ROI BASED ENTERPRISE QA FRAMEWORKS

A Project presented to
The College of Engineering
San Jose State University

In Partial Fulfillment
of the requirements for the Degree
Masters of Science in General Engineering

By
Deepika Paruchuri
May 2009
APPROVED FOR COLLEGE OF ENGINEERING

Associate Prof. Leonard P. Wesley, CMPE Dept., MSE Director
San Jose State University

Prof. Dr. Jerry Gao, Ph.D., CMPE Dept, San Jose State University

Karen Burley, Engineering and QA Director, Hewlett Packard

Nimesh Minawala, QA Manager, Hewlett Packard

APPROVED FOR THE UNIVERSITY
Table of contents

1 Introduction................................................................................................................. 7
2 Hypothesis................................................................................................................... 8
3 Acronyms.................................................................................................................... 9
4 Economic Justification.............................................................................................. 10
  4.1 Executive Summary ........................................................................................... 10
  4.2 Market Need (Problem Statement)..................................................................... 11
  4.3 Solution and Value Proposition.......................................................................... 11
  4.4 Market Share ...................................................................................................... 12
  4.5 Competitors ........................................................................................................ 14
  4.6 Customers........................................................................................................... 15
  4.7 SWOT Analysis.................................................................................................. 15
    4.7.1 Strengths ..................................................................................................... 15
    4.7.2 Weaknesses ................................................................................................. 16
    4.7.3 Opportunities............................................................................................... 16
    4.7.4 Threats......................................................................................................... 17
  4.8 Cost - Benefit Analysis ...................................................................................... 17
    4.8.1 Cost ............................................................................................................. 17
    4.8.2 Benefit......................................................................................................... 19
  4.9 Breakeven Analysis............................................................................................ 20
  4.10 Net present Value ........................................................................................... 21
  4.11 Quarterly cost benefit analysis breakdown..................................................... 22
  4.12 Exit Strategy .................................................................................................... 23
  4.13 Product Roadmap ........................................................................................... 23
5 Project Schedule........................................................................................................ 24
  5.1 Gantt chart.......................................................................................................... 24
6 Milestones .................................................................................................................. 25
7 EAF Architecture...................................................................................................... 25
  7.1 EAF Architectural flow...................................................................................... 26
  7.2 EAF Test suite flow............................................................................................ 28
    7.2.1 Atomic Operation (Ao)............................................................................... 29
    7.2.2 SubTest Operation (Sto)............................................................................. 29
    7.2.3 Test Operation (To).................................................................................... 30
    7.2.4 TestSuite Operation (Tso).......................................................................... 30
  7.3 EAF Standards.................................................................................................... 31
  7.4 EAF Advantages ................................................................................................ 32
8 EAF Process.............................................................................................................. 32
Table of Figures

FIGURE 1: TOTAL MARKET SHARE ................................................................. 13
FIGURE 2: ROI BASED FRAMEWORK MARKET SHARE ......................... 14
FIGURE 3: BREAK EVEN ANALYSIS .......................................................... 20
FIGURE 4: P&L GRAPH .............................................................................. 21
FIGURE 5: GANTT CHART .......................................................................... 24
FIGURE 6: MILESTONES ........................................................................... 25
FIGURE 7: WORK FLOW DIAGRAM OF AN EAF ........................................ 27
FIGURE 8: WORK FLOW DIAGRAM OF AN EAF ....................................... 30
FIGURE 9: GRAPHICAL REPRESENTATION OF AN EAF ......................... 31
FIGURE 10: EAF DESIGN FLOW CHART .................................................... 34
FIGURE 11: AUTOMATION PROCESS FLOW ............................................ 38
FIGURE 12: AUTOMATION RELEASE PROCESS FLOW THRU QUALITY CENTER 39
FIGURE 13: RELEASE MODULE IN QUALITY CENTER ............................. 40
FIGURE 14: AUTOMATION TEST PLAN WORKS FLOW THRU QUALITY CENTER 42
FIGURE 15: AUTOMATION TEST PLAN MODULE IN QUALITY CENTER ...... 43
FIGURE 16: AUTOMATION TEST LAB WORK FLOW IN QUALITY CENTER ... 45
FIGURE 17: AUTOMATION TEST LAB ...................................................... 46
FIGURE 18: AUTOMATION TEST LAB REPORT FROM QUALITY CENTER ... 47
FIGURE 19: QUALITY SURVEY RESULTS .................................................. 51
FIGURE 20: AUTOMATION TEST CASE PRIORITIZATION ....................... 57
FIGURE 21: EAF ROI ................................................................................ 60
FIGURE 20: FAILED SCREENSHOT OF ATOMIC OPERATION .................. 228
FIGURE 21: FAILED SCREENSHOT OF SUBTEST OPERATION .................... 229

Table of Tables

TABLE 1: DEFINITIONS AND ACRONYMS ............................................. 9
TABLE 2: STARTUP COSTS ...................................................................... 17
TABLE 3: PERSONNEL COSTS (SALARY.COM) ....................................... 18
TABLE 4: FIXED COSTS (HP.COM) .......................................................... 18
TABLE 5: VARIABLE COSTS .................................................................... 19
TABLE 6: NET PRESENT VALUE ............................................................... 22
TABLE 7: QUARTERLY COST AND BENEFIT ........................................... 23
1 Introduction

ROI based EAF (Enterprise level automation framework) will make a change in Quality assurance automation development life cycle. EAF has defined the relationship between manual testing team and automation engineers. EAF provided a way to automate customer use cases, state based testing method for any component in a product, and objective oriented testing. EAF has provide a way to improve the quality of the product for customers. EAF will give maximum ROI. It needs to be used in an optimal fashion. EAF development is fast and provides a stable framework.

The technical literature throws the light on amount of work done in software quality assurance and software quality assurance automation. Literature throws a limited light on automation frameworks implementation in Enterprise level. During 1980’s a major revolution took place in the industrial sectors of America in Quality. Deming, Stewart, and Taguchi’s principles came into lime light. Break down barriers between departments (Deming, 1940s) is key approach implemented in designing relation between development, Manual quality assurance engineers and Automation engineer.

Economic Surveys of the Quality Assurance market from various websites reflect the unprecedented awareness and growth opportunity in Quality Assurance for the technology (American society of Quality). The services base in the start-ups, small to Subtest scale business, and large scale business market margins differs from project to
project. There is also an increased competition to tap into this growing phenomenon. As industry prepares for this next wave of growth, quality, costs, and opportunities have become important considerations. The Service market is exploring solutions beyond the software.

Proposed EAF promises a cost effective, reliable, and efficient utility solution. In an enterprise setting, it makes sense to implement such a framework in the early stages of the project. Economic analysis shows that this technology has positive ROI and can breakeven relatively quickly and has positive Net Present Value proving the commercial viability of the proposed EAF.

2 Hypothesis

Enterprise level automation framework will help automation engineers in developing the test script on an enterprise solution application with maximum ROI. For the project manager it will be easy to track the progress. The manual testing team can easily customize the automated test cases, customer support engineers can also easy track the test cases developed in the customer perspective. The higher management can easily estimate the quality of the product before the release and will have access to the test reports and quality metrics.
### Acronyms

<table>
<thead>
<tr>
<th>Term</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>CMM</td>
<td>Capability Maturity Model</td>
</tr>
<tr>
<td>CMMI</td>
<td>Capability Maturity Model Integration</td>
</tr>
<tr>
<td>QA</td>
<td>Quality Assurance</td>
</tr>
<tr>
<td>EAF</td>
<td>Enterprise level Automation Framework /Enterprise Automation Framework</td>
</tr>
<tr>
<td>POA</td>
<td>Plan of Action</td>
</tr>
<tr>
<td>PRD</td>
<td>Product Requirement Documents</td>
</tr>
<tr>
<td>QTP</td>
<td>Quick Test Professional</td>
</tr>
<tr>
<td>SDLC</td>
<td>Software Development Life Cycle</td>
</tr>
<tr>
<td>QC</td>
<td>Quality Center</td>
</tr>
<tr>
<td>ROI</td>
<td>Return on Investment</td>
</tr>
</tbody>
</table>

*Table 1: Definitions and Acronyms*
4 Economic Justification

4.1 Executive Summary

Automation frameworks help QA vertical in any company to steer the automation needs in desired direction by providing flexibility and extendibility. There are many open source automation frameworks like STAFF which exist in the market today, but none of the frameworks are designed to work with QTP. QTP is an industry standard automation tool. ROI based QA framework is designed to work with QTP and QC. It shall be delivered as a complete QA automation solution which would help the companies reach their quality goals. The clients “quality” needs would be fulfilled without knowing the intricacies of the framework. The clients would need to implement the business logic using the scalable framework.

By focusing on the large fortune 500 companies as end clients and their quality needs the company would target towards revenue of $5M by the end of 2011 and profit of $5.6M in next 3 years. The market share can be achieved by reaching customers like HP and expanding the base to other major companies like Microsoft and Apple.

Major competitors are the service companies like Infosys, HCL and Wipro which provide QA solutions for clients. The solution is unique as automation framework is also included in the complete package. Partnering with the major competitors would reduce the risk. To achieve complete customer satisfaction a 24x7 support team is planned to operate from across the globe.
The Company would need $3M to incorporate the company and sustain for 3 years. The breakeven would occur in the first year. The financial estimates are conservative given the market conditions and the economic slowdown.

4.2 Market Need (Problem Statement)

Automation QA Framework which works reliably with QTP and QC is not available in the market today. Open source frameworks like STAFF work with Selenium and Perl and are not user friendly. QA automation frameworks are ready to significantly alter the face of the manual testing in the software industry. The software industry in general need the quality products delivered and hence the automation consultant’s need grew at a very fast pace in past several years. Hiring consultants to develop the custom QA frameworks would be costly as it would include fixed costs and variable costs as long as they are hired.

4.3 Solution and Value Proposition

ROI based QA frameworks are a unique and complete QA solution which would capture the quality needs of the customers. The ROI based framework is integrated with QTP and QC and provide many EoU(Ease of Use) features.
Hiring consultants to develop custom frameworks that would work with QTP and QC would incur cost to the company. Cost to develop a custom framework would include the following

- Hiring 3 senior QA automation consultants for at-least 3 months
- Fixed and Variable costs that are associated with senior QA consultant

ROI based EAF framework would be a cost effective solution as only the business logic has to be embedded into the framework which would need 1 junior QA automation consultant for 3 months. Once the business logic is implemented the development team could customize the automation based on the customer use cases and test the product to meet the quality goals of the company hence showing ROI to the senior management.

