CASE STUDY

Test Automation for digiChart

About the Company

digiChart introduced the first OB/GYN-specific EHR in 2002, capitalizing on the expertise and guidance of more than 150 contributing obstetrician-gynaecologists. digiChart later expanded to offer solutions to enhance and support the overall management of the practice. Today, digiChart enjoys a strong and growing base of providers serving more than 1,000 OB/GYN clinicians nationwide. digiChart is singularly focused on providing solutions that support the clinical, financial, and administrative functions of OB/GYN physician practices.

Business Needs

digiChart is rapidly building new products and modifying existing products to introduce new features that support the changes in the healthcare industry as it relates to ICD-10, payment reforms, meaningful use and revenue cycle management. The software development team has been making changes that cut across the layers of products, adding features that widely affected and changed the current products. The QA team needed to keep with the rapid changes made by the development team - ensuring that the new enhancements worked to the requirements, and that the existing functionality continue to work as before. And this needed to be done on a timely fashion, keeping with the fast sprints by the development team.

Challenges

Many of digiChart's products were built as traditional client/server, installed executables. They were built using Microsoft's WPF technologies. No open source tool is available to support this WPF technology. digiChart supports a variety of Windows operating system versions. The testing needed to ensure that the installed application works on all these variants and that updates from on product version to another smooth and the install process was trouble free. Many customers use multiple products from EHR suite of digiChart and this increases the complexity in testing the application interactions including data flow between them and different user role/permissions. For each incremental version release, a through regression testing of the application for compliance with standards related to ICD10 and meaningful use, etc. need to be done.

Tools Used

- ✓ **Test Rail -** was used as a test management tool to create test plan, test suite for each release, and update test cases.
- ✓ **TFS –** repository for all automation scripts and harnesses
- ✓ **Silk Test -** automation tool with C# and NUnit Framework.

Solution

Trigent evaluated and selected Borland's Silk Test as it supported a broad range of technologies from web to rich client/server applications. Trigent also created a successful proof-of-concept to verify the suitability for the unique needs of digiChart. Trigent developed automation scripts and configured them to run on multiple OS to test compatibility. Trigent developed detailed test suite to cover all possible flows, and created test data to replicate different user roles for validating seamless data flow and integrity. A comprehensive regression test suite was built and automated using Silk4Net (Silk Test) tool and NUnit unit testing framework. An automated test suite with about 150 complex test scenarios were developed to validate complex functionalities across many components of the product.

Benefits

With a reliable, automated test suites, Trigent was able to support digiChart's rapid development cycle. The development team was able to add new functionality with a confidence of no regression. The automation regression test suite increased the test coverage to include most business critical usage flows and reduced the effort needed towards bug fixes by early detection.

digiChart was able to increase their target market by supporting a variety of operating system, hardware configurations and browsers.

The automated regression suite reduced QA cycle time from few days to few hours and allowed testers to spend more QA time on other focussed areas like ICD10 implementations.