SUCCESS STORY.
Test Automation Framework For Healthcare Analytics Application

About the Client

The client, subsidiary of a Fortune 100 Technology conglomerate, is a leading provider of healthcare data management, analytics, decision support, process automation and related information technology solutions. The solutions include customized IT, HEDIS® compliance reporting and physician performance measurement, value measurement, fraud, and waste and abuse. These solutions help governmental and commercial healthcare payers improve patient outcomes, enhance market position and reduce costs.

Business Challenge

The client's software applications serve a diverse mix of commercial and government clients, from large health care plans, major managed care organizations to leading federal agencies and contractors. The software applications get constantly enhanced and upgraded with new version releases every year.

Each new version release involved hundreds of hours of regression testing – a tedious, time consuming process where hundreds of data points would be manually inputted to check for functional or system errors. The testing was conducted by domain experts with knowledge in health care claims processing. The client considered automating the regression testing activity in the past using QTP; however the size and complexity of the application were a deterrent.

In 2010, the client decided to take up the test automation exercise on priority. The client selected Silicus for executing the QTP automation testing project on the back our demonstrated experience using QTP test automation package, and a favorable completion timeline despite the size and complexity of the software application.

Silicus Solution

The task before Silicus was to create a QTP test automation framework to help automate regression functional testing of the application during a new version release. The first challenge faced by the QTP test automation experts was unraveling the functional complexities and interlinking in each module. Another challenge was setting up the test environment needed to host and run the application within Silicus premises.

THE SOFTWARE APPLICATION

The software application was developed in VB, encompassing millions of lines of code and a complex interlinking between various functional modules and components. In all, the application comprised of 16 functional modules, and all modules were interlinked. The application generates and compares reports for submission to HEDIS® for compliance reporting and physician performance measurement. The typical last date for submission of reports is June 30
th

. Each year, the governing body changes the parameters for reporting and the client's development team releases a new software version to comply with the changes.

The task for the QTP automation testing team was to create a framework that was easily configurable to support different parameters, without having to re-work on the framework each year.

ENVIRONMENT SET-UP

The environment was set-up at Silicus with assistance from the client team. The environment included the software application, QTP application, JBoss application servers and all versions of SQL server.

QTP FRAMEWORK SELECTION

Following a detailed study of the software application functionality, modules and interlinking, Silicus QTP test automation experts decided to use a data driven framework. This framework was selected as the application was data intensive in inputs and there was data dependency on other modules and components.

Another consideration for framework selection was the changing nature of the application each year. The QTP test automation team had to develop a generic and configurable framework that could be used each year without making too many changes or incurring high maintenance costs.

WRITING AUTOMATION TEST CASES

The application did not have an existing repository of manual test cases that could have served as a reference for the QTP team. The team was provided with the application tutorial given to end users (customers). The QTP automation testing team went through the tutorials, understanding the functional aspects of the application, and creating the automation test cases. The QTP test automation team created as many as 90 test cases to cover the entire software application.

QTP TEST SCRIPTS, TEST CONTROL TABLE AND TEST DATA CREATION

Creating the test scripts was a challenging exercise as the application contained multiple 3rd party controls and embedded web controls. The QTP automation testing engineers had to use “discrete programming” techniques to ensure that these controls were recognized by QTP. The logic was written for each test case, along with creation of test data for each test case.

The test control table stored information related to location of test scripts, reporting, database information such as user name and password, and application authentication related information.

TEST RESULTS REPORTING

Custom code was written to transform QTP generated reports into more intelligent reporting formats. The QTP test team created reporting options in Text, HTML and MS Excel. An e-mail option was also provided for mailing the reports to concerned parties from within the application.

USABILITY OF QTP TEST SCRIPTS AND FRAMEWORK

The test scripts and frameworks were created for users with basic QTP knowledge. A user can access the QTP application through a series of excel work sheets that allow them to select test cases, specify test data, execute test scripts and view the results.

MAINTAINABILITY

The QTP automation testing team followed a “one folder” approach to maintaining scripts, test data and test control table. This made it easier for maintaining the QTP environment. The application administrator could manage settings for reporting preferences, location of interfaces and scripts etc. Application authentication information was stored within the test control table in an encrypted format.

At all points in the project, the focus were on creating a robust and scalable framework and test scripts. This attention to design ensured that QTP script generation and maintenance wasn't time consuming enough to negate the benefits from the QTP test automation project

Technologies Used

DATABASE
SQL Server 200x

TEST AUTOMATION PLATFORM
HP Quick Test Pro

APPLICATION UNDER TEST (AUT)
VB

APPLICATION SERVER
JBoss

CONFIGURATION MANAGEMENT
Team Foundation Server (TFS)

Client Benefits

ROBUST AND SCALABLE QTP FRAMEWORK DESIGN

The QTP framework was created after carefully studying the complexities and functionality of the software application. The QTP test automation team realized that the application would undergo significant functionality changes every year, and hence an open, flexible and reliable framework was created.

REDUCTION IN TESTING TIME, JUMP IN PRODUCTIVITY

The QTP test automation framework created by Silicus resulted in significant time and cost savings for the client. The framework brought about significant productivity benefits, with resources able to accomplish more within the same timeframe.

EASE OF MAINTAINABILITY

For test automation projects, Silicus follows a basic premise where the effort and time for maintaining and upgrading the framework and test scripts should be significantly lower than the time saved through automation. Every aspect of design and logic creation was driven by this premise, creating a QTP test automation framework and scripts that was easy to maintain and use.

PROJECT COMPLETED IN LESS THAN 4 MONTHS

The client was looking to complete the automation testing project in less than 4 months, a tough proposition given the complexity and size of the application. Despite not having full functional understanding of the application, Silicus brought its extensive QTP automation testing expertise and experience to bear to ensure that the project was completed within the required time frame.