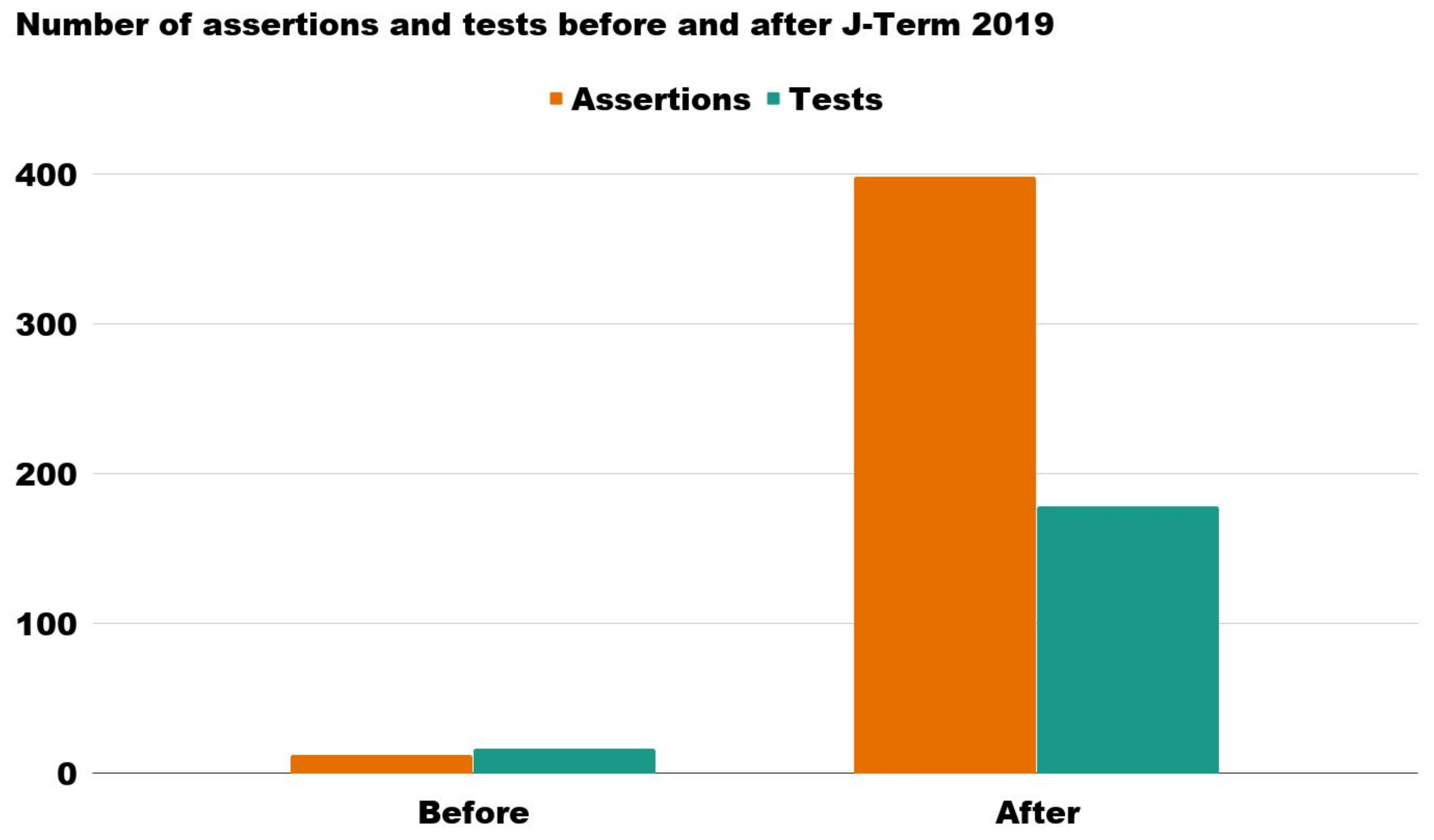
# Programming Contest Server Tests

Daniel Thomas, Andrew Blomenberg, and Dr. Jonathan Geisler

## Why Test?

Tests are often left by the proverbial side of the coding road. Junior Software Engineers are frequently relegated to this ‘unimportant’ task. Despite this preconception, testing is critical to the long term health of a codebase.

This programming contest server, Touche, desperately needed extensive testing. When introduced to Touche we were unaware of what features worked and what did not. Since 2000, Touche has gone through several different overhauls. Despite this, there was no test suite in place to ensure the integrity of the server. Writing and running these tests allows us and future developers to know what works and what doesn’t after each change.



What is the difference between assertions and tests? A test replicates an action from one user, such as logging in or submitting a solution. An assertion is an expectation for something to occur during a test, such as seeing a success message.

