Printing Solutions Project Analysis Report

DeRisk IT Inc. recently completed a project working with a client that is considered an industry leader in printing software, is recognized as a reliable source of OEM software for print or electronic document solutions. With their ever-growing interest and expanding solutions, they sought to Automate the testing of their new software builds. The client acquired licenses of TestComplete to assist with the automation of daily regression and functional testing that were currently being performed manually. TestComplete, a dynamic automation testing tool offered by Automated QA, was obtained in the last quarter of 2008 by the client. Throughout the course of the project TestComplete proved to be an invaluable source that if maintained properly will cut testing efforts by one-half or more.   In October 2008, DeRisk IT, Inc. offered an offsite testing solution to assist with the implementation of TestComplete and the automation process. The printing solutions company procured DeRisk IT as their automated testing solution and opted to purchase 500 service hours, allowing DeRisk IT to create automated scripts.

DeRisk IT Inc. provided a two week ramp up period free of charge, starting November 4, 2008, during which time they gained access to the client's new software via FTP, and specifications documents. The tester assigned to the project spent many man hours navigating through the applications and becoming familiar with the workings of both pieces of software. DeRisk IT was fortunate to get involved during the early build stages of both applications and follow through until the first release to selected customers of the client.

During the course of the project there were many obstacles and hurdles to overcome, which is common in most software testing environments. With a unique collaboration, which included DeRisk IT, the printing solutions company, and Automated QA, the complex Project although was largely a success with a total of 224 automated scripts written against the two applications under test.

Setup/Testing Stats:

DeRisk IT and their client worked together to come up with the most beneficial testing approach/access approach possible.

- ✓ Test Complete v.6 originally slated for Automated Testing tool, due to developmental differences TestComplete v.7 is used. (see Successes/Failures for in-depth explanation)

✓ Testing occurred through daily Webex meetings. Meeting invitations were sent out daily to tester via email. Meetings were hosted using WebEx until March 2009. Go To Meeting was utilized from March-April 2009.

✓ VMWARE Images were set-up by the point of contact on the client side and loaded onto the Host machine. This setup provided access using the online meetings to conduct testing and script creation.

✓ New software builds, specs, and test files were loaded onto the client FTP site and access was provided to DeRisk IT to pull these down as needed for local computers, Ramp Up, and as needs arose.

✓ Printing Solutions client provided walk-through of VMWARE Image under test. Operating System utilized was Win XP, which housed test files, TestComplete, and builds of both applications. The applications were loaded onto the VMWARE image by the client's personnel when the build was considered stable by Development.

DeRisk IT/Printing Solutions Company Work Flow:

Webex Meeting Started by client
(VMWARE Image Shared, Skype converions)

Automated Test Case Creation on Image by DeRisk IT
(Individual Tests, Linked full Tests, Daily Regression Tests)

Defects/Concerns (sent Via email)

POC on Client Side
(Maintains Image, Answers questions, provides feedback)

Project Successes

- ***Availability of New Builds***

The applications developed by the printing solutions company DeRisk IT supported, were readily available with daily and/or weekly builds to further the testing effort and eliminate any reported defects. Because the automation scripts were being created against an ever-changing application due to development, many scripts were altered to reflect changes in the software. Through good communication between the client and DeRisk IT, many defects were found through automated script creation that had not previously been detected by the manual testing efforts.

- *Communication*

Daily Communication via Skype to the client side point of contact at the beginning of each daily meeting allowed for any unresolved reported problems to be discussed. Emails were also sent often to point out any differences seen in new builds, and questions which arose between specifications and the actual functionality of the applications.

- *Training*

DeRisk IT provided Training at the end of the project to the point of contact so that they may maintain any current scripts and continue to create automated scripts using Test Complete. The training occurred via Skype and online meetings to show test script creation on their real-life application, best practices, debug opportunities, and "know-how". The training occurred daily for approximately two hours during the last two weeks of the project (80 hours). This allowed the client's representative to ask questions, learn and maximize the features of TestComplete, and gain a better understanding of how to use TestComplete against any other software DeRisk IT's client may develop in the future.

Project Failures

- *Change in Application Programming*

Prior to the Project Start Date, DeRisk IT noticed that many objects, windows and controls in the client's software were not accessible using TestComplete. In other words, TestComplete's object browser/finder tool could not "see" these items and would only attach to the main window of the application. During the course of the software builds, the printing solutions company had migrated to using QT Library (previously TrollTech) as the software platform. TestComplete v.6 did not support QT Libraries and therefore this could have been a Project show stopper. After many conference calls between DeRisk IT/Client and DeRisk IT/Automated QA respectfully, a solution was made. Automated QA made the beta version of TestComplete (TestComplete v.7) available for use on this project alone. This allowed the automated script creation to be possible and the Project to resume once the beta was released. TestComplete beta v. 7 was released for DeRisk IT and their client's use only (no public release of beta) on February 6, 2009. This means that the automated script creation was delayed by several months due to the holiday season, and development of the beta.

