



WHITE PAPER

ZERO TO CONTINUOUS TEST WITH MAQS

Magenic's Automation Quick Start (MAQS) is a modular framework that is written and maintained by the engineers that use it day in and day out. It can be used as your primary automation framework or as a supplement to other automation frameworks.

See how two organizations, one with failed automation and one without any automation, were able to leverage MAQS to quickly start benefiting from continuous test automation.

Troy Walsh

Practice Lead – Quality Assurance



Struggle of Two Clients

Magenic was engaged by two clients to help build out their automation capabilities. One client started with no automation and the other had automation but it was fragile, could not be run on demand, and was providing no value. Essentially, both clients needed to start from scratch.

In both cases the testing efforts were taking far too long and were not providing the timely feedback the projects needed. Executing regression was taking nearly an entire sprint and bugs found during regression were often introduced weeks or months earlier. This all led to significant technical debt and poor product quality.

Start of the Engagement

At the beginning of the first client's engagement, Magenic started with an assessment. Based on those findings, the client decided to use Magenic's open source MAQS framework because it made getting started easy, included several high value features, and it seamlessly integrated with their build and test management infrastructure. MAQS basically provided all the features they wanted and needed without requiring a prohibitively long and expensive buildout. Within days, the client had a fully functional automation smoke test suite.

The second client had decided to go with MAQS before Magenic even started on the project. Lessons learned from prior failed automation endeavors lead them to believe MAQS would be a good fit for their organization and infrastructure. With the second client, the QA team hit the ground running and started creating automation on day one.

Build\Deployment Integration

As the first project gained steam, a robust infrastructure and uniform processes were enacted, including automated environment and test data setup. Test automation leveraged this default test data and augmented it with additional web service and database data setup, using MAQS. As the process matured, automation shifted to in-sprint. This allowed automated smoke and feature tests to be executed with every CI build, along with the development unit tests. With every release to QA, the full regression suite was run as part of the deployment. In addition to smoke, feature, and regression tests, performance tests were also implemented to run daily to assure the application remained performant. The automation and development code lived in the same repository, allowing the automation to remain in sync with development as it was propagated to higher environments. The build server made simple configuration overrides, such as the base website URL, to configure the tests for these higher environments.





The second project started very quickly and, with the help of the build master, a small set of smoke tests were created and integrated into the continuous integration build on the very first day. As with the other project, Magenic built a robust infrastructure and implemented uniform processes. The test environments were loaded with predefined test data on every deployment and automation was driven based on that test data. This allowed Magenic to build out an extensive test suite very quickly while minimizing maintenance. Much like the other project, MAQS was used to create additional test data via database and web services. Once again the automation code lived in the same repository as development and was dynamically configured by the compile server.

Reporting

The primary value of testing is information. If the results of your automation and/or manual testing efforts are not reported, then they provide little to no value. Far too often reporting is either forgotten or not prioritized. This is a key reason logging is a core capability of MAQS.

Reporting on both projects was built out in phases. In the first phase, Magenic leveraged the compile server to build reports; these reports give you simple pass/fail numbers via email on every test run. This means your team does not need to go looking for test results, the results are automatically emailed to them immediately after every run. The second phase was team dashboards. The dashboards give you a high-level view of your test suite, allow you to quickly focus on trouble areas, and allow the entire team/project to see the state of testing in one convenient location. The third phase, which only one of the clients reached (the other was switching test management systems), was to automatically upload the test results to the client's test management system. This allowed manual and automated test results to be reported from one place and allowed for much better reporting and coordination.





Take away

Both clients saw a significant reduction in regression time and improvements in the quality of their overall products. Bugs that in the past would have gone weeks or sometimes months without being discovered were now being found as early as the development phase. This improvement has helped reduce the technical debt of both projects and helped shorten their time to market.

The value that MAQS returned on these projects was two-fold. First, it provided a ready-made framework and infrastructure that allowed them to start creating maintainable automation on day one. When creating an automation framework, there are many things like logging, configuration, setup/teardown, tooling, and dependencies teams need to think about and MAQS takes care of that for you. Second, it provided flexibility. MAQS supports web browser, web service, database, mobile, and email automation. It can run on multiple build servers like VSTS, Jenkins, and Bamboo. On top of all that, it is built, maintained, and used by professional automation engineers – the people that need it to work. MAQS allows its users to quickly start creating automation without the expensive, time-consuming practice of creating a framework.



About Magenic

Magenic is a leader in business technology consulting. We understand the challenges companies are facing and apply the right technology to transform their business.

Visit us at magenic.com or call us at **877.277.1044** to learn more or to engage Magenic today.



This white paper is for informational purposes only. Magenic Technologies, Inc., makes no warranties, express or implied, in this summary. Other product and company names mentioned herein might be the trademarks of their respective owners. © 2017 Magenic Technologies Inc. All rights reserved.

