Automation Scripts Conversion from Selenium to Test Complete

Jagan Mohan Rao Doddapaneni
Jaganmohanrao.d@gmail.com

Abstract:

In modern software quality assurance, regression testing is critical for ensuring the stability and reliability of systems undergoing significant transformation. This paper explores the transition of a 1,000+ scenario regression test suite from Ruby and C# using Selenium to Python with Test Complete. The migration was necessitated by the incompatibility of the existing scripts and tools with Oracle Forms, introduced after transitioning presale activities from AS400 to Oracle. This case study delves into the technical challenges faced, the decision-making process, and the benefits achieved through the migration, offering insights into addressing compatibility issues and optimizing automation frameworks.

Introduction:

Regression testing is a cornerstone of software quality assurance, ensuring that new changes or updates do not disrupt existing functionalities. In this case study, the regression suite of over 1,000 scenarios was initially developed using Ruby and C# with Selenium, a robust tool for web-based testing. However, with the migration of presale activities from AS400 to Oracle, the test scripts became incompatible with the new Oracle Forms platform. Additionally, Selenium lacked the necessary support for Oracle Forms, creating a critical bottleneck in the automation process.

This paper provides a detailed account of how the team addressed this challenge by migrating the entire regression suite to Python scripting using Test Complete, a tool compatible with Oracle Forms. It highlights the technical considerations, migration strategies, and benefits achieved, presenting a comprehensive roadmap for similar transformations in quality assurance practices.

Challenges:

1. **Script Incompatibility:**

   ○ The existing regression scripts written in Ruby and C# were tightly coupled with the AS400 system workflows and Selenium's capabilities. Oracle Forms introduced a new set of interfaces that rendered the scripts obsolete.

2. **Tool Limitations:**

   ○ Selenium, while powerful for web-based testing, lacked native support for Oracle Forms. This limitation necessitated the exploration of alternative tools capable of addressing the specific needs of Oracle-based platforms.

DOI: 10.55041/IJSREM9574

3. **Migration Complexity:**

   ○ Rewriting over 1,000 test scenarios in a new scripting language required significant time, resources, and technical expertise. Ensuring consistency and accuracy during the migration was a critical challenge.

4. **Learning Curve:**

   ○ Adopting Python and TestComplete introduced a learning curve for the QA team. Training and upskilling the team were essential to leverage the new tools effectively.

5. **System Downtime:**

   ○ The migration process created a temporary gap in automated regression testing coverage, increasing reliance on manual testing and posing risks to delivery timelines.

Technical Solution:

The following steps were taken to address these challenges:

1. **Tool Selection:**

   ○ After evaluating several options, TestComplete was chosen for its comprehensive support for Oracle Forms and integration with Python, a versatile scripting language.

2. **Script Migration:**

   ○ The existing Ruby and C# scripts were analyzed for functionality, modularized, and rewritten in Python to ensure compatibility with TestComplete.

3. **Framework Development:**

   ○ A new automation framework was developed in Python, incorporating best practices such as modularization, reusable components, and detailed reporting capabilities.

4. **Team Training:**

   ○ The QA team underwent extensive training in Python scripting and TestComplete functionalities to build proficiency and ensure a smooth transition.

5. **Phased Rollout:**

   ○ The migration was executed in phases, prioritizing high-impact scenarios and gradually extending coverage to the entire regression suite. Parallel manual testing ensured uninterrupted quality assurance during the transition.

Key Achievements:

1. **Enhanced Compatibility:**

   ○ The new regression suite, developed in Python, seamlessly integrated with Oracle Forms, enabling comprehensive test coverage.

2. **Improved Efficiency:**

   ○ The modular framework reduced script execution time and simplified maintenance, enhancing overall efficiency.

3. **Future-Proofing:**

   ○ Python's versatility and TestComplete's robust features positioned the QA team to adapt to future requirements and system upgrades.

4. **Cost Savings:**

   ○ Automation of previously manual processes reduced testing cycles and resource utilization, resulting in significant cost savings.

Technical Skills Utilized:

1. **Programming Languages:** Python, Ruby, C#
2. **Automation Tools:** Selenium, TestComplete
3. **Testing Methodologies:** Regression Testing, Automation Framework Development
4. **Platforms:** Oracle Forms, AS400
5. **Integration Tools**: Test management and reporting solutions

---

Conclusion:

The migration of the regression suite from Ruby and C# using Selenium to Python with TestComplete was a transformative initiative for the QA team. By addressing compatibility issues and optimizing the automation framework, the project ensured seamless testing coverage for Oracle Forms. This case study underscores the importance of adaptability, strategic tool selection, and team collaboration in overcoming technical challenges and achieving quality assurance excellence.

Acknowledgments:

This successful transition was made possible through the collective efforts of the QA team, developers, and project stakeholders. Their dedication, technical expertise, and collaboration were instrumental in achieving the project's objectives.

References:

1) "Selenium WebDriver 3 Practical Guide" by Unmesh Gundecha and Satya Avasarala Published in 2016.

2) "TestComplete Cookbook" by Gennadiy Alpaev Published in 2015.

3) "Python Testing with Pytest" by Brian Okken Published in 2017.

4) "Automated Testing of Oracle Forms-Based Applications" (IEEE Paper)

5) "The Role of Automation in Agile Development" (Journal of Systems and Software)

6) "Migrating Legacy Systems to Modern Platforms: A QA Perspective" (ACM Digital Library)

7) Selenium Official Documentation (selenium.dev)

8) TestComplete Automation Resources (SmartBear)

9) Python Automation Tutorials (Real Python)

10) "Modernizing Legacy Systems with Test Automation" (SmartBear Whitepaper)

11) "Enhancing QA Efficiency Through Tool Migration" (Capgemini Research)

12) "Continuous Integration and Continuous Delivery with Jenkins" by Sander Rossel Published in 2017.

13) "Agile Testing: A Practical Guide for Testers and Agile Teams" by Lisa Crispin and Janet Gregory Published in 2009.