

How to Run pytest and Selenium Tests - Complete Guide

Part 1: Basic Test Running

Prerequisites - Make Sure You Have Everything Installed

```
bash

# In your mingus project directory, check if you have pytest
pytest --version

# If you get an error, install pytest
pip install pytest pytest-flask requests selenium webdriver-manager
pip install pytest-cov pytest-html # For reports
```

Running pytest Tests (Backend Testing)

1. Basic Single Test Run

```
bash

# Run one specific test file
pytest tests/test_basic.py

# Expected output:
# ===== test session starts =====
# tests/test_basic.py::test_app_exists PASSED [25%]
# tests/test_basic.py::test_client_exists PASSED [50%]
# tests/test_basic.py::test_sample_data_exists PASSED [75%]
# tests/test_basic.py::test_multiple_test_users PASSED [100%]
# ===== 4 passed in 0.12s =====
```

2. Run All Tests in Directory

```
bash
```

```
# Run all tests in the tests/ directory
```

```
pytest tests/
```

```
# Run with more detailed output
```

```
pytest tests/ -v
```

```
# Expected output with -v (verbose):
```

```
# tests/test_api_endpoints.py::test_health_endpoint PASSED
```

```
# tests/test_api_endpoints.py::test_get_user_profile_unauthorized PASSED
```

```
# tests/test_user_registration.py::test_valid_registration PASSED
```

```
# tests/test_profile_completion.py::test_initial_profile_completion PASSED
```

3. Run Specific Test Categories

```
bash
```

```
# Run only API tests (if you used markers)
```

```
pytest tests/ -m "api"
```

```
# Run only fast tests
```

```
pytest tests/ -m "fast"
```

```
# Skip slow tests
```

```
pytest tests/ -m "not slow"
```

4. Generate Test Reports

```
bash
```

```
# Run tests with HTML report
```

```
pytest tests/ --html=reports/test_report.html --self-contained-html
```

```
# Run with coverage report
```

```
pytest tests/ --cov=backend --cov-report=html --cov-report=term-missing
```

```
# Run with both coverage and HTML report
```

```
pytest tests/ --cov=backend --cov-report=html --html=reports/test_report.html --self-contained-html
```

Running Selenium Tests (UI Testing)

1. Basic Selenium Test Setup

First, make sure you have the web drivers:

```
bash
```

```
# Install webdriver-manager (handles browser drivers automatically)
```

```
pip install webdriver-manager
```

```
# Or manually download drivers:
```

```
# Chrome: https://chromedriver.chromium.org/
```

```
# Firefox: https://github.com/mozilla/geckodriver/releases
```

2. Run Selenium Tests

```
bash
```

```
# Run Selenium tests (if you put them in selenium_tests/)
```

```
pytest selenium_tests/ -v
```

```
# Run with visible browser (for debugging)
```

```
pytest selenium_tests/ -v -s
```

```
# Run in headless mode (browser runs in backgrou
```