

ABSTRACT

The purpose of this project was to improve developer efficiency and code quality. Behavioral API tests were implemented in the ASWWU Web server for user authentication and the Elections, Jobs, and Mask projects and development workflow was standardized. Tests were designed to allow project growth as the department adopts new technologies.

INTRODUCTION

Prior to this project tests were performed manually by running a local development server and making requests to it through Postman, cURL, or our frontend interface. While this was sufficient when adding new components to the server with little risk of affecting existing components, it became difficult to make system wide changes with high certainty that nothing was broken across any of our projects.

Behavioral tests were a natural first step in adding verification to the server for existing code. They ensure that the server operates as expected from a user's perspective, regardless of how the API is implemented internally. Tests also work as an important form of documentation for new developers joining ASWWU Web by providing a working demonstration of how the server is expected to behave. This is especially valuable for maintaining departmental continuity from year to year as student developers are hired and graduate.

DATA AND ANALYSIS

Roughly 2500 lines of new code were written for this project; due to refactoring and cleaning of code the overall project grew by roughly 2200 lines. Nearly 700 lines of unused code were cleaned from the project.

In total, 47 tests were written, bringing code coverage up to 58%. Before this project there was no continuous integration in use.  

```
Run bash scripts/run_pytests.sh
===== test session starts =====
platform linux2 -- Python 2.7.18, pytest-4.6.9, py-1.8.1, pluggy-0.13.1
rootdir: /home/runner/work/python_server/python_server
plugins: cov-2.8.1
collected 47 items

tests/aswwu/behaviors/auth/test_auth.py ... [ 6%]
tests/aswwu/behaviors/elections/ballot/test_ballot.py .... [ 14%]
tests/aswwu/behaviors/elections/candidate/test_candidate.py ..... [ 25%]
tests/aswwu/behaviors/elections/election/test_election.py ..... [ 38%]
tests/aswwu/behaviors/elections/position/test_position.py ... [ 46%]
tests/aswwu/behaviors/elections/vote/test_vote.py ..... [ 57%]
tests/aswwu/behaviors/jobs/test_jobs.py ..... [ 80%]
tests/aswwu/behaviors/mask/test_mask.py ..... [100%]

----- coverage: platform linux2, python 2.7.18-final-0 -----
Coverage XML written to file coverage.xml

===== 47 passed in 43.96 seconds =====
```

File	Coverage
base_handlers.py (authentication)	81.05%
route_handlers/elections.py	91.34%
route_handlers/forms.py (jobs)	60.90%
route_handlers/mask.py	67.74%

Other benefits resulting from this project include the addition of deployment and installation scripts, Pipenv, and predefined settings environments to make development and deployment more consistent.

TEST STRUCTURE

The new tests interact with the server through the API only, without preloading information directly into the database, to avoid dependence on any frameworks or libraries that might be replaced in the future.

A behavioral test simulates a set of actions that a user is expected to take. During each of our tests users would need to perform certain actions like signing in and creating a profile. Many components of tests had to be repeated. Subtests were added to perform common actions from multiple independent tests, and worked as parameterized tests to fit the context of their parent tests.

Between each test, the database and any temporary files created during the test are cleaned, and the server and its thread are stopped and restarted.

SUMMARY

This project implemented much needed API testing for ASWWU Web, and covers the department's most significant projects. With the testing framework in place, the team can continue improving test coverage. These tests allow the Web team to safely make necessary upgrades, such as the transition to Python 3 and the addition of new features to benefit the student body.

REFERENCES

Server Repo: https://github.com/aswwu-web/python_server
 Project PRs: https://github.com/stephenermshar/python_server/issues/1
 Codecov: https://codecov.io/gh/ASWWU-Web/python_server
 Postman: <https://www.postman.com/>