- *Communication of Priority Script Creation*

DeRisk IT was provided with the functional specifications of the applications. However, their client never provided or implicitly expressed a set priority of what should be tested when, nor copies of any current manual scripts they were executing. These items were requested from the client several times via email, but largely ignored. On April 15, 2009 (end date of Project was April 23, 2009) a full daily regression test currently performed manually was provided to automate for both applications. These test cases were provided so late into the project that DeRisk IT could not completely automate all the cases before the project deadline. To their best effort they did complete a full regression test against one of the applications. This required writing over 40 new test scripts within 5 days.

- *Hardware Failure*

  Throughout the course of the Project, the Web Ex daily meetings would end abruptly leaving the tester unavailable to create any scripts on the VMWARE image. This led to many scripts being created locally, then moving them over to the image, which was very time consuming and basically meant re-writing the same scripts again. DeRisk IT began to also notice low disk space errors on the machine in mid-February, which they reported to the POC. The point of contact added more disk space to the image and testing continued as did the errors. It was found that files were being stored in the temp folder on the hard drive, which was taking up all the disk space. The solution was that daily, the tester would empty the temp on the image so that the errors did not occur and work would not be lost. The Web-ex meetings and Go-To meetings also created severe lag. This meant that while typing, the image would freeze then resume, causing the same letter to be typed often several hundred times in a row. In March the meeting ended abruptly and the client was contacted via email that the meeting had ended. In return, the client reported to DeRisk IT within the following days that the hard drive had crashed, and no backup was in place so all test scripts created to that point (65) were lost. Luckily, DeRisk IT stored many created scripts locally and was able to salvage 45 scripts and move them back over to the image. This worst-case scenario delayed the project 3 weeks, which were spent re-creating scripts, re-mapping the entire application on the image, and moving over locally saved scripts. On the last day of the project, once all scripts were created (including linking functionality scripts to create several small regression test and a full daily test), the application incurred an error during the run of an install/uninstall test. This caused TestComplete to close. Upon recovery and launching TestComplete, several scripts were missing, and the entire project seemed to have reverted to an earlier date. The tester noticed that there were now two folders located on the hard drive for the applications where there was only one in the past. The tester used the remaining hour in the project to re-work many test cases, and re-link all scripts for the daily regression test. They also emailed instructions

to the point of contact on how to resolve any test script problems, as well as a list of missing scripts.

- ***Time Difference***

DeRisk IT and their client continued great communication throughout the course of the project, however, the time difference made any project delays exemplified due to the point of contact being in a different time zone (India). If the meetings ended, or test creation was halted, this resulted in a whole day of work on the VMWARE image lost. DeRisk IT did continue to work locally to create scripts, but later would have to use more time to move them to the image.

- ***Outstanding Defects in Software Application***

During the course of the project DeRisk IT noted and voiced concerns over several defects in the applications, and reported was varying functionality. For example, when files were saved, instead of the application saving the file with the specified name, it would save the file periodically with the prefix "TMP". DeRisk IT notified the client several times; however no solution or fix was offered or available in new builds. This could have been due to the upcoming release to select customers, however no explanations were ever provided. Another outstanding defect was the top "menu bar" becoming disabled during the testing. This was also dismissed as a VMWARE issue, however if the scripts continue to be maintained and ran on an image, they will fail periodically due to this. There were also many outstanding defects such as hotlinks not working properly. The functionality associated with open defects could not be automated since the functionality did not exist.

DeRisk IT's Solution to Overcoming Failures:

The biggest obstacle faced during the project was hardware failure. DeRisk IT worked diligently to create functioning and usable automated scripts, however, this hardware failure could have in-part been avoided in the following ways:

✓ *Documentation*
DeRisk IT proposes that at the on-set of any project one should have a defined Test Plan in place for release of test criteria, and test interruption, so that there is always a "Plan B" which will save Project allotted time. Also ensure that priorities are given to specifications and application functionality, so that the software is released with little to no outstanding defects (as set in the release criteria). Also, make any test plans and manual test cases available at the beginning of the Project to ensure they are completed during purchased service hours.

✓ *Backup*
DeRisk IT recommends that any computers, servers, etc. holding data or software pertinent to the Project always have a back-up. This ensures no loss of data should a worst-case scenario occur. Also, designate a person to make certain that the backup is maintained and current.

✓ *Attention to Details*
To provide the best possible piece of software, DeRisk IT proposes that all defects or testing concerns are given attention and all questions answered. This will ensure 100% coverage of all software functionality, and provide a better overall testing approach.