



Coveros Increases Software Quality

with Automated Functional
and Security Testing

CASE STUDY





CASE STUDY

Coveros Increases Software Quality with Automated Functional and Security Testing



Homeland Security

The US Department of Homeland Security Immigration and Customs Enforcement (ICE) enforces federal laws governing border control, customs, trade, and immigration to promote domestic security and public safety. ICE's Enforcement and Removal Operations (ERO) division enforces the nation's immigration laws as it identifies, apprehends, detains, and, when necessary, removes immigrants living in the United States illegally.

CHALLENGES

- Testing was performed manually, leading to longer development cycles
- No unit tests were being written or executed against the code
- There were limited test environments and testing infrastructure

ICE ERO was experiencing challenges keeping their testing cycles in sync with the pace of agile development. This was because the tests were executed manually, often at the tail end of development sprints and under limited time constraints. Consequently, both new feature validation and regression testing often extended beyond the duration of the sprints and into the next sprint. Testing for a current sprint's work often took place weeks after a story was completed. This was

further compounded in cases where bugs were found, because these had to be corrected before release, leading to even more long delays before the application could be released.

Another challenge was that ICE ERO's applications didn't incorporate any forms of testing other than at the UI level, which meant that they relied solely on the browser-based tests to catch all defects and ensure the overall quality of the application. As a part of its software modernization initiative, the agency sought to improve the speed and quality of their testing through automation.

SOLUTION

Coveros performed an initial assessment of ICE ERO's testing process and infrastructure, then developed solutions that included both short- and long-term recommendations. Coveros then worked side by side with ICE teams to implement these recommendations, constantly evaluating the effectiveness of the approach to ensure that we were achieving positive changes.

We started by identifying the root causes for the bottlenecks, one of which was the lack of a defined continuous integration (CI) process for newly developed code. Rather than pushing out individual features to test teams as they were created, developers would often wait until all tasks were completed before promoting a build to test. One reason for this was a lack of

CHALLENGES

- Manual testing
- Long development cycles
- No unit tests
- Limited test environments and infrastructure

SOLUTIONS

- Automated test suites
- Continuous integration process
- Custom tooling frameworks
- Disposable test environments in the cloud



U.S. DEPARTMENT OF
HOMELAND SECURITY



Coveros worked side by side with ICE teams to implement these recommendations, constantly evaluating the effectiveness of the approach to ensure that we were achieving positive changes.

CASE STUDY

Coveros Increases Software Quality with Automated Functional and Security Testing



understanding of the drawbacks of this approach, but another was due to more practical limitations: All ICE teams shared one test environment, so deployments had to be coordinated among teams in order to reduce the risk of conflicts.

Consequently, one of our first actions was to engineer a solution to introduce a CI process, focusing on one application to serve as a proof of concept for other applications to follow. We started with changing the way Jira tickets were written to include testing as a part of the definition of done for each task. Then, using Jenkins to orchestrate the process, we built a CI pipeline that executed unit tests against, compiled, and deployed all newly committed code. This enabled developers to quickly and frequently push new code and gain quick feedback on the quality of the build.

We then addressed the test environment issues by developing disposable environments using cloud technologies. These environments, which can be created on demand by either developers or testers, provided individual testers with isolated sandboxes where they could quickly validate features without interference.

Coveros then worked with ICE test teams to build a suite of automated tests utilizing our custom-built testing framework, Selenified. Our tests fully exercised ICE's applications at various levels, including the API, database, and front end. We ensured that these tests

were properly integrated into the CI pipeline and were automatically triggered with each new build.

The result was that each tester could pair with a developer to ensure every new functionality was properly tested in a feature branch before being integrated and promoted to the next environment for regression and integration testing. This significantly reduced the lag between feature development and testing for the application.

We improved upon these changes by incorporating quality gates into the CI pipeline, which required passing unit and functional tests in order to promote a build. We also introduced additional tools to the pipeline, such as SonarQube, to provide metrics on unit test coverage, as well as HP Fortify to help detect potential security vulnerabilities.

TECHNOLOGY SOLUTIONS

Test Automation

- Front-end, API, and database testing with the Selenified testing framework
- Unit tests
- Continuous integration
- Jenkins

CASE STUDY

Coveros Increases Software Quality with Automated Functional and Security Testing



BUSINESS VALUE

Coveros helped ICE ERO alleviate the challenges of their existing testing practices by implementing an optimum combination of process changes and tooling. We leveraged automation at various levels of the development process, including testing in order to increase the productivity of test teams and improve collaboration with developers. By introducing more layers of testing at the back-end and API levels, we uncovered

defects that had previously been undetected through manual UI testing alone. By automating a majority of regression tests, we significantly reduced regression test cycle time while producing detailed test reports that gave ICE's management confidence that their systems were being tested thoroughly. Finally, we shared our approach with other ICE test teams and helped to drive the push for automation across the entire agency by demonstrating its effectiveness.

By introducing more layers of testing at the back-end and API levels, we uncovered defects that had previously been undetected through manual UI testing alone.

CONNECT WITH COVEROS



coveros.com



info@coveros.com



929.341.0139



twitter.com/coveros



linkedin.com/company/coveros