4.4 Market Share

The total market size for QA is $21869 million. Below pie chart shows the detail breakdown of the major companies involved in QA services business. The below data is based on the company financial results and fidelity portfolio management data sheets.
For the ROI based framework a 0.01% of market share is required to show a positive NPV over 5 years. The Company market share is expected capture about 42 percent in three years based on the market capture rate of the prototype deployed at HP, already 3 projects are using this framework. Service companies captured only 75% of the market share and are unable to cater to the needs of the customers. The ROI based QA frameworks shall capture the un-hit market as this is a low cost solution.
<table>
<thead>
<tr>
<th>Year</th>
<th>Market Share (Thousands of Dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Company Market Share</td>
</tr>
<tr>
<td>2009</td>
<td>500</td>
</tr>
<tr>
<td>2010</td>
<td>3000</td>
</tr>
<tr>
<td>2011</td>
<td>5000</td>
</tr>
<tr>
<td>2012</td>
<td>3900</td>
</tr>
<tr>
<td>2013</td>
<td>4800</td>
</tr>
</tbody>
</table>

Figure 2: ROI Based Framework Market Share

4.5 Competitors

The major competitors to the company are the service companies like Ernst and Young, Price Water Coopers, Infosys, HCL and Wipro. These competitors would serve the major companies which would need to attain the high quality needs and have huge budget for quality.

Partnering with these companies to reduce their costs for ROI based frameworks would mitigate the risk associated with competitors. If the companies do not partner due to any reason then selling the product to startups where cost is a very important factor would mitigate the risk.
4.6 **Customers**

The company will target major companies with a low cost solution. A prototype demo has been given to Hewlett Packard and it has agreed to sign up for one of its major products. The targeted customers are expected to be as follows

- 80% of all existing divisions of HP
- Microsoft – Volume licensing for divisions across Portland office
- Apple – Customized solutions
- Startups which need basic solutions. Part of the frameworks can be deployed at retail pricing.

4.7 **SWOT Analysis**

SWOT analysis helps in analyzing the growth aspects of the business.

4.7.1 **Strengths**

- Reliability– EAF is generally reliable. It can be easy to integrated with various platforms when compare to other automation framework.
- Low cost – As per Moore’s law, cost of hardware will go down every year as features increase (Wikipedia, 2007). Compared to the other frameworks EAF is cheaper in development and need less resources which save time and cost to the company.
o Location advantage – Silicon Valley has a culture of supporting innovations. This will be a huge competitive advantage for the EAF.

o Manufacturing: No manufacturing cost

### 4.7.2 Weaknesses

o Training: Since EAF is involved with training and since the core team does not have direct relevant experience, proposed Automation lead would need to acquire this skill set and train the remaining team which could increase business risks in terms of costs and sharing the proprietary information with the consultant.

### 4.7.3 Opportunities

o Partnering with ERNST AND YOUNG – ERNST AND YOUNG has a comparatively lesser market share compared to PRICE WATER COOPERS and hence the targeted customer is ERNST AND YOUNG. It would like to implement the new frameworks and improve the market share.

o Partnering with global presence companies – Major market leader for Subtest to large business is HCL, and Infosys and our company would like to partner with it to improve the quality of the products of the customers.

o Consulting as a service company- Our Company can continue as Service Company.
4.7.4 Threats

Vendor acceptance– The contracts sign off with ERNST AND YOUNG, HCL, and Infosys is the biggest threat as our business relies on it.

4.8 Cost - Benefit Analysis

The initial process will be developed based on the complexity of the company’s product and deployed. The implementation is done in phase level. The main objective or goal is to keep the costs to minimum and improve the companies Quality Assurance and generate maximum ROI product.

4.8.1 Cost

The initial cost to develop the prototype is measured in dollar amounts and displayed in a tabular format. Fixed costs, variable costs, and personnel costs are calculated per year.

The following table as the costs associated to start the company

<table>
<thead>
<tr>
<th>Initial Cost to start company</th>
<th>Total Cost (In Dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td>CPA/ Lawyer</td>
<td>250.93</td>
</tr>
<tr>
<td>Company Incorporation and Setup costs</td>
<td>1000.00</td>
</tr>
<tr>
<td>Total</td>
<td>1250.93</td>
</tr>
</tbody>
</table>

Table 2: Startup Costs

The following table has the costs associated to personnel.
The following table has the fixed costs. Assumptions are as follows.

- Need 6 HP workstations costing $2K each
- Need 4 dell servers costing $6K each
- Corporate software licenses required for a year costing $40K
- Environmental Health certificate costs $1000.00

<table>
<thead>
<tr>
<th>Fixed Costs</th>
<th>Total Cost (In Dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Work station+ servers</td>
<td>36000</td>
</tr>
<tr>
<td>Bills</td>
<td>70000</td>
</tr>
<tr>
<td>Software’s</td>
<td>40000</td>
</tr>
<tr>
<td>Environmental Health</td>
<td>1000</td>
</tr>
<tr>
<td>Total</td>
<td>147,000</td>
</tr>
</tbody>
</table>

Table 4: Fixed Costs (hp.com)

The following table has the variable costs.

<table>
<thead>
<tr>
<th>Personnel</th>
<th>Total Cost Per Hour (In Dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Administrative Assistant 1</td>
<td>15.00</td>
</tr>
<tr>
<td>Office Manager</td>
<td>30.09</td>
</tr>
<tr>
<td>Marketing Manager</td>
<td>50.25</td>
</tr>
<tr>
<td>Total for a year</td>
<td>209,748</td>
</tr>
</tbody>
</table>

Table 3: Personnel Costs (Salary.com)
## Table 5: Variable Costs

<table>
<thead>
<tr>
<th>Variable Costs</th>
<th>Total Cost (In Dollars)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Office Space and Maintenance</td>
<td>150000</td>
</tr>
<tr>
<td>Office Supplies</td>
<td>25801</td>
</tr>
<tr>
<td>Travel</td>
<td>100000</td>
</tr>
<tr>
<td>Total</td>
<td>275,801</td>
</tr>
</tbody>
</table>

Total company setup cost = $1001.93 + 209,748+$147,000 + $275,801 = 633550.93 ~ $64K

### 4.8.2 Benefit

The Economic Survey of consultant business has indicated mixed results growth due to the economic conditions in 2009. Survey results were neutral; few people opinion was due to the economic condition companies need to produce quality product in order to product client should strong Quality Assurance, in order to raise the bar they hire the consultant. Other side of the coin, companies’ may not prefer consultants due to cost and downfall of the economy. As per my business analyst we have the scope to increase in market.

Assuming Quality Consultant would be adapted by 4% of Fortune 500 companies and 2% by Subtest to large companies, the potential customer base for us would be 6% customers by end of 2012. Starting with HP in one division (which has 10 different projects) and will pay $500K. Assuming in HP we can expand our business to other divisions, by the end of 2010 cost would be $1M and revenue would be $3M.
4.9 Breakeven Analysis

The following table shows the breakeven analysis. The following are the assumptions for the analysis.

1. In HP, we can expand the business to other groups/divisions
2. At the end of 2010, the business will expand to $3M
3. May start a sister consulting company in India in order to minimize the cost.
4. Marketing from India thru call center
5. Every year the adoption will be double

![Breakeven Analysis Graph]

<table>
<thead>
<tr>
<th></th>
<th>2009</th>
<th>2010</th>
<th>2011</th>
<th>2012</th>
<th>2013</th>
</tr>
</thead>
<tbody>
<tr>
<td>Total Cost(Millions)</td>
<td>0.63</td>
<td>1</td>
<td>1.2</td>
<td>2.4</td>
<td>3.2</td>
</tr>
<tr>
<td>Total Revenue(Millions)</td>
<td>0.5</td>
<td>3</td>
<td>5</td>
<td>3.9</td>
<td>4.8</td>
</tr>
</tbody>
</table>

Figure 3: Breakeven Analysis

Total cost in 2009 to establish the business is $0.6M. In 2010, assuming marketing will ramp up and estimated costs for 2010, $1M. In 2011, assuming that we will spend on R&D of the business goes to $1.2M. In 2012, expanding the business in India and other developing countries goes to $2.4M. As this business is started with a low budget, we
can breakeven in 2010 as the total cost is $1M and total revenue to the company is $3M. In 2011, we are expecting to expand in other divisions of HP

Below is the P&L chart which spans across Q2 FY’10. The company would be profitable starting Q4 FY’09.

![P&L Chart](image)

**Figure 4: P&L graph**

4.10 Net present Value

The net present value is calculated based on the dollar value in future for the present investment. Based on the rate of return for the next four years the NPV (Net present value) is calculated as below.

<table>
<thead>
<tr>
<th>Year</th>
<th>Total Cost (Millions)</th>
<th>Total Revenue (Millions)</th>
<th>Profit (Millions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>2009</td>
<td>0.63</td>
<td>0.5</td>
<td>-0.13</td>
</tr>
<tr>
<td>2010</td>
<td>1</td>
<td>3</td>
<td>2</td>
</tr>
</tbody>
</table>
From the above the NPV is calculated using the excel formula for NPV, which is NPV (Discounted rate of return, return in year 1…, return in year n).

Discounted rate is 10.5% and hence the NPV calculated will be +0.10M. Since the NPV value is positive, it indicates that the business is viable and can be pursued.

4.11 **Quarterly cost benefit analysis breakdown**

Below table indicates the quarterly costs and benefits.
Table 7: Quarterly Cost and Benefit

<table>
<thead>
<tr>
<th></th>
<th>2010</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Total</td>
<td>0.63</td>
<td>0.5</td>
</tr>
<tr>
<td>Q1</td>
<td>0.4</td>
<td>0.54</td>
</tr>
<tr>
<td>Q2</td>
<td>0.3</td>
<td>0.62</td>
</tr>
<tr>
<td>Q3</td>
<td>0.3</td>
<td>0.78</td>
</tr>
<tr>
<td>Q4</td>
<td>0.2</td>
<td>1.06</td>
</tr>
<tr>
<td>Total</td>
<td>1.2</td>
<td>3</td>
</tr>
</tbody>
</table>

4.12 Exit Strategy

The company can be bought out by major competitors like Ernst and Young or Price Water Coopers to survive the competitive threat from outsourced companies like Infosys, Wipro and HCL. Another exit strategy option would be that HP would buy the product and integrate it to the existing automation and quality center suite.

