# Contents

Executive Summary....................................................................................................................................03  
Need for Test Automation...........................................................................................................................04  
Overcoming Limitations of Open Source Tools..........................................................................................04  
Test Automation- Myths and Facts.............................................................................................................05  
Building an Automation Framework............................................................................................................08  
Recommended Architecture and Features.................................................................................................09  
Impact of Test Automation across Industries..............................................................................................10  
AccuRate for Test Automation......................................................................................................................11  
Case Studies..............................................................................................................................................12  
Case Study - 1: Educational Institution.......................................................................................................12  
Case Study - 2: Heavy Equipment..............................................................................................................13  
Conclusion..................................................................................................................................................14
Executive Summary

Since more and more enterprises are focusing on productivity improvement, it is important that due focus is given to Test Automation. In the current context of Agile and DevOps, as the product is evolving, features are getting added in every sprint, followed by regular releases. During each release, it is important to ensure that the features implemented in the previous releases are not broken. This emphasizes the importance of test automation in continuous integration and continuous deployment model.

This paper explains the importance of test automation, challenges when using available open source tools and an approach to overcome these limitations by building a customized framework. It addresses the challenges and benefits of using the special software which helps to execute tests, report outcomes and compare results in an efficient manner. The paper also provides details on how the test automation solution suite, AccuRate, has helped leverage open source testing tools to release high quality applications to one of the higher education providers in the United States and one of the largest crane rental companies in the United States.
Need for Test Automation

Any software application has to be tested against the customer requirement specification to ensure that the application meets the customer’s (user) expectations and all the features and functionalities are working fine before it is released in the market. Software has to be tested for all critical use cases.

For large scale applications with frequent releases, the application needs to be regressed before every release to ensure that it is defect free and all the features implemented in prior releases are intact with the recent code change. So regression test suite has to be prepared carefully and then executed for each release. Regression suite is the ideal candidate for functional automation. Once the automated scripts have been developed, the tests can be performed by developer or tester. It can also be configured to run overnight without manual intervention.

In general manual testing is tedious, and it gets cumbersome when tests are repetitive. In case of web applications, the scripts have to be run on multiple browsers. In case of responsive web, the same test has to be repeated on desktop browsers and various device browsers on different platforms. In case of mobile applications, the same set of tests has to be repeated on various devices, OS platforms (Android, iOS, Windows) and many versions, various mobile browsers and mobile devices with different screen sizes.

Automating the tests will reduce the testing effort and help improve productivity. Automation is most effective whenever the tests are highly repetitive in nature. It will, also, reduce the manual errors and help to improve test coverage. Agile development demands for minimum time to market. Automation can play a pivotal role in achieving this without major cost overruns.

Overcoming Test Automation Challenges

There are certain challenges faced by enterprises in test automation.

- Rapid technology changes in mobile and web development make test automation complex. Many a times test automation tools have to be upgraded to synchronize with new technologies.

- Automation engineers encounter various external factors that make automation more challenging like tool limitations, ever growing number of target devices, multiple browser versions etc.

- All the functional test automation tools are dependent on the front-end technologies and User Interface (UI). Changes to the UI will cause a ripple effect in the automation scripts.

In the above cases, test automation becomes a tedious job and needs a lot of effort to build and maintain the scripts, hence, could cause cost overruns. Most of the existing popular automation tools are unable to provide test automation coverage, single-handedly, for all the given business contexts. Let us take a closer look at the two scenarios below:

- In case of responsive web applications, the tests need to be performed on desktop browsers and mobile browsers, on various devices, with different screen sizes, platforms and OS versions. In conventional automation, we need to use Selenium for automating desktop browsers and Appium for automating mobile browsers.
In case of mobile apps, the test scenarios need to be repeated in Android, iOS, phones and tablets, various OS versions and mobile browsers. It is cumbersome to write scripts for each platform and screen sizes. In conventional automation, the scripts have to be written, separately, for iOS, Android and mobile browsers.

An ideal automation framework, that leverages the capabilities of various open source tools and frameworks, will help us to overcome the above mentioned challenges. The framework has to be developed in such a way that it should be reusable for multiple applications of the same business unit or across the organization for a better cost-benefit ratio and faster ROI.