## Test Statistics

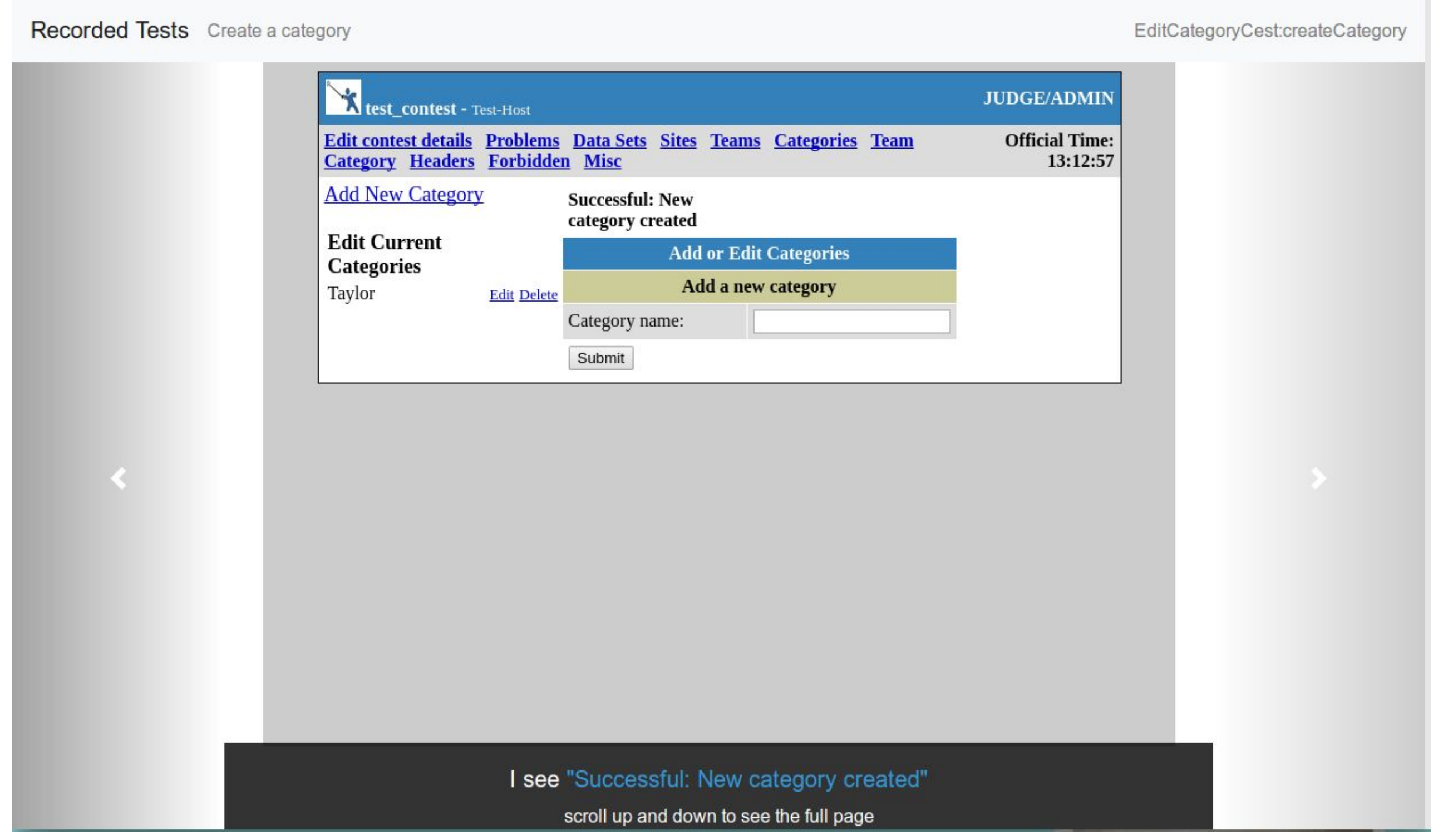
- More than 60 commits to GitHub
- Over 4300 lines of code written
- 40 Minute test runtime
- 398 Assertions
- 178 Tests

## Challenges

The team that had previously worked on these tests left behind little documentation on how the tests should work and what still needed to be tested. We built a new testing structure from the ground up, following good coding practices.

One of four different actors carries out each of the tests, imitating the actions of a user of the software. Creator, Admin, Judge, and Team actors were all integral users whose actions we needed to test. Each of these roles have different responsibilities and actions that they can carry out.

Each actor is a class that inherits from the main actor, Acceptance Tester. If an action needs to be done by every actor, that action appears in the Acceptance Tester’s code.



Official Time: 13:12:57

Successful: New category created

Edit Current Categories

Taylor

Category name:

I see "Successful: New category created"

scroll up and down to see the full page

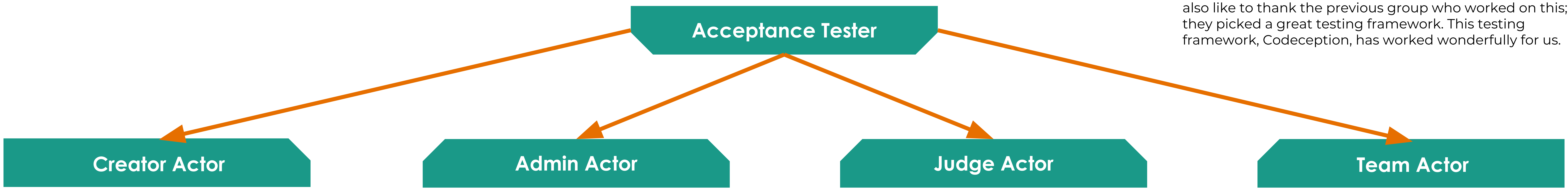
The testing framework grabs screenshots of each step of the test, making it easy to debug each error that occurs.

## Outcomes

We now have a fully functional suite of tests that can be run at the drop of a hat. Because of this coverage, Touche’s code can be changed easily, without concern that something will break. We saw this in action when we were rewriting portions of the codebase. Our main goal was to make this project easier to work on, which we believe we have achieved.

## Acknowledgments

This project would not have been possible without the strong guidance we received from Dr. Geisler. We would also like to thank the previous group who worked on this; they picked a great testing framework. This testing framework, Codeception, has worked wonderfully for us.



### Acceptance Tester

### Creator Actor

The Creator Actor has two essential functions: to create and delete tests.

### Admin Actor

The Admin Actor takes care of various contest details, such as managing account credentials, contest languages, and contest duration.

### Judge Actor

The Judge Actor can judge the submissions from each team, view standings, respond to clarifications, and also start the contest.

### Team Actor

The Team Actor can submit solutions to different problems, view standings, and request clarifications.