4.13 Product Roadmap

The ROI based frameworks is developed in phases and the beta is released to the customers in April of 2009. A major release 1.0 is planned by the end of 2009. The API based framework and enhanced utility functions shall be in second major release and shall be targeted by mid of 2010. Capability of integration with HP test suite shall be developed by beginning of 2011.
5 Project Schedule

5.1 Gantt chart

Figure 5: Gantt chart
6 Milestones

Figure 6: Milestones

7 EAF Architecture

High level design includes definition of various blocks of the design and their inter-relationships/interoperability. In the context of the codes, high level design means dividing the design into functional blocks such as invoking data sheets into QTP, capturing screenshots, and logs in the Test Suites of the component modules, customer
scenarios, and defect automated test cases. EAF design consists of several, standard and methods blocks which are shown below.

7.1 **EAF Architectural flow**

EAF breaks down the automation components based the size of the operation. Below diagram shows the formation of the automation operations.

**Quality Center Database servers:** Database server all vb scripts and perl scripts are stored in the database. This is inbuilt from the Quality center, during this project development, We took complete advantage of the inbuilt database server.

**Reusable Libraries for Enterprise Level Automation Framework:** Reusable libraries are used for the development of enterprise level automation framework, like to import data from the datasheets, during errors to take the screenshots, common libraries for logging which includes high-level and detailed logs of the application.

**Application Level API using VB script and .Net Framework:** Scripts are developed in API level using VB script and .Net Framework. All atomic operations, Subtest operations, Test operations and Test Suites are developed using VB scripts and to check emails, send emails and outlook client testing scripts were developed using .Net Framework standards.
Figure 7: Work flow Diagram of an EAF
Datasets thru XML, Datasheets: All the data passed thru the external spreadsheets and XML files. Environment parameters are configured thru the XML file and spreadsheets are import into the atomic operations, Subtest operations, Test operation and Test suite operations during the runtime. The datasheets are easily configured for any environment for any user to use on any client machines.

Quick Test Professional and Quality Center: VB Scripts are developed using the QTP IDE and thru Quality Center these QTP scripts are invoked during executions. During development of scripts, we took the advantage of inbuilt functions of the QTP like split functions, system utilities, run time, etc.

Quality center test plan and test lab modules are used to checkin the QTP scripts and for the execution. QC is used to track and report the progress of the scripts.

Clients: Different clients were used to execute the test case on different configurations and test environment

7.2 **EAF Test suite flow**

EAF Test suite flow consists of Atomic Operations, SubTest Operations, Test Operations, and Test Suites. User has the ability to build the test suites on the fly based on the customer cases and Engineering request.
7.2.1 Atomic Operation (Ao)

This lowest the logical piece of user operations. This can be single click on a web button from both backend and front. This is the basic block of the automated suite. In EAF framework QA Automation Engineer will breaking the application actions and events into lowest logical operation programmatically. These Atomic operations to build is easy using descriptive programming.

7.2.2 SubTest Operation (Sto)

Sto is group of logical Ao’s. Example: user entering name in a text field box is one Ao, user entering Address field 1 is another Ao, grouping of logical Ao is Sto. Once QA Automation engineers develop the atomic operation, based on the testcases objective automation engineer will group the atomic operations into meaningful SubTest operation. For example: Entering username is one atomic operation, Entering password is another atomic operation, and clicking login button is another atomic operation. Calling these atomic operations in a sequence is called SubTest Operation.
7.2.3 **Test Operation (To)**

Group of Sto’s is TO’s. Entering bio-data is one Sto and purchasing an order is another Sto. Group of logical Sto’s is Test Operation. Calling logical SubTest Operations together are called Test Operation.

7.2.4 **TestSuite Operation (Tso)**

Logical group of To’s is Testsuite Operation. This can be customer scenario or defect or component test. It is always not necessary to call only Logical To’s to make TestSuite operation, based on the test scenario Automation Engineer can call Ao’s and SubTest Operations also.
7.3 EAF Standards

Following are the EAF standards

- Relation between manual test engineer and automation engineer
- Relation between manual test lead and automation lead
- Defect tracking
- Coding standards
• Naming conventions for the scripts
• Agile model of deliverables, Automation scripts deliverables in sprints methods
• Identification of to be automated test cases

7.4 EAF Advantages

The following are the advantages of using the EAF.

• EAF is scalable and reusable: For new customers, EAF is easy to support on multiple platforms. EAF can be reusable on multiple projects
• API use by development: EAF API can also be used by development and integration tester for unit level and integration testing.
• Data Driven: User can do testing with user defined datasets
• Configurable: EAF is easy to configurable to make test suites
• Reduce Cycle Time: Reduce development cycle time, bring in more product features, and increase flexibility to design changes

8 EAF Process

EAF is Enterprise Level Automation framework. This framework is basically developed for the automation in an enterprise solution environment. It can also scale to the project level and product level automation. It is build using VB script, Perl, QTP and QC. QA engineers have to perform various kinds of testing such as integration, smoke/sanity
testing, functional testing, regression testing, performance, and load and stress testing.

EAF helps in all levels of testing.

8.1 EAF Process Flow

Here are the EAF process flow diagrams which explain the Enterprise level Automation Framework. This process flow plays very vital role in implementation phase of the EAF and also during the execution of the automated test cases.
Quality Center server invoke/execute scripts design specs:

To meet the desired goal, specifications are needed for the design of the test cases. These specifications describe standards, methods, operational uses, interface and overall
architecture of the automation design of the product. The design feature and specifications will be determined by following criteria

✓ Reusability of the component
✓ Complexity of the component
✓ Configurability of component on different test environments
✓ Special Hardware required to test a component
✓ Manual test efforts
✓ Quality of the component, if it’s a existing feature

Based on above mentioned criteria, thru QC test engineer can execute test suite based on the priority and severity.

❖ **Component based automated test scripts thru QTP**

Component based development is used for shorter development life cycle to meet marketing needs; in order to meet in the schedules with development QA has do component based testing. Several well-defined component-based development methods been published to support component-based software. Cheeseman and Daniel UML Component[8], D’Souza and Will’s Catalyst, and Herzum and Sims’ Component Factory[10] are typical examples.

❖ **Customer Scenario’s automated test scripts**

Automation engineer will develop test scripts based on the customer cases that were logged in the field. Customer scenarios test cases will be reviewed by the customer support engineers.

❖ **Defects automated test scripts**
Severity 1 and 2 defects will be automated by the automation engineer based on the feedback from the engineering and manual testers in the team.

- Collect logs thru Quality center or Test Environment

Executed test scripts logs can be viewed in the quality center database or in the test environment automated test cases folder specifically designed for the logs. Logs will be provided with screenshots, clear description and latest result of the last run.

- Analyze the test results

The test results will be analyzed by the automation engineer. If the test results are logged in red color, it means test is failed and automation engineer will go in details and collects logs, and files and defect in QC, green test case is passed.

- Update senior management with weekly report

Automation lead will be updating the senior management with weekly progress in order to keep the track of the project and quality of the product.
8.2 EAF Automation Flow

The automation process steps are detailed is described in detailed below.

Following are the details of each automation process steps:

1. Collection of Test plan based on component: Test plan component data is collected for the automation development
2. Analysis of Customer cases: Automation engineer will work with Customer support to analyze the customer scenarios

3. Analysis of component defects based on defect severity: Defects are analyzed from the defect tracking tool based on severity, all high severity defects will be automated by the automation team

4. Developing Automation Test plan: Automation consultant has to develop the automation test plans base manual test plans, customer cases and defects and define POA.

5. Customizing EAF: Automation consultant will customize the framework based on the product

6. Training automation engineers: Automation consultant will train the automation engineers on the EAF

7. Track Project Progress: Project e tracked based on the Agile Sprints

8. Submit QA metrics and product quality to senior management: Automation engineers will perform regression on every qualified Build of the development, if there are no high severity defects opened by the manual tester and updates senior management every week with test results. These test results will be automatically generated by using QC
9 Quality Center Standards

9.1 Releases

Quality Center standards were defined to the automation engineers in order to standardize the process. The standards were defined based on the Quality center standards.

![Diagram: Automation Release Process Flow thru Quality Center]

Figure 12: Automation Release Process Flow thru Quality Center

Releases Module Tab in Quality center
Mandatory Fields are defined to take the advantage of the release module:

1- For each release QA cycle, we have defined the Start Date and End Date – these fields helped us in planning development of the test scripts to define the start time and end time and also included the execution time for the regression test cases.

2- Assign Test Set folders assigned to cycle- This helps the senior management to track the Automation progress.
3- Assigned product defects to QA Cycles - Defects found during test execution was raised and assigned to the test cycle in order to keep track of the quality of the QA cycle.

9.2 Test Plan Module

Test plan module is used to plan the test strategy for test automation, all the manual test cases are checked into Quality Center, based on the manual test plan, automation engineer and the manual testers analyzed the test cases which can be automated based on the manual frequency execution, test time, hardware setup time, and manual test effort.

They are organized per release and divided into three categories viz.

**Defined Automation test Strategy:** Based on the manual frequency execution, test time, hardware setup time, and manual test effort test cases automation priority is defined

**Define Test Subjects and Design Tests:** Manual tester write test cases to as per the components

**Design Steps:** Manual tester developed detailed test cases steps in Quality Center. These test cases are developed based on the component test cases, customer scenario and defect based test cases

**Automated Tests:** Automation engineer will develop test cases and check in to the Quality Center based on the category of the test cases

**Analyze Test plan:** Based on the automated test cases check in to quality center, development of automated test cases are analyzed
i. **Component Test Cases** – Test cases for features new in this release.

ii. **Customer scenario Test Cases** - Test cases developed from customer cases (Legacy), for this feature.

iii. **Defect based Test Cases** – Cases developed as outcome of defect verification, for this feature.
These instructions were developed to the manual engineers to follow during development of the test cases

Test Cases are written feature wise. It contains of the following details:

1. **Owner**: Owner / developer of the test case

2. **Test Name**: Name for the test cases. The convention for this is

   \[ \text{<Feature Name>_<Feature Type>_<Test Case Number>} \] where

   a. **Feature Name**: Shortened form to identify the feature. For ex: And or Query : AOQ, OS Upgrade: OSUp etc. This may not be valid for hot fix.

   b. **Feature Type**: Can be one of

      i. **NF**: New Feature
      ii. **CS**: CSAT
      iii. **DV**: Defect Verification
      iv. **HF**: Hot fix
c. **Test Case Number:** Number for test cases with format XXX starting from 001.

3. **Status:** Status of test case – values as in QC

4. **Keyword:** is a mandatory field. The values are
   a. **Component test cases:** For new feature cases
   b. **CS:** For Customer scenario Test Cases
   c. **DV:** For Defect Verification

5. **Test Setup:** In the **description** field for the test include the following details
   a. **Valid for versions after:** Version name after which this test cases holds
   b. **Setup required:**
      i. **Hardware:**
      ii. **Third Party Apps:**
      iii. **Any other setup info:**
   c. **Large Scale:** Yes / No
   d. **Support case no:** If valid
   e. **Defect number:** If test case is based on defect verification
   f. **Any other info:** Provide additional notes.

