: { "email": "john.smith@example.com" }. 4. Send request.	**ER**: Status code is **200 OK**. The response body contains the message: "Code has been sent".
AUTH-003-TC-002	Request with Non-existent Email	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/request_reset. 3. Set Body to raw JSON with a non-existent email. 4. Send request.	**ER**: Status code is **404 Not Found**. The response body contains the message: "Email not found".
AUTH-003-TC-003	Request with Missing Email Field	1. Set method to POST. 2. Set URL to {{baseURL}}/auth/request_reset. 3. Set Body to an empty raw JSON object: {}. 4. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains the message: "Missing email field in JSON".
AUTH-003-TC-004	Successful Password Reset	1. Perform AUTH-03-TC-001 to receive a reset token. 2. Set method to PUT.	**ER**: Status code is **200 OK**. The response body contains the message:

		3. Set URL to {{baseURL}}/auth/reset?token=<valid_token>. 4. Set Body to raw JSON with a new password: { "new_password": "new_secure_password" }. 5. Send request.	"Password reset successfully".
AUTH-003-TC-005	Reset with Invalid Token	1. Set method to PUT. 2. Set URL to {{baseURL}}/auth/reset?token=an-invalid-or-expired-token. 3. Set Body to raw JSON with a new password. 4. Send request.	**ER**: Status code is **500 Internal Server Error or similar**, indicating the token is invalid.
AUTH-003-TC-006	Reset with Missing New Password	1. Perform AUTH-03-TC-001 to receive a reset token. 2. Set method to PUT. 3. Set URL to {{baseURL}}/auth/reset?token=<valid_token>. 4. Set Body to an empty raw JSON object: {}. 5. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains the message: "Missing new password".

- **Results**

Test Case ID	Result	Status	Notes
AUTH-003-TC-001	Similar to ER	PASSED	N/A

AUTH-003-TC-002	Similar to ER	PASSED	N/A
AUTH-003-TC-003	Similar to ER	PASSED	N/A
AUTH-003-TC-004	Similar to ER	PASSED	N/A
AUTH-003-TC-005	Similar to ER	PASSED	N/A
AUTH-003-TC-006	Similar to ER	PASSED	N/A