

Introduction

This document provides the API test cases for the employee management module, which is defined in `employee_route.py`. These tests are designed to verify the functionality of the endpoints responsible for creating and retrieving employee profiles, which are the foundation for user roles and permissions within the Hammer & Grammar Management System.

The test cases cover two primary features:

1. **Add Employee:** Validating the creation of new employee records.
2. **Get All Employees:** Verifying the retrieval of all employee records.

Note: Similar to the account routes, the endpoints in `employee_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 vulnerability.

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 Employee

- **Feature ID:** EMP-01
- **Endpoint:** `POST /employee/add`
- **Description:** Verifies the creation of new employee profiles.
- **Test Cases:**

Test Case ID	Description	Test Steps	Expected Result
EMP-001-TC-001	Successful Employee Creation	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/employee/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with valid, unique data for a new employee. 5. Send request. 	ER: Status code is 201 Created . The response body contains the newly created employee object.
EMP-001-TC-002	Duplicate ID	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/employee/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with an id that already exists (e.g., "EM001"). 5. Send request. 	ER: Status code is 400 Bad Request or 409 Conflict . The response body contains an error message indicating a primary key violation.
EMP-001-TC-003	Duplicate Email	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/employee/add. 3. Set Authorization header to Bearer {{managerToken}}. 4. Set Body to raw JSON with an email that already exists. 5. Send request. 	ER: Status code is 400 Bad Request or 409 Conflict . The response body contains an error message indicating a unique

			constraint violation.
EMP-001-TC-004	Missing Required Field	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to <code>http://127.0.0.1:5000/employee/add</code>. 3. Set Authorization header to <code>Bearer {{managerToken}}</code>. 4. Set Body to raw JSON but omit the role field. 5. Send request. 	ER: Status code is 400 Bad Request . The response body contains a validation error message indicating the role field is missing.
EMP-001-TC-005	Invalid Role	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to <code>http://127.0.0.1:5000/employee/add</code>. 3. Set Authorization header to <code>Bearer {{managerToken}}</code>. 4. Set Body to raw JSON with an invalid role. 5. Send request. 	ER: Status code is 400 Bad Request . The response body contains an error message indicating a database check constraint violation for the role field.
EMP-001-TC-006	Invalid teacher_status for non-teacher role	<ol style="list-style-type: none"> 1. Set method to POST. 2. Set URL to <code>http://127.0.0.1:5000/employee/add</code>. 3. Set Authorization header to <code>Bearer {{managerToken}}</code>. 4. 4. Set Body to raw JSON with role: "Manager" and teacher_status: "Available". 	ER: Status code is 400 Bad Request . The response body contains an error message indicating a database check constraint

		5. Send request.	violation for the teacher_status field.
EMP-001-TC-007	Unauthorised Access (Attempt by non-Manager)	1. Set method to POST. 2. Set URL to http://127.0.0.1:5000/employee/add. 3. Set Authorization header to Bearer {{teacherToken}}. 4. Set Body to raw JSON with valid, unique data for a new employee. 5. Send request.	ER: Status code is 403 Forbidden . The response body contains an access denied message. <i>(Note: This test will fail until role protection is added to the endpoint.)</i>

- **Results:**

Test Case ID	Result	Status	Notes
EMP-001-TC-001	Similar to ER	PASSED	N/A
EMP-001-TC-002	• 500 Internal Server Error	FAILED	See EMP-BUG-001
EMP-001-TC-003	• 500 Internal Server Error	FAILED	See EMP-BUG-002
EMP-001-TC-004	• 500 Internal Server Error	FAILED	See EMP-BUG-003
EMP-001-TC-005	• 500 Internal Server Error	FAILED	See EMP-BUG-004
EMP-001-TC-006	• 500 Internal Server Error	FAILED	See EMP-BUG-005

EMP-001-TC-007	Assume that the output is 403 Forbidden	FAILED	See EMP-BUG-006
-----------------------	--	--------	-----------------

Feature: Get All Employees

- **Feature ID:** EMP-02
- **Endpoint:** GET /employee/
- **Description:** Verifies the retrieval of all employee profiles.
- **Test Cases:**

Test Case ID	Description	Test Steps	Expected Result
EMP-002-TC-001	Successful Retrieval	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/employee/ . 3. Set Authorization header to Bearer {{managerToken}}. 4. Send request.	ER: Status code is 200 OK . The response body is a JSON array of all employee objects.
EMP-002-TC-002	Unauthorised Access (Attempt by non-Manager)	1. Set method to GET. 2. Set URL to http://127.0.0.1:5000/employee/ . 3. Set Authorization header to Bearer {{teacherToken}}. 4. Send request.	ER: Status code is 403 Forbidden. The response body contains an access denied message. <i>(Note: This test will fail until role protection is added to the endpoint.)</i>

- **Results:**

Test Case ID	Result	Status	Notes
EMP-002-TC-001	Similar to ER	PASSED	N/A
EMP-002-TC-002	Assume that the output is 403 Forbidden	FAILED	See EMP-BUG-006