6. **Design Steps:** Include specific steps for the test case. Cut the steps into atomic steps each containing instructions to execute and expected result for each step.
9.3 Test Lab Module

Test Lab module is used to execute the test cases developed in test plan based on the test sets. These test sets are assigned to Release modules. The test lab is set of test cases to be executed for the sprint/drop from the test plan. The test lab contains subset of test cases extracted from test plan.

![Automation Test Lab Work Flow in Quality Center](image)

**Figure 16: Automation Test Lab Work Flow in Quality Center**

The test lab for major release consists of component Test cases, Customer scenario test cases, Defect verification. This is to be organized as under:
After executions from Quality center, engineer can generate detail execution report with total numbers total test cases, total number of test cases passed, total number of test cases failed. This gives a report can automatically submit it to the senior management.

Figure 17: Automation Test Lab
Figure 18: Automation Test Lab report from Quality Center
10 Templates developed for Test Design Specifications

For manual testing, detail test design specifications are developed for the component testing. This test design specification template was useful for the automation team in understanding the component.

1- Revision History

2- Sign off

QA Team

Engineering Team

Product Management (Optional)

Support (Optional)

Program Manager (Optional)

3- Overview

3.1 Introduction

-Introduction (Introduction of test design)

-Scope

3.2 Background (Background of this feature)

4- Requirements

4.1 Hardware Requirements

4.2 Software Requirements
5- Test Matrix (Brief description of Testing approach matrix and Test configuration matrix)

5.1 Testing Approach Matrix

<table>
<thead>
<tr>
<th>Test Folder</th>
<th>Approach /goal</th>
<th>Test cases</th>
</tr>
</thead>
<tbody>
<tr>
<td>Login</td>
<td>Use cases (Verification Go)</td>
<td>T1 T2 T3</td>
</tr>
<tr>
<td>Search mail</td>
<td>Use cases (Verification Go)</td>
<td>T4 T5 T6</td>
</tr>
</tbody>
</table>

5.2 Testing configuration Matrix

<table>
<thead>
<tr>
<th>Feature</th>
<th>Configuration</th>
<th>Test cases applicable</th>
</tr>
</thead>
<tbody>
<tr>
<td>Search Mail</td>
<td>Use cases (Verification Go)</td>
<td>T1 and/or T4 and/or T3 and/or T5</td>
</tr>
</tbody>
</table>
11 EAF Implementation

11.1 EAF Survey

To convince the senior management that EAF process is needed to improve quality in the company a survey was given to the following employees across the division. The survey consists of 15 questions which were multiple choices by nature.

The following employees were involved.

- QA Managers
- QA Directors
- Program Managers
- Project Managers

The division had 5 major projects which were in various phases of the product lifecycle. The survey concentrated on the understanding the following.

1. Current Quality Process: The employees assessed if the current quality process in terms of various kinds of manual testing and documentation was satisfactory and was yielding results.

2. Current Automation Process: The employees assessed if the current automation process in terms of documentation, execution and reporting was satisfactory and was yielding results.
3. Quality Levels: The quality of the product is measured using the defect data accrued at various levels of project and the absolute number was taken from employees as part of this survey.

4. Customer Satisfaction: The support cases were the major contributor to this question. The customer satisfaction level was an absolute number the employee could provide.

The survey results are as shown below:

![Quality Survey Results](image)

**Figure 19: Quality Survey Results**

Below is the analysis of the survey results.
Project 1 and Project 2: The current quality and automation process was in place and the quality levels were acceptable. But the customer satisfaction was lesser than the quality levels as assessed by the project. This indicated that the defects were caught early in the project but were not focused on the customer use cases.

Project 3: The current quality levels were pretty low. The automation has just started and the product was not delivered to customer. This project is in the initial phases were the quality improvements can be incorporated.

Project 4: The quality levels and current quality and automation process was ranking high. A process engineer was hired for this project and hence the customer satisfaction was also high.

Project 5: Even though the quality levels were so low, the support team did an excellent job fixing the defects as required. This is similar to a stealth level startup project were the product is delivered to the customer with defects and support teams with engineering support works hard to resolve issues as needed. This is generally not a sustainable model.

11.2 **EAF Prototype**

The senior management studied the survey results and the conclusions and advised that a prototype development on project 3. Project 3 was in the initial phases of development and had a room to implement the EAF process. Project 3 had development and QA teams
spread across the globe. The QA team was in India and Development teams were in UK and USA.

11.2.1 **Trainings**

A series of training classes on the following were given to all the employees who were working on this project.

- Quality Engineering Best Practices
- Agile Testing
- Test Plan documentation
- Test Design Specifications
- Test Case Specifications
- EAF Process
- Quality Center Modules
- Quick Test Professional
- Backend custom test tools

For each of training the training material was posted on the QA wiki, sharepoint and the participants had a glance at the training materials before attending the training sessions. Each training session end with a Q&A and practice questions, just to make sure that all the participants understood the process.

Travel programs were arranged so that the global teams were also trained in various areas. The training process took around 4 weeks to complete. The training process was very intensive and many inputs were collected as part of the training process. Some of them include the following.
• Test Plan specification should be clearly divided into two sections. One section defining the technical aspects and other section defining the scheduling and resources part of it.

• Test Design specifications need not be written for features which already have developed test cases. Only new features need a test design specification document.

• Even though test cases are already developed they have to categorize as per the EAF framework.

• Execution teams have to be separate from the automation development teams.

11.2.2 Test Plan Specification

Test Plan was developed for Project 3 with the help of QA manager and QA Director.

Test Plan followed the EAF framework.

Test Plan clearly defined the following:

• Features that need to be tested
• Kinds of testing performed on each feature
• Testing environment for each feature
• Regression requirements for legacy features
• Regression requirements for the types of builds
• Scheduling and Resource requirements
• Engineering and QA plan integration to meet the program level milestones.
11.2.3 Test Design Specification

Each major feature in the project had a responsible EQE (Embedded Quality Engineer). The EQE was responsible to write the test design specification for the feature Engineer is responsible for. The test design specifications were developed following the EAF process. The document was reviewed by QA team in the first pass and then engineering reviewed it in the second pass. Once the test design specification was signed off by engineering the next step of writing the test case specification was taken.

Total of 13 test design specification were developed by 4 EQE over a period of 1 man month.

11.2.4 Test Case Specifications

Test cases were developed by the EQE’s based on the test design specification document. Each major feature had a test case specification document which consists of the test cases. All the test case specification documents followed the EAF process and each test case was classified based on the 4 categories. These documents were reviewed by QA and then by engineering. After the engineering sign off, the test cases were ready for manual execution.

Total of 9 test case specification documents were developed by the EQE’s over a period of 1.5 man months.

Existing 4 test cases documents were converted to format which EAF process proposed. Additional 68 critical test cases were identified as part of this conversion.
11.2.5 Automation Test Plan

Test Design Specifications and Test Cases Specifications documents are the major inputs to the automation test plan. The automation test plan also uses the QA test plan to synchronize automation releases with the QA and engineering releases. Based on the necessary level of regression for each feature and ROI gained by automation the test plan was developed.

Automation Test Plan consists of the following:

- Features that need to be automated
- The level of automation required for each feature
- Automation Strategy
- Automation Environment
- Automation schedules

11.2.6 Defining Automation Test Cases

Manual test case specification document was the input to define automation test cases for each feature.

Each manual test case was prioritized based on the following factors:

- Is the test case automatable
- What is the complexity of the test case
- What is the manual test effort required to execute the test case
- What is the hardware setup time required before executing a test case
- What is the execution frequency of the test case
The below figure shows that the major consideration to define the automation test case priority which ranges from 1 to 3.

![Figure 20: Automation Test Case Prioritization](image)

A total of 200 test cases were prioritized based on the above factors which took 2 man days to finish the task. The test cases were prioritized based on the QA and engineering inputs.

### 11.2.7 Agile testing

Agile testing process consists of developing automation test cases in short cycles of 3-4 man weeks. The challenge lies in defining the short cycles of development which is also known as sprints.
The priority 1 test cases which were relevant to the feature that has been developed to 90% were taken into consideration as the first sprint for automation of Project 3.

11.2.8 Manual Execution of the Automation Test Cases

The automation team manually executed the test cases to get a better understanding of the test cases before developing the automation scripts. The test cases were uploaded to quality center. The automation documents were in a format that quality center could directly import.

11.2.9 Developing Automation Test Cases

Once the test cases were completely understood by the automation team, the test cases were ready to be automated. Automation design meetings were held to design the test cases. Utilities and data sets were defined for the automation scripts. The atomic operation, sub test operations, test operations and test suites were developed and tested. Once the test cases were automated they were peer reviewed by the automation team for the best practices to be followed. All automation scripts were readable, scalable and maintainable as they have followed the 16 point checklist which was developed by automation lead with engineering manager’s support.

A total of 88 test cases were automated over a period of 1 man month with 2 automation resources.
11.2.10 Agile Demo

Once the test cases were automated an engineering and QA demo was done. In the demo the deliverables were explained. A live demo of the automation was given to various teams and the inputs were collected and implemented based on the priority.

11.2.11 Regressions

The automated test cases were run over every qualified QA build to see if any of the existing functionality was broken.

11.2.12 Return on Investment

Using the EAF process the automation was developed for Project 3. And after the first sprint of the automation the QA automation team found 35 defects which were extraordinary. The following figure depicts the ROI for 88 automation test cases of project 3. Out of 35 defects, 27 were P1 priority defects which were fixed by the engineering teams immediately. One intermediate releases was stopped to the customers due to the defects that were found by automation team following the EAF process. Project 1 did not follow the EAF and hence was able to find only 7 defects over a period of 2 years.
Below are the ROI calculations for the 35 defects found for Project 3.

With the above ROI the senior management was really convinced. EAF process was implemented for project 2 as well.

12 Tools and Resources

Following tools and resources have been used (and will be used):

- Quick Test Professional 9.5: This provides a functional, customer scenario and objective simulator integrated within QA Environment.
- Quality Center: This is most popular tool; it is used for test management.
• Martin Luther King Library: Technical papers, journals and books from MLK library were used for literature survey.

• Internet: Web sites such as yahoo finance, Wikipedia were used for economic survey and economic analysis.

• CMPE 287, Tech 230 and Tech 232 course material has been used to understand concepts such as state based testing, Six Sigma, and statistical control process.
13 Conclusion

The proposed ROI based QA frameworks provide complete low cost quality solution to the software industry. The company’s value proposition is unbeatable with the current market conditions under consideration. The company shows economic value and shall achieve a profit of $5.67M at the end of 3 years. The NPV is positive indicating the business will be profitable.

