

Pytest Guide for Internship Placement Manager

This guide provides an overview of how to write and run tests for the Internship Placement Manager API using **Pytest**. Our testing setup is designed to be efficient, reliable, and easy to use.

1.0 Key Concepts

- **Fixtures:** A key feature of Pytest, fixtures are functions that set up a baseline for your tests. They can create a database session, an API test client, or even a test user. They ensure that each test runs in a clean, isolated environment.
- **Dependency Injection:** Pytest automatically detects and injects fixtures into your test functions. You simply define the fixture as an argument in your test function signature, and Pytest handles the rest. For example, by including session in your test function, you automatically get a new database session.
- **In-Memory Database:** For testing, we use an in-memory **SQLite** database. This is a temporary database that exists only in your computer's memory while the tests are running. It is extremely fast and ensures that your tests don't leave any permanent data behind.

2.0 How to Run Tests

From the root directory of your project, use the following commands in your terminal:

- **Run all tests:**

```
pytest
```
- **Run all tests with verbose output:**

```
pytest --verbose
```

This command provides more detailed information for each test, including the name of the test function. It's very helpful for seeing exactly which tests are being run and for debugging failures.

- **Run a specific test file:**

```
pytest tests/test_endpoints.py
```

- **Run a specific test function within a file:**

```
pytest tests/test_endpoints.py::test_students_crud
```

3.0 Understanding the conftest.py File

The conftest.py file contains the fixtures that are shared across all your test files. . This is where the magic happens for our test setup.

- `session_fixture()`: This fixture sets up a temporary, in-memory SQLite database. It creates all the tables needed for your tests and then automatically drops them after the test is complete, guaranteeing a clean state.
- `client_fixture()`: This fixture provides an instance of FastAPI's `TestClient`. It uses the session fixture to override the application's database dependency, so your tests use the in-memory database instead of the live PostgreSQL database.

By using these fixtures, your tests are independent, fast, and reliable.