Introduction

This document provides the API test cases for the account management module (account_route.py). These tests are designed to verify the functionality of the endpoints responsible for creating, viewing, and searching for user accounts in the Hammer & Grammar Management System.

The test cases are divided into three main features:

1. **Add Account**: Validating the creation of new user accounts.
2. **Get All Accounts**: Verifying the retrieval of all account records.
3. **Search for Account**: Ensuring that individual accounts can be retrieved by their ID.

Note: The endpoints in account_route.py currently lack role-based protection. The test cases will assume that a 'Manager' role should be required for these actions and will include tests for unauthorised access to highlight this security gap.

Testing Environment

- **Tool**: Postman
- **Base URL**: http://127.0.0.1:5000
- **Prerequisites**:
    - The application is running and the database is seeded.
    - A valid JWT for a user with the 'Manager' role has been obtained and stored (e.g., as {{managerToken}}).
    - A valid JWT for a user with a non-manager role (e.g., 'Teacher') has been obtained (e.g., as {{teacherToken}}).

Feature: Add Account

- **Feature ID**: ACC-001
- **Endpoint**: POST /account/add
- **Description**: Verifies the creation of new user accounts.
- **Test Cases**:

Test Case ID	Description	Test Steps	Expected Result
ACC-001-TC-001	Successful Account Creation	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with valid, unique data for a new account linked to an existing employee (e.g., EM004). 5. Send request.	**ER**: Status code is **201 Created**. The response body contains the newly created account object (excluding the password).
ACC-001-TC-002	Duplicate Username	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with a username that already exists. 5. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains a message like Violate database constraint.
ACC-001-TC-003	Duplicate Employee ID	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{managerToken}}.	**ER**: Status code is **400 Bad Request**. The response body contains a message like Violate database constraint.

		4. Set Body to raw JSON for an employee_id that already has an account. 5. Send request.	
ACC-001-TC-004	Missing Required Field	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON but omit the password field. 5. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains a validation error message indicating the password field is missing.
ACC-001-TC-005	Non-existent Employee ID	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with an employee_id that does not exist (e.g., EM999). 5. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains a message indicating a foreign key constraint violation.
ACC-001-TC-006	Unauthorised Access (Attempt by non-Manager)	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/account/add. 3. Set Authorization header to Bearer {{teacherToken}}. 4. Set Body to raw JSON with valid data.	**ER**: Status code is 403 Forbidden. The response body contains an access denied message.

		5. Send request.	

- **Results:**

Test Case ID	Result	Status	Notes
ACC-001-TC-001	Similar to ER	PASSED	N/A
ACC-001-TC-002	Similar to ER	PASSED	N/A
ACC-001-TC-003	• **201 Created**	FAILED	See ACC-BUG-001
ACC-001-TC-004	Similar to ER	PASSED	N/A
ACC-001-TC-005	Similar to ER	PASSED	N/A
ACC-001-TC-006	Assume that the output is **403 Forbidden**	FAILED	See ACC-BUG-002

- **Feature: Get All Accounts**
  - **Feature ID**: ACC-002
  - **Endpoint**: GET /account/
  - **Description**: Verifies the retrieval of all user accounts.
  - **Test Cases:**

Test Case ID	Description	Test Steps	Expected Result
ACC-002-TC-001	Successful Retrieval	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/. 3. Set Authorization header to Bearer {{managerToken}}.	**ER**: Status code is 200 OK. The response body is a JSON array of account objects.

		4. Send request.	
ACC-002-TC-002	**Unauthorised Access** (Attempt by non-Manager)	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/. 3. Set Authorization header to Bearer {{teacherToken}}. 4. Send request.	**ER**: Status code is 403 Forbidden. The response body contains an access denied message. *(Note: This test will fail until role protection is added to the endpoint.)*

- **Results**

Test Case ID	Result	Status	Notes
ACC-002-TC-001	Similar to ER	PASSED	N/A
ACC-002-TC-002	Assume that the output is **403 Forbidden.**	FAILED	See ACC-BUG-002

- **Feature: Search for Account**
  - **Feature ID**: ACC-03
  - **Endpoint**: GET /account/search
  - **Description**: Verifies the retrieval of a single account by its ID.
  - **Test Cases**:

Test Case ID	Description	Test Steps	Expected Result
ACC-003-TC-001	Successful Search (Existing ID)	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/search?id=<existing_account_id> 3. Set Authorization header to Bearer {{managerToken}}. 4. Send request.	**ER**: Status code is **200 OK**. The response body is a JSON object containing the correct Employee ID.
ACC-003-TC-002	Search for Non-existent ID	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/search?id=ACC999 3. Set Authorization header to Bearer {{managerToken}}. 4. Send request.	**ER**: Status code is **200 OK** (**or 404 Not Found**). The response body contains the message: No account found.
ACC-003-TC-003	Missing ID Parameter	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/search 3. Set Authorization header to Bearer {{managerToken}}. 4. Send request.	**ER**: Status code is **400 Bad Request**. The response body contains a message indicating the id parameter is missing.

ACC-003-TC-004	Unauthorised Access (Attemp by non-Manager)	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/account/search?id=<existing_account_id> 3. Set Authorization header to Bearer {{teacherToken}}. 4. Send request.	**ER**: Status code is **403 Forbidden**. The response body contains an access denied message. *(Note: This test will fail until role protection is added to the endpoint.)*

- **Results**

Test Case ID	Result	Status	Notes
ACC-003-TC-001	Similar to ER	PASSED	N/A
ACC-003-TC-002	• **409 Conflict**	FAILED	See ACC-BUG-003
ACC-003-TC-003	Similar to ER	PASSED	N/A
ACC-003-TC-004	Assume that the output is **403 Forbidden**	FAILED	See ACC-BUG-002