14 Future Improvements

The ROI based EAF could be further enhanced to include the statistical analysis of the process success. Statistical analysis could include Shewart charts. Currently the sample data is available for only 2 projects. As the process is implemented across various divisions and projects in HP, the data could be sampled to see how efficiently the process is working.

The ROI based EAF process could be improved to implement the six sigma and DMAIC principles for various kinds of testing performed by the QA team. The framework could be made generic in terms of utilities etc such that other divisions can easily adapt the process. This shall be implemented in the release 2.0 of the product.
15 Team and Committee

Prof. Jerry Gao: Dr. Jerry Gao is Professor in College of Engineering at San Jose State University. His specialization is in Software quality Assurances. He worked indifferent companies as a consultant and author and co-author of many books. He specializes in quality assurance process in manual and automation. His expertise was utilized to develop the Enterprise level software automation framework and guide me over the project.

Karen Burley: Karen Burly is director of Engineering, Quality Assurance and Customer Support in Hewlett Packard. In past she served as Vice President for Symantec Corporation. She is contributing to the software industry for more than 30 years now. Her vast knowledge on the product management, project management, QA and customer management is utilized to understand the requirements and analyze the data and finally develop/sell the project.

Nimesh Minawala: Nimesh Minawala is software Quality Assurance Manager in Hewlett Packard. In the past he has worked as a Sr. Quality Assurance Manager in Symantec. He is serving in software industry for more than 18 years. His vast QA knowledge and experience is used in developing the Enterprise level automation framework.
**Deepika Paruchuri:** Deepika Paruchuri is working as Quality Assurance Team Lead in Hewlett Packard Company and a part-time student of SJSU. She has been serving the quality assurance software industry from the last 9 years. Her immense experience in Quality Assurance will help to complete the project. She is also a member of American Society of Quality from Hewlett Packard Company.

16 References


https://workplaceservices100.fidelity.com/netbenefits/navstation/navigation

[13] Infosys Quarterly Results retrieved on February 10th 2009
http://www.infosys.com/investors/reports-filings/default.asp
Appendix A – Sample EAF Utilities

Recovery Scenarios

'Const LogFilePath = "C:\Automator"

Function CaptureBitMap(filename)
    Dim TimeStamp
    Dim CompleteFilename
    TimeStamp = rDateTime("_")
    CompleteFilename = "C:\Automator\Logs" & "\" & filename & TimeStamp & ".png"
    Desktop.CaptureBitmap CompleteFilename, false
    End Function

Function importRes( actionName, currentRowIndex )

    'msgbox actionName
    dataFile = "C:\Automator\DataSet\UI\AtomicOperation\" & actionName & ".xls"
superActionName = "Action1 [" & actionName & "]"

'msgbox superActionName

val = DataTable.ImportSheet( dataFile,  actionName, superActionName)
DataTable.GetSheet(superActionName).SetCurrentRow(currentRowIndex)

if( actionName = "example") then
    superActionName = "Action2 [" & actionName & "]"
    val = DataTable.ImportSheet( dataFile,  actionName, superActionName)
    DataTable.GetSheet(superActionName).SetCurrentRow(currentRowIndex)
    End If

End Function

Function SetDataRowForChildActions( sheetName, currentRowIndex )
    DataTable.GetSheet(sheetName).SetCurrentRow(currentRowIndex)
End Function

Public Function importDataFileToDataTable ( expName, currentName)
Dim returnValue
If expName <>currentName then
    Dim tempString, sheetName

actionName = environment("ActionName")

tempString = Split (actionName, "]", -1, 1)

'msgbox tempString(0)

'msgbox tempString(1)

    secondTempString = Split (tempString(1), "]", -1, 1)

    sheetName = secondTempString(0)

    returnValue = sheetName

importDataFileToDataTable = sheetName

Else

    returnValue = currentName

    importDataFileToDataTable = currentName

End If

'MsgBox returnValue & "Hi"
dataFileAbsolutePath = PathFinder.Locate(returnValue & ".xls")

Datatable.AddSheet returnValue

Datatable.ImportSheet dataFileAbsolutePath, 1, returnValue

End Function

Public Function importDataFileToDataTableNoCheck ( sheetName )

'MsgBox returnValue & "Hi"
dataFileAbsolutePath = PathFinder.Locate(sheetName & ".xls")

Datatable.AddSheet sheetName

Datatable.ImportSheet dataFileAbsolutePath, 1, sheetName

End Function

Public Function importDataFileToDataTableGlobal ( expName, currentName) As String

Dim returnValue

If expName <> currentName then

    Dim tempString, sheetName
    actionName = environment("ActionName")
    tempString = Split (actionName, "[", -1, 1)
    'msgbox tempString(0)
    'msgbox tempString(1)

    secondTempString  = Split (tempString(1), "]",-1,1)
    sheetName = secondTempString(0)
    returnValue = sheetName

    importDataFileToDataTableGlobal = sheetName

Else

    returnValue = currentName

    importDataFileToDataTableGlobal = currentName

End If
'MsgBox returnValue & "Hi"

dataFileAbsolutePath = PathFinder.Locate(returnValue & ".xls")

Datatable.AddSheet "Global"

Datatable.ImportSheet dataFileAbsolutePth, 1, "Global"

End Function

Function CheckError(Object, Method, Arguments, retVal)

    CaptureBitMap(environment("ActionName"))

    LogWrapper "FAILURE", "ERROR", " - " & " Atomic Operation: " & environment("TestName") & " - " & environment("ActionName") & " Failed"

End Function


************************************************************************
*************Logs
Const ForWriting = 2
Const ForAppending = 8

Const INFO = "INFO - 
Const ERROR1 = "ERROR - 
Const WARNING = "WARNING - 

Const START = "c:\automator\Logs\test.started.txt"
Const STATUS = "c:\automator\Logs\test.status.txt"
Const FAILURE = "c:\automator\Logs\test.failed.txt"
Const SUCCEED = "c:\automator\Logs\test.succeeded.txt"

Sub LogWrapper(fileName, msgType, details)
    Select Case fileName
        Case "START" LogStart msgType, details
        Case "STATUS" LogStatus msgType, details
        Case "FAILURE" LogFail msgType, details
    End Select
End Sub
Case "SUCCEED"    LogSucceed msgType, details

Case Else    LogStatus msgType, details

End Select

End Sub

'***********************************************************************
*********************
' Function:    LogStart
' Paramaters:
' Description:   Log the start message

'***********************************************************************
*********************

Sub LogStart(msgType, info)
     Log START, msgType, info

End Sub

'***********************************************************************
*********************
' Function:    LogStatus
' Paramaters:
' Description:   Log the statusmessage
Sub LogStatus(msgtype, info)
    Log STATUS, msgtype, info
End Sub

Sub LogFail(msgtype, info)
    Log FAILURE, msgtype, info
End Sub

Sub LogSucceed(msgtype, info)
    Log SUCCESS, msgtype, info
End Sub
Sub LogSucceed(msgtype, info)
    Log SUCCEED, msgtype, info
End Sub

Sub Log(statusfile, msgtype, info)
    Dim msg
    msg = GenLogMsg(msgtype, info)
    LogToFile statusfile, ForWriting, msg
    LogToFile "c:\automator\Logs\log.txt", ForAppending, msg
End Sub
'Wait 1 seconds for automator's agent to update

Wait(1)

End Sub 'Log

***********************************************************************
***********************************************************************

Function GenLogMsg

Paramaters:

Description: Generate the log message

***********************************************************************
***********************************************************************

Function GenLogMsg(msgtype, msg)

If Not (msg = "") Then

    Dim datetime

    datetime = rDateTime(".")

    msg = datetime & "," & msgtype & msg

End If

GenLogMsg = msg
End Function

'***********************************************************************
********************
' Function: LogToFile
' Parameters:
' Description: Log the result to file
'***********************************************************************

Sub LogToFile(sLogFile, logtype, msg)
    Set oLogFSO = CreateObject("Scripting.FileSystemObject")
    Set oLogFile = oLogFSO.OpenTextFile(sLogFile, logtype, True)
    oLogFile.WriteLine(msg)
    oLogFile.Close
    oLogFSO = Null
End Sub

'***********************************************************************
********************
' Function: rDateTime
' Parameters:
' Description: Create the time stamp in the form YYYY.MM.DD hh:mm:ss
Function rDateTime(delimiter)

    Dim rDate, rTime

    DIM myDay, myMonth, myYear, myTime
    DIM myHour, myColon, myMinute, mySecond

    rDate = Date()
    rTime = Time()

    'get AM or PM from the time
    myAMPM = Right(rTime, 2)

    'get Hour and Minute
    myColon = instr(1, rTime, ":", vbBinaryCompare)

    IF myColon = 2 THEN
        myHour = Left(rTime, 1)
        myMinute = Mid(rTime, 3, 2)
        mySecond = Mid(rTime, 6, 2)
    ELSE
        myHour = Left(rTime, 2)
        myMinute = Mid(rTime, 4, 2)
mySecond = Mid(rTime, 7, 2)

END IF

If (myAMPM = "PM") Then
    myHour = myHour + 12
    End If

'concatenate the time
myTime = myHour & delimiter & myMinute & delimiter & mySecond

'get the day, e.g. oct 19, 2002, will return 19
myDay = Day(rDate)

'get month number
myMonth = Month(rDate)
    If (myMonth < 10) Then
        myMonth = "0" & myMonth
        End If

'get the year
myYear = Year(rDate)

'concatenate variables
'2007.01.16 14.41.36

rDateTime = myYear & delimiter & myMonth & delimiter & myDay & delimiter & myTime

End Function 'rDateTime

***********************************************************************
*************For outlook scripts to import data sheets
***********************************************************************

Sub ImpDiffXLTOAO(dtFileName, dtType, DestSourceShtTyp, FDestShtName, FSourceShtName )
    Dim DTFilepath, DTSourceShtName, DTDestShtName
    If dtType = "AO"Then
        DTFilepath = Pathhash.item("DT_path") & dtFileName & ".xls" ' AO dataset paths
    End If
End Sub
Elseif dtType = "TO" Then

    DTFilepath = Pathhash.item("DTto_path") & dtFileName & ".xls" 'TO dataset paths

Elseif dtType = "OTF" Then

    DTFilepath = Pathhash.item("AO_path") & dtFileName & ".xls" ' Lib dataset (send mail timestamp and send calendar timestamp path)

Else

    LogWrapper "Error - ", "Wrong function 'ImportDiffXLTOAO' call", "STATUS"