**Test Automation - Myths and Facts**

There are certain myths of test automation that impede the expansion of automation. These must be addressed in order to ensure proper development and help the testing teams to, correctly, apply them, wherever required, in the most cost-effective and efficient manner.

**Myth #1: Test Automation is costly and a lot of effort is required to learn and develop the scripts.**

**Fact:**

a) **Test automation with appropriate framework improves productivity and reduces effort.**

Let us look at the standard effort for executing 100 test cases. On an average, the time required for testing an application with 100 test cases manually on one browser or device is 1 day. It will take 4 days to automate 100 tests with a customized keyword driven framework.

Now, let us consider the 2 scenarios, mentioned in the previous section (Overcoming Limitation of Open Source Tools) and a situation where customized framework is available.

**Scenario 1 - Responsive web:** The tests to be run on 4 desktop browsers (IE, Safari, Chrome, Opera), on mobile browsers - Android (at least 1 phone and 1 tablet), on mobile browsers – iOS (at least 1 phone and 1 tablet).

**Scenario 2 - Mobile app:** On Android (4 devices with different screen size/OS version and 2 tablet with different screen sizes), iOS (4 devices with different screen size/OS version and 2 tablets with different screen sizes).

You can see the factors in the table below, for the first cycle, which involves script development effort, as well. The time required to develop the scripts is 32 hours and it takes 3 hours to execute it.
Now, let us look at the total effort needed for 10 sprint cycles.

In case of automation, 32 hours is needed for script development, 1 hour each for maintenance (assuming that there will be UI changes) and 3 hours to run the tests.

### Table: Effort and Execution Time for Different Types of Testing

<table>
<thead>
<tr>
<th>Types of application</th>
<th>No. of browsers and devices</th>
<th>Manual testing effort (in hours)</th>
<th>Automation testing with a keyword driven framework (in hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Responsive Web</td>
<td>8</td>
<td>64</td>
<td>35</td>
</tr>
<tr>
<td>Mobile Apps (Android, iOS)</td>
<td>12</td>
<td>94</td>
<td>35</td>
</tr>
</tbody>
</table>

Now, let us look at the total effort needed for 10 sprint cycles.

In case of automation, 32 hours is needed for script development, 1 hour each for maintenance (assuming that there will be UI changes) and 3 hours to run the tests.

### Table: Effort and Execution Time for Different Types of Testing

<table>
<thead>
<tr>
<th>Type of testing</th>
<th>Effort (in hours)</th>
<th>Execution time</th>
<th>Remarks</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Testing (Responsive Web)</td>
<td>640</td>
<td>640</td>
<td>Effort and time are same.</td>
</tr>
<tr>
<td>Manual Testing (Mobile Apps)</td>
<td>940</td>
<td>940</td>
<td>Effort and time are same.</td>
</tr>
<tr>
<td>Automated Testing (Responsive Web)</td>
<td>42</td>
<td>24</td>
<td>Execution does not require manual effort.</td>
</tr>
<tr>
<td>Automated Testing (Mobile Apps)</td>
<td>42</td>
<td>36</td>
<td>Execution does not require manual effort.</td>
</tr>
</tbody>
</table>

**b) Reduced cost**

Considering the above table, effort gets reduced when tests are automated. This will reduce the cost, significantly. However, there is an initial investment of building the framework.

The ROI of the framework can be achieved faster if it is done for a business unit (large organization) or organization. Long term benefits will be significant since each application, in a business unit/organization, will have multiple releases and multiple test cycles.

**c) Improved quality**

Test automation reduces manual errors. It provides the flexibility to perform regression testing, as many times as you want, whenever there is a code change. Automation helps to improve the test coverage. All these eventually help to improve the quality.
d) Improved test coverage

Test coverage can be achieved by automating tests at various levels. A framework, which enables automation at multiple levels, will add greater value.

- Acceptance Testing - In the BDD context, acceptance tests can be automated. The BDD tests will be written in business context by a business analyst. This can be automated, using open source tools like Cucumber, which uses English language to write tests. A framework can incorporate tools like Cucumber, to enable automation of acceptance testing.

- Functional Testing - Regression suite is the ideal candidate for automation to ensure that the features and functionalities are defect free. A framework, with combination of tools like Selenium, Appium, and Protractor will enable functional automation of web and mobile.