End If

If DestSourceShtTyp = "1" Then

    DTSourceShtName = FSourceShtName

    DTDestShtName = FDestShtName

Elseif DestSourceShtTyp = "2" Then

    DTSourceShtName = FSourceShtName

    DTDestShtName = FDestShtName & "[" 

    FDestShtName & "]"

End If

' msgbox DTFilepath & "   " & DTSourceShtName & "   " & DTDestShtName

Datatable.ImportSheet DTFilepath, DTSourceShtName, DTDestShtName
LogWraper "INFO - ", "Executing Sub : ImpDiffXLTOAO. Data table Import is successful from path : " & DTFilepath & ". Source Sheet Name : " & DTSourceShtName & ", Destination Sheet Name : " & DTDestShtName, "STATUS"
End Sub

'#######################################################################
#######################################################################
'Function Name          :      LoadINIFiles
'Description          :       This Sub read O2K3_path.ini file and loads the Hash variable with Relative path for Libraries, DataSet, Result File etc.
'#######################################################################
#######################################################################
Public Pathhash

Sub LoadINIFiles ( )

    Dim filesys, filetxt, filesystxt, INIPath, INIPathAbs, TxtLine, INIVal

    LogWraper "INFO - ", "Executing Sub : LoadINIFiles. The paths in INI files will be loaded into Pathhash variable", "STATUS"

    Set Pathhash= CreateObject("Scripting.Dictionary")

    INIPath = "C:\Automator\Lib\Client\O2K7_Lib\"


Set filesys = CreateObject("Scripting.FileSystemObject")

Set filesystxt =
filesys.CreateTextFile("C:\Automator\Lib\Client\O2K7_Lib\Txtfile\INI_Paths.txt", True)

INIpathAbs = INIPath & "O2K7_paths.ini"
If filesys.FileExists(INIpathAbs) Then
  Set filetxt = filesys.OpenTextFile(INIpathAbs, 1)
  Do While filetxt.AtEndOfStream <> True

  TxtLine = Trim(filetxt.ReadLine)

  If TxtLine <> "" AND Left(TxtLine,1) <> "#" Then
    If InStr(TxtLine, "#") > 0 Then
      TxtLine = Left(TxtLine, InStr(TxtLine, "#") - 1)
    End If

    INIVal = Split (TxtLine, "=")
    If TRIM(INIVAL(0)) <> "" AND TRIM(INIVAL(1)) <> "" Then
      Pathhash.Add TRIM(INIVAL(0)), TRIM(INIVAL(1))
    End If
  End If

  End If

End If
LOOP

Else

Msgbox "O2K3_path.ini file does not exits : Check your files"

End If

For Each Key1 IN Pathhash.Keys

filesystxt.WriteLine(key1 & "=" &Pathhash.item(key1))

Next

filetxt.close

filesystxt.Close

Set filesys = nothing

Set filetxt = nothing

Set filesystxt = nothing

End Sub

'#######################################################################
'#####################################################################
'File Name           :     PrepRun ( )
'Description : Prepare Test run by setting QTP priority to High. Kill all running process. Check all required file exits.

Public programname, priority

Sub PrepRun (TOfile_name )

Dim LH_Value, MOS_Value

LoadINIFiles () ' Sub Call for loading paths of attachment, Subject_TimeStamp path etc

programname= "QTPro.exe"

priority = 128

If SetTaskPriority(programname , priority)= 0 then

LogWrapper "INFO - ",programname&" process priority is set to " & GetPriority(priority) & " successfully in Task Manager","STATUS"

Else

LogWrapper "ERROR - ","Setting priority of "&programname&" process in Task Manager to priority " & GetPriority(priority) & " has Failed","STATUS"

End If
'Computer details on which Test is run

    LH_Value = Environment.Value("LocalHostName")
    MOS_Value = Environment.Value("OS")

    LogWrapper "INFO - ", "Test Operation is running on computer" & " " & LH_Value &" and on OS : "&MOS_Value, "STATUS"
    Reporter.ReportEvent micDone,
    Environment.Value("LocalHostName"), "Test Operation is running on computer : " & LH_Value &" and on OS : "&MOS_Value

    KillRunningProcess( ) ' Sub call to kill all process

'Check all the required AO's exist before running the script.

    CheckAOExist TOfile_name 'Sub call

End Sub

'###########################################################################
'###########################################################################
'File Name           :     PrepRun ( )
'Description          :     Kills any running process like OUTLOOK.exe, WORD.exe. Also prints Computer name and OS to log file
Sub KillRunningProcess()

    'Kill any Running Outlook process before opening new instance of Outlook.

    If SystemUtil.CloseProcessByName("WINWORD.exe") Then

        LogWrapper "WARNING - ", "Killed WINWORD Processes from CleanUp lib", "STATUS"

    End If

    If (NOT Window("Microsoft Outlook") IS Nothing) AND Window("Microsoft Outlook").Exist(3)Then

        hwnd = Window("Microsoft Outlook").GetROProperty("hwnd")

    End If

    If SystemUtil.CloseProcessByHwnd(hwnd) Then

        LogWrapper "WARNING - ", "Killed Outlook Processes and all its child window by logic hwnd from CleanUp lib", "STATUS"

    End If
End If

If SystemUtil.CloseProcessByName("OUTLOOK.exe")

Then

LogWrapper "WARNING - ", "Killed Outlook Processes from CleanUp lib", "STATUS"

End if

wait 1

End Sub

'########################################################################
'########################################################################
'File Name           :     PopulateDtColNames
'Description          :     Reads column heading of the active Dataset xls file and load the header to array.
'########################################################################
'########################################################################

Sub PopulateDtColNames ( ExcelFile, sheetName, subjectRow, startCol, ByRef rsultArray)

Dim col, objExcel, objSheet, Objws
LogWrapper "INFO - ", "Executing Sub : PopulateDtColNames. Read the columns heading of Data table to Array variable.", "STATUS"

Set objExcel = CreateObject("Excel.Application")
Set objSheet = objExcel.Workbooks.Open(ExcelFile)
Set Objws = objExcel.WorkSheets(sheetName)

col = startCol

Do Until (Objws.Cells(subjectRow,col).Value) = ""

Redim preserve rsultArray(col)

rsultArray(col) = Objws.Cells(subjectRow,col).Value

col = col + 1

Loop

objExcel.Quit

If col = 1 Then

    capture_desktop( )

    KillProcess("OUTLOOK.exe")

    LogWrapper "ERROR - ", "There is no data in excel file : " & ExcelFile & ", Sheet : " & sheetName , "FAILURE"

    ExitTest

End If

Set objExcel = Nothing
Set objSheet = Nothing
Set Objws = Nothing
End Sub

Sub PopulateHashFromDt (ExcelFileName, SheetName, ByRef mainHash, filename)

    Dim rowHash, interimHash, rowHash1, countIndex, anotherhash, Rows, Cols, rowcount, dtCols( )

    LogWrapper "INFO - ", "Executing Sub : PopulateHashFromDt. Load the data table values to Hash variable based on the designed loop logic. ", "STATUS"

'}
PopulateDtColNames ExcelFileName, SheetName, 1, 1, dtCols

' For I = Lbound(dtCols)+1 to Ubound(dtCols)

' msgbox "dtCols("&I&")="&dtCols(I)

' Next

Cols = Datatable.GetSheet(SheetName &" ["& SheetName &"]").GetParameterCount

Rows = Datatable.GetSheet(SheetName &" ["& SheetName &"]").GetRowCount

' Cols = Datatable.GetSheet(SheetName).GetParameterCount

' Rows = Datatable.GetSheet(SheetName).GetRowCount

If Rows > 15 Or Cols > 17 Then

' Stop test and check DataSet

LogWrapper "ERROR - ","Check # of Row/Columns in DataSet and delete blank Rows/Columns if necessary. QTP reads Rows: " & Rows & " and Columns as : " & Cols,"STATUS"

StopRun ""

End If

Set mainHash = CreateObject("Scripting.Dictionary")
ColIndex = Cols
Emailcount = 0
Do While Colindex >= 1
    Datatable.SetCurrentRow 1
    tblCelltxt = trim(Datatable(Colindex, dtLocalSheet))
    If tblCelltxt <> "" Then
        If mainHash.count <= 0 Then ' If mainHash is empty enter If condition
            For I = 1 To Rows
                Set rowHash = CreateObject("Scripting.Dictionary")
                Datatable.SetCurrentRow I
                tblCelltxt = trim(Datatable(Colindex, dtLocalSheet))
                If tblCelltxt="" Then
                    Exit For
                End If
                rowHash.Add dtCols(Colindex), tblCelltxt
            End For
            Emailcount = Emailcount + 1
            mainHash.Add filename & "_" & Emailcount, rowHash
        End If
    End If
End Do
Set rowHash = nothing
Next
Else

Set anotherHash =

CreateObject("Scripting.Dictionary")

Set rowHash1 = CreateObject("Scripting.Dictionary")

For

rownum = 1 To Rows

Datatable.SetCurrentRow rownum

tblCelltxt = trim(Datatable(Colindex,dtLocalSheet))

If tblCelltxt="" Then

Exit For

End If

For Each key1 In mainHash.keys
Set interimhash = CreateObject("Scripting.Dictionary")

Datatable.SetCurrentRow rownum

tblCelltxt =
trim(Datatable(Colindex,dtLocalSheet))

If tblCelltxt <> "" Then

interimhash.add dtCols(Colindex),tblCelltxt

For Each AddCacheKey In mainHash.item(key1).keys

    interimhash.add AddCacheKey,
mainHash.item(key1)(AddCacheKey)

Next
rowcount = rowcount + 1

anotherhash.add filename & "_" & rowcount, interimhash

Set interimhash = nothing

End If

Next

Next

mainHash.Removeall

For Each key17 In anotherhash.keys

mainHash.add key17, anotherhash(key17)

Next

Set anotherhash = nothing

End If

End If

Colindex = Colindex - 1

rowcount = 0

Loop

Print_HashToFile mainhash, filename
Sub Print_HashToFile (ByRef mainhash, filename)

    Dim  filesysHash

    LogWraper "INFO - ", "Executing Sub : Print_HashToFile. Print data table Hash variable to text file.", "STATUS"
    Set filesysHash = CreateObject("Scripting.FileSystemObject")
    Set filetxtHash =

    filesysHash.CreateTextFile(Pathhash.item("Lib_path") & "Txtfile" & filename & ".txt", True)

    filetxtHash.WriteLine("Email Iterations : ") & vbcrlf

    For Each key1 In mainhash.keys
        filetxtHash.WriteLine(key1)
    Next key1
End Sub
For Each key2 In mainhash.item(key1)
    filetxtHash.WriteLine("   " & key2 & ":" & mainhash.item(key1)(key2))
    Next
    filetxtHash.WriteLine(""")
    Next
    filetxtHash.close

Set filesysHash = nothing
Set filetxtHash = nothing
End Sub

'#######################################################################
'#####################################################################
'Function Name          :      SetGetError
'Description          :        GlobalErrorNum variable holds the total #
of errors in check mail script
'#######################################################################
'#####################################################################
Public GlobalErrorNum
GlobalErrorNum = 0
Sub SetGetError (ErrorNum )
GlobalErrorNum = ErrorNum + GlobalErrorNum

End Sub

Sub CreateDelFile (XLfilename, XLpath, Create_del)
    Dim xlApp, xlBook

    Set FSO = CreateObject("Scripting.FileSystemObject")

    'Check if the file exist
    If Create_del = "VerifyXLSfile" Then
        If FSO.FileExists(XLpath & XLfilename&".xls") Then
            'Dont do anything
        Else
            'Some code here
        End If
    Else
        'Some code here
    End If
End Sub
capture_desktop() '  
LogWrapper "ERROR - ", "Xls files does not exits in path : " & XLpath & XLfilename & ".xls. Please check the file and continue. ", "FAILURE"
KillProcess("OUTLOOK.exe")
ExitTest
End If
End If

'Delete file if exist
If Create_del = "DeleteXLSfile" Then
If FSO.FileExists(XLpath & XLfilename & ".xls") Then
    FSO.DeleteFile XLpath & XLfilename & ".xls"
    LogWrapper "INFO - ", "Subject timestamp xls file is deleted from path : " & XLpath & XLfilename & ".xls" , "STATUS"
End If
End If

Set FSO = Nothing

'Create Xls file.
If Create_del = "CreateXLSfile" Then
Set xlApp = CreateObject("Excel.Application")
Set xlBook = xlApp.Workbooks.Add
xlBook.SaveAs(XLpath & XLfilename&".xls")
LogWrapper "INFO - ", "Blank xls file is created in path : 
& XLpath & XLfilename&".xls", "STATUS"
xlApp.Quit
Set xlApp =nothing
Set xlBook =nothing
End If

End Sub

'#######################################################################
'Function Name :       SetTaskPriority
'Description :        Set the priority of  Outlook.exe and Qtppro.exe
'#######################################################################

Function SetTaskPriority(programName, priority)
Const NORMAL_PRIORITY  =  32
Const IDLE =  64
Const HIGH_PRIORITY =  128
Const REALTIME  =  256
Const BELOW_NORMAL = 16384
Const ABOVE_NORMAL = 32768

strComputer = "."
SetTaskPriority = -1
'Wscript.echo programName & priority
Set objWMIService = GetObject("winmgmts:" &
"{impersonationLevel=impersonate}!\" & strComputer & "\root\cimv2")
Set colProcesses = objWMIService.ExecQuery("Select * from
Win32_Process")

For Each objProcess in colProcesses

'Wscript.echo objProcess.Name
strName = replace(objProcess.Name, vbCrLf, ")
If Lcase(trim(strName)) = LCase(programName) Then

'Wscript.echo "I am here"
SetTaskPriority =
objProcess.SetPriority(priority)

End If

Next
End Function
Function GetPriority(priority )

Select Case priority
    Case 32
        GetPriority = "NORMAL"
    Case 64
        GetPriority = "IDLE"
    Case 128
        GetPriority = "HIGH PRIORITY"
    Case 256
        GetPriority = "REALTIME"
    Case 16384
        GetPriority = "BELOW NORMAL"
    Case 32768
        GetPriority = "ABOVE NORMAL"
End Select
End Function
'Function Name : ErrorCheck

'Description : Print Error message to Log file if error exist.

'#########################################################################

Sub ErrorCheck (ActionName, ErDescription, ErNumber )

If ErNumber <> 0 Then
    LogWraper "ERROR - ",  Error Loading Action : ", & ActionName & ".
Error Description : ", & ErDescription ,"STATUS"
    Err.Clear
    KillProcess("OUTLOOK.exe")
    LogWraper "ERROR - ", "Atomic Operation scripts not found or Datatable Import failed. Please check AO scripts/Datatable and continue.",
"FAILURE"
    ExitTest
    'msgbox ErNumber
    End If
End Sub

'#########################################################################

'Function Name : AOFileExists

'Description : Check whether folder exist or not.
Sub AOFileExists(AOfilename)
    dim filesys
    Set filesys = CreateObject("Scripting.FileSystemObject")
    If filesys.FolderExists(Pathhash.item("AO_path") & AOfilename)
    Then
        'don't do anything\n    Else
        capture_desktop() '
        KillProcess("OUTLOOK.exe")
        LogWrapper "ERROR - ", "Atomic Operation : " & AOfilename & " scripts not found. Please check AO scripts and continue.", "FAILURE"
        ExitTest
    End If
    Set filesys = Nothing
End Sub

'Function Name          :       CheckAOExist
'Description : Populate AO file names from text file to an
Array. Make a call to AOFexists sub

Public AONames

Sub CheckAOExist (TOname)

    Dim filesys, filetxt, AOlistfile, TxtLine

    LogWraper "INFO - ", "Executing Sub : CheckAOExist. Check all
required AO's are present on the test computer, beforing test starts to run. ", "STATUS"

    Set filesys = CreateObject("Scripting.FileSystemObject")

    AOlistfile = Pathhash.item("TO_path") & TOname
    &"\AtomicOperation.txt"

    counter = 0

    ReDim AONames(counter)

    If filesys.FileExists(AOlistfile) Then

        Set filetxt =

        filesys.OpenTextFile(AOlistfile, 1)

        Do While filetxt.AtEndOfStream <> True

            TxtLine =

            Trim(filetxt.ReadLine)
ReDim Preserve AONames(counter)

If TxtLine <> "" Then

    AONames(counter) = TxtLine

    counter = counter + 1

End If

LOOP

filetxt.close

Else

    KillProcess("OUTLOOK.exe")

    LogWrapper "ERROR - ","AtomicOperation.txt file NOT FOUND in TO : " & TOname & ". Check you files and continue to Run scripts." ,"FAILURE"

    ExitTest

End If

Set filesys = nothing

Set filetxt = nothing

For i= Lbound(AONames) to Ubound(AONames)
' msgbox AONames(i)
AOFileExists (AONames(i) ) ' Sub call

Next

End Sub

'#######################################################################
'Function Name     :      KillProcess
'Description     :       Kills the given process from Task Manager
'#######################################################################

Function KillProcess(strProcessKill)

    Dim objWMIService, objProcess, colProcess, strComputer

    strComputer = "." 

    Set objWMIService = GetObject("winmgmts:" & 
         "\{impersonationLevel=impersonate}!\" & strComputer & "\root\cimv2")


Set colProcess = objWMIService.ExecQuery("Select * from Win32_Process Where Name =" & strProcessKill & "")

For Each objProcess in colProcess
    objProcess.Terminate()
Next

LogWrapper "ERROR - ", "Process killed : " & strProcessKill
,"STATUS"

'WScript.Echo "Just killed process " & strProcessKill & " on " & strComputer

'WScript.Quit

End Function

'#######################################################################
'File Name           :      Logout ( )
'Description          :     Logout of Outlook application.
'#######################################################################
Sub Logout()

    Window("Microsoft Outlook").Close

    Wait 2

    If Window("Microsoft Outlook").Exist (2) Then
        Wait 10

            If Window("Microsoft Outlook").Exist (5) Then
                Window("Microsoft Outlook").Close
                LogWrapper "INFO - ", "Logged out of Outlook profile", "STATUS"
                Reporter.ReportEvent micPass,
                "Outlook Logout", "Logged out of Outlook profile"

            Else

                LogWrapper "Error - ", "Microsoft outlook is still open. Close outlook and continue.", "STATUS"

            End If

    End If

End Sub
Sub RestartOutlook( )

    LogWraper "INFO - ", "Executing Sub : RestartOutlook. Handle message box quit and restart Outlook. . ", "STATUS"

    'Handle Message - The Microsoft Exchange administrator has made a change that require you quit and restart Outlook.

    If Dialog("text:=Microsoft Office Outlook").Exist Then
        Dialog("text:=Microsoft Office Outlook").Activate
        Dialog("text:=Microsoft Office Outlook").Click

    msgval = Dialog("text:=Microsoft Office Outlook").GetROProperty("text")

    Dialog("text:=Microsoft Office Outlook").WinButton("text:=OK").Click
LogWrapper "INFO - ", "Handled Message : 
"&msgval, "STATUS"

Reporter.ReportEvent micPass, " RestartOutlook(
), "Handled Message : 
"&msgval

Else

LogWrapper "INFO - ", "Unable to handle Exchange
administration message popup.", "STATUS"

Reporter.ReportEvent micWarning, 
"RestartOutlook( ). ", "Exchange administration has made changes to Outlook message 
was not handled"

End If

LogWrapper "INFO - ", "Completed execution of RestartOutlook( ) sub.", 
"STATUS"

execution of RestartOutlook( ) sub"

End Sub

'#######################################################################
'#####################################################################
'Function Name          :      Get_RUEmailCount
'Description': Get Email Read / Unread emails count in Inbox.

Public: unreadIN, readIN, unreadPF, readPF, unreadMT, readMT, Totalemail

Sub Get_RUEmailCount(Types)

Const olFolderInbox = 6
Const olFolderCal = 9
Const AllPublicFolders = 18

Set Application = CreateObject("Outlook.Application")
Set objNamespace = Application.GetNamespace("MAPI")

If Types = "INBOXPF" Then

    Set objInboxPF =
    objNamespace.GetDefaultFolder(olFolderInbox)

    Totalemail = objInboxPF.Items.Count
    'Get total number of emails in Inbox.
    If Totalemail >= 0 Then
        LogWrapper "INFO - ", "Total number of emails in Inbox are: " & Totalemail, "STATUS"
    Else
LogWrapper "ERROR - ", "

Number of Email in Inbox are : " & Totalemail , "STATUS"

ErrorNum = ErrorNum+1

End if

Set colItemsInbox = objInboxPF.Items

'Unread count in Inbox.