- API/Web Service Testing - API/Web service level testing will help in early identification of defects. API/Web Service testing can be automated, using open source tools like Jmeter. Jmeter is flexible and has got various other capabilities. It can be easily integrated with Selenium or Appium.

- Unit Testing - Unit testing is performed by developer and there are various tools, like jUnit, which are available for automating unit test cases.

When we look at the effort, time, cost and quality, an automation framework provides great value at enterprise level.

e) Early detection of defects

Different levels of testing will enable to identify defects much early in the software development process. Unit testing and Web Services/API testing will enable to identify the defects in the development phase of the application.

Myth #2: Tool support is limited and 100% automation is not possible.

Fact: All open source tools are generic and may not support all the use cases. But open source tools are highly extensible and a customized framework, built on top of it, can overcome these challenges. The framework should build a library to overcome tool limitations, which can be used across enterprise.
Myth #3: Script maintenance is a tedious task.

Fact: An automation framework, with extensible architecture, can reduce the script maintenance. Framework should separate the object repository from scripts. Changes, in the UI components, should not enforce changes to the scripts. The change can be made in object repository, which is easy.

Myth #4: It is difficult to identify the right tool and strategy for automation.

Fact: The first and foremost task is to list the requirements, which includes the front-end technology, level of automation, number of releases, type of application etc. Also, you need to look at the type of applications and technologies across the business unit. Then, identify the major limitations of prominent open source tool and plan the framework, based on the evaluation.

Building an Automation Framework

Given the fact that most of the open source automation tool has some limitations, looking from business and technology perspective, it is important to have an automation framework in place, to succeed with test automation.

The test automation framework is focused on reusability, keyword driven approach, data driven approach, to support BDD, and integrates with various open source tools and frameworks, to bring in the required capabilities for automation.

Remarkably, the framework brings web, responsive web and mobile, under a single umbrella. This framework enables us to write scripts once and run on desktop browsers, mobile browsers and mobile devices (Android and iOS), both native and hybrid apps.

Design considerations for the customized framework:

1. Understand your business and technology context across enterprise that can be automated.
2. Select the open source automation tools that suit the given business and technology context.
3. Integrate different tools to satisfy all the requirements.
4. Decide the common features and functionalities that need to be developed in-house.
5. Ensure reusability of the scripts, across various environments.
Recommended Architecture and Features

Considering the fact that no single tool, available, can help to do E2E automation, an automation framework, relevant to the given business and technology context, have great value.

This section explains the recommended* architecture (at high level) and key features for a test automation framework.

*This is a recommended architecture based on experience building test automation framework and not limited to only these integration, technology and open source tools.

The key features of automation framework (web, responsive web and mobile) include:

- Reusability of scripts across platforms, browsers, devices with various screen and sizes.
- Reusability of common features, functionalities and actions for web and mobile.
- Data-driven, with a common input source.
- Keyword-driven approach to reduce scripting.
- Continuous integration with CI servers.
- Parallel execution on multiple browsers and mobile devices.
- Reporting and dashboard.

A Whitepaper by RapidValue Solutions
Separate object repository to ease maintenance.
- Interface to integrate various tools and frameworks.
- Integrate with defect management tool.
- Integrate with device cloud or tools which provide browser simulation.
- Error handling and error logging.
- Exception handling.

**Impact of Test Automation across Industries**

Testing your web/mobile applications is a must, before it is made live for the user, for any industry. Listed are examples of some industries which are highly impacted. Automation plays a vital role for testing these applications.

**Healthcare**

Irrespective of a patient centric application, payer centric or provider centric IT applications or connected healthcare, most of the applications will be integrated. In case of System Integration Testing (SIT), various applications need to be integrated and tested. All these applications will be developed, separately, and may use different technologies.

An automation framework will help to achieve SIT automaton at a greater level. The same automation framework can be used for functional automation of individual application. Quality is most important in healthcare application, since an error in software can affect the life of a patient. So, test automation has a significant role to play, since it provides flexibility to run tests, many times, to ensure quality.

**Banking**

Irrespective of investment or core banking, there will be a large number of applications that are integrated to achieve the end goal. Some of them will be web and some of them will be responsive web or mobile. A single automation framework can have a greater effort reduction. Automation can be a solution to improve quality and reduce manual errors.

**Education**

E-learning and virtual classrooms enable learning from anywhere as per the user’s convenience. In order to have seamless learning experience, quality of the application is treated as a critical factor. Automated tests improve the quality.
AccuRate for Test Automation

Based on our experience working with open source tools, RapidValue has built a test automation solution suite to fast-track app release.

AccuRate is an all-in-one solution for test automation across various mobile platforms (iOS, Android, Web) and desktop browsers, in real-time. It includes 600+ pre-built functions for mobile and web automation leveraging open source tools like Appium, Selenium and Protractor. AccuRate provides rich features for test execution, test control and test management with data-driven and keyword-driven testing capabilities. AccuRate helps enterprises accelerate their app release while realizing significant business benefits.

Solution Benefits

- Develop scripts, once, and run on desktop browsers and mobile devices with different platforms, OS and different screen sizes (reusability of scripts).
- Keyword-driven approach to reduce script development time by 50-60%.
- Supports both mobile and web.
- Supports continuous integration.
- Integrate with various device cloud, browser simulation tools, defect management tools, etc.

Retail (E-commerce and M-commerce)

Customers largely use e-commerce and m-commerce applications. Any error, whether affects the purchase of a product or security of the payment information is critical. This will impact business revenue and lead to customer dissatisfaction. The release cycles are frequent and test automation will save time, cost, effort and improve quality.
Case Studies
AccuRate has helped to test automate many projects at RapidValue. In this section, the paper provides details of two projects and how AccuRate is used to release high quality applications for the customers.

Case Study - 1: Educational Institution

The Customer
One of the largest higher education providers in North America.

Services
A Regression suite was developed using automation framework. The overall duration was 4+ years with a release cycle of every 2 weeks.

Results

<table>
<thead>
<tr>
<th>Type of testing</th>
<th>Initial effort + maintenance effort (in hours)</th>
<th>Time taken for single release on 8 devices (in hours)</th>
<th>Total effort for a year-2 releases per month (in hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Regression</td>
<td>NA*</td>
<td>32</td>
<td>768</td>
</tr>
<tr>
<td>Automated Regression</td>
<td>360</td>
<td>3</td>
<td>432</td>
</tr>
</tbody>
</table>

* NA- Not Applicable
Case Study - 2: Heavy Equipment

The Customer
One of the largest crane sales rental co. in the United States and the world's largest Terex crane dealer.

Services
A Regression suite was developed using automation framework for the customer. The overall duration was 1+ year with a release cycle of every week.

Results

<table>
<thead>
<tr>
<th>Type of testing</th>
<th>Initial effort + maintenance effort</th>
<th>Time taken for single release (in hours)</th>
<th>Total effort for a year-weekly release (in hours)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Manual Regression</td>
<td>NA*</td>
<td>12</td>
<td>576</td>
</tr>
<tr>
<td>Automated Regression (60% - 100 Test Cases)</td>
<td>50</td>
<td>3</td>
<td>194</td>
</tr>
</tbody>
</table>

* NA - Not Applicable
Conclusion

Automation engineers are dedicated to design sophisticated test frameworks and are skilled software developers. They automate, maintain, and run test cases. The future trend with regard to test automation is likely to provide more structured representation of tests and high-level representation of tests.

Automated testing tools will become less complex, easy to use and more adaptive. Automated testing has a long way to go. Test automation is able to improve the development process of a software product. It can run fast and frequently, which characterizes the cost-effective feature for the software products, along with the characteristic of a long maintenance life.

This paper is written by Vidya VG, Practice Manager - Testing Services, RapidValue Solutions.

If you’d like to learn more on our test automation solutions and services, please write to us at... contactus@rapidvaluesolutions.com. We’d be happy to hear from you.
About RapidValue

A global leader in digital transformation for enterprise providing end-to-end mobility, omni-channel, IoT and cloud solutions. Armed with a large team of experts in consulting, UX design, application development, integration and testing, along with experience delivering projects worldwide, in mobility and cloud, we offer a wide range of services across industry verticals. We deliver services to the world’s top brands, fortune 1000 companies, Multinational companies and emerging start-ups. We have offices in the United States, UK and India.

www.rapidvaluesolutions.com
www.rapidvaluesolutions.com/blog
+1 877.643.1850
contactus@rapidvaluesolutions.com