Set colFilteredItemsInb = colItemsInbox.Restrict("[Unread]=true")

If colFilteredItemsInb.Count >= 0 Then
    LogWrapper "INFO - ", "

    Number of Unread Email in Inbox are : " & colFilteredItemsInb.Count , "STATUS"

    unReadIN =

    colFilteredItemsInb.Count

Else
    LogWrapper "ERROR - ", "

    Number of Unread Email in Inbox are : " & colFilteredItemsInb.Count , "STATUS"

    ErrorNum = ErrorNum+1

End if

Set colFilteredItemsInb = colItemsInbox.Restrict("[Unread]=False")

If colFilteredItemsInb.Count >= 0 Then
LogWrapper "INFO - ", "
Number of Read Email in Inbox are : " & colFilteredItemsInb.Count , "STATUS"

readIN =
colFilteredItemsInb.Count

Else
LogWrapper "ERROR - ", "
Number of Read Email in Inbox are : " & colFilteredItemsInb.Count , "STATUS"

ErrorNum = ErrorNum+1

End if

Datatable.GetSheet("Global").SetCurrentRow 1

PF_FoldName = DataTable.GetSheet("Global").GetParameter("PF_FolderName")

Set objPF = objNamespace.GetDefaultFolder(AllPublicFolders).

 objPF = " " Then
LegWrapper "ERROR - ", "Public Folder does not exist. Create PF on Outlook client or check the dataset ", "STATUS"

ErrorNum = ErrorNum+1

End If

Set colItemsPF = objPF.Items

Set colFilteredItemsPF = colItemsPF.Restrict("[Unread]=true")
If colFilteredItemsPF.Count >= 0 Then
    LogWrapper "INFO - ", "

    Number of Unread PF Email are : " & colFilteredItemsPF.Count, "STATUS"

    unreadPF = colFilteredItemsPF.Count

    Else

    LogWrapper "ERROR - ", "

    Number of Unread PF Email in are : " & colFilteredItemsPF.Count, "STATUS"

    'ErrorNum = ErrorNum+1

    End if

Set colFilteredItemsPF = colItemsPF.Restrict("[Unread]=False")

If colFilteredItemsPF.Count >= 0 Then
    LogWrapper "INFO - ", "

    Number of Read PF Email are : " & colFilteredItemsPF.Count, "STATUS"

    readPF = colFilteredItemsPF.Count

    Else

    LogWrapper "ERROR - ", "

    Number of Read PF Email are : " & colFilteredItemsPF.Count, "STATUS"

    'ErrorNum = ErrorNum+1

    End if
Set objInboxPF = Nothing
Set colItemsInbox = Nothing
Set colFilteredItemsInb = Nothing
Set objPF = Nothing
Set colItemsPF = Nothing
Set colFilteredItemsPF = Nothing
End If

SetGetError ErrorNum ' Sub call
Set Application = Nothing
Set objNamespace = Nothing
End Sub

'#######################################################################
'Function Name          :      PrintEmailCnt_ToFile
'Description          :       Print  Read / Unread emails count to file
'#End Sub

'#######################################################################
Sub PrintEmailCnt_ToFile (IN_UnReadCt, IN_ReadCnt, Types, PF_UnReadCt, PF_ReadCnt, PrintTo_filename, IN_Totalemail)

    Dim fso, filetxt

    'LoadINIFiles ( )

    Set fso= CreateObject("Scripting.FileSystemObject")

    If Types = "INBOXPF" Then

        Set filetxt =
        fso.CreateTextFile(Pathhash.item("TO_path")&"TO_O2K7_SendEmails_20001\Email_Cnt"&PrintTo_filename&".txt",True)

        'msgbox

        Pathhash.item("AO_path")&"AO_O2K3_SendMail\Email_Cnt"&PrintTo_filename&".txt"

        filetxt.WriteLine("Total Number of email in Inbox = ") & IN_Totalemail

        filetxt.WriteLine("Unread Inbox Email Count = ") & IN_UnReadCt

        filetxt.WriteLine("Read Inbox Email Count = ") & IN_ReadCnt

        filetxt.WriteLine("Unread PF Email Count = ") & PF_UnReadCt

        - 116 -
filetxt.WriteLine("Read PF Email Count = ") & PF_ReadCnt

filetxt.close

End If

Set fso= Nothing
Set filetxt = Nothing
End Sub

'#######################################################################
#####################################################################
'Function Name          :      ReadEmailCnt_FromFile
'Description          :       Print  Read / Unread emails count  to file
'#######################################################################
#####################################################################
Sub RUEmailCnt_Validate (UnReadCt, ReadCnt, Types, PF_UnReadCt, PF_ReadCnt, ReadFrom_filename, TotalemailCt, logcounter)

Dim filesys, filetxt, filepath, ErrorNum

'LoadINIFiles ( )

If Types = "INBOXPF" Then
filepath = Pathhash.item("TO_path")&"TO_O2K7_SendEmails_20016\Email_Cnt" &ReadFrom_filename& ".txt"

LogWrapper "INFO - "," Email and Public Folder emails Read and Unread count Verification ","STATUS"

End if

Set filesys= CreateObject("Scripting.FileSystemObject")

If filesys.FileExists(filepath) Then

Set filetxt =

filesys.OpenTextFile(filepath, 1)

Do While filetxt.AtEndOfStream <> True

TxtLine =

Trim(filetxt.ReadLine)

INIVal = Split

(TxtLine, ";")

If TRIM(INIVAL (0)) <> "" AND TRIM(INIVAL(1)) <> ""

Then

If TRIM(INIVAL(0)) = "Total Number of email in Inbox" Then
If TRIM(TotalemailCt) = TRIM(INIVAL(1)) OR TRIM(TotalemailCt+1) = TRIM(INIVAL(1)) Then

LogWraper "INFO - ", "Total number of emails in Inbox MATCHES and the value is : " & UnReadCt,"STATUS"

Else

LogWraper "ERROR - ", "Total number of emails in Inbox DOES NOT MATCH. Total Email value in file is :
"&TRIM(INIVAL(1)) &". Total Email count from Inbox is : "& UnReadCt,"STATUS"

ErrorNum = ErrorNum+1

End If

End If

If TRIM(INIVAL(0)) = "Unread Inbox Email Count" Then

If TRIM(UnReadCt) = TRIM(INIVAL(1)) Then
LogWrapper "INFO - ","

Unread Inbox Count MATCHES and the value is : " & UnReadCt,"STATUS"

Else

If TRIM(UnReadCt+1) = TRIM(INIVAL(1)) Then

LogWrapper "INFO - "," Unread Inbox Count MATCHES and the value is : " & UnReadCt,"STATUS"

Else

LogWrapper "ERROR - "," Unread Inbox Count DOES NOT MATCH. Unread value in file is : ",&TRIM(INIVAL(1)) ",&. Unread Email count from Inbox is :
"& UnReadCt,"STATUS"

ErrorNum = ErrorNum+1
ReadDumpEmail "Emaildump1" & logcounter

End If

End If

End If

If TRIM(INIVAL(0)) = "Read Inbox Email Count" Then

If TRIM(ReadCnt) = TRIM(INIVAL(1)) Then

LogWraper "INFO - "," Read Inbox Count in Inbox and file MATCHES and the value is : " & ReadCnt,"STATUS"

Else

If TRIM(ReadCnt-1) = TRIM(INIVAL(1)) Then

Then
LogWrapper "INFO - 
"

,"

Read Inbox Count MATCHES and the value is : " & ReadCnt,"STATUS"

Else

LogWrapper "ERROR - 
"

,"

Read Inbox Count DOES NOT MATCH. Read value in file is :

"&TRIM(INIVAL(1)) &". Read Email count from Inbox is : " & ReadCnt,"STATUS"

ErrorNum =

ErrorNum + 1

ReadDumpEmail

"Emaildump1" & logcounter

End If

End If

End If
If TRIM(INIVAL(0)) = "Unread PF Email Count" Then

If TRIM(PF_UnReadCt) = TRIM(INIVAL(1)) Then

LogWrapper "INFO - ", Unread PF Count MATCHES and the value is : " & PF_UnReadCt,"STATUS"

Else

LogWrapper "ERROR - ", Unread PF Count DOES NOT MATCH. Unread value in file is : ", Unread Email count from Inbox is : " & PF_UnReadCt,"STATUS"

'ErrorNum = ErrorNum+1

End If

End If

If TRIM(INIVAL(0)) = "Read PF Email Count" Then
If TRIM(PF_ReadCnt) = TRIM(INIVAL(1)) Then

    LogWrapper "INFO - ","Read PF
    Count MATCHES and the value is : " & PF_ReadCnt,"STATUS"

Else

    LogWrapper "ERROR - ","Read PF
    Count DOES NOT MATCH. Read value in file is : "&TRIM(INIVAL(1)) &". Read
    Email count from Inbox is : "& PF_ReadCnt,"STATUS"

    'ErrorNum = ErrorNum+1

End If

End If

End If

LOOP

filetxt.close

Else
LogWrapper "ERROR - ","The email count files does not exist in path 
"&filepath,"STATUS"

      ErrorNum = ErrorNum+1

      End if

      SetGetError ErrorNum

      Set fso= nothing

      Set filetxt = nothing

      End Sub

'#######################################################################
'Function Name          :      ReadEmailCnt_FromFile
'Description          :       Print Read / Unread emails count to file
'#######################################################################

Sub SearchEmail (dtSubject )

      Dim myItem, objOutlook, myNameSpace, ClientFolder, colItems,
      colFilteredItems, objOutlookMsg, found, loopcount
Set objOutlook = CreateObject("Outlook.Application") ' Create an Outlook object to interact with.

Set myNameSpace = objOutlook.GetNameSpace("MAPI")
Set ClientFolder = myNameSpace.GetDefaultFolder(6)
Set colItems = ClientFolder.Items

found = 0
loopcount = 0
While found = 0 and loopcount < 15
    Set colFilteredItems = colItems.Restrict("[Subject] = '" & dtSubject & "'")
    If colFilteredItems.count > 0 Then
        For Each myItem In colFilteredItems
            If myItem.Subject = "" Then
                Exit For
            End If
        End For
        If dtSubject = myItem.Subject Then
            'email found in Inbox
        End If
    End If
Loopcount = loopcount + 1
Wend
LogWrapper "INFO - ","FOUND email in Inbox : " & dtSubject
,"STATUS"

found =1

Exit For

End

If

Next

End if

Window("Microsoft Outlook").Type micF9
Wait 20
loopcount = loopcount + 1
Wend

PFRes = SearchPFemail ( dtSubject) ' Sub call

If found =1 Then

LogWrapper "INFO - ","FOUND email in Inbox. Continue to take Email counts","STATUS"

If PFRes = 0 Then

capture_desktop( )

- 127 -
KillProcess("OUTLOOK.exe")

LogWrapr "ERROR - ","Emails sent were NOT SYNCHRONIZED in Public Folder. Email Read/Unread count does not match."","FAILURe"

ExitTest

End If

Else

If PFRes = 0 Then

LogWrapr "ERROR - ","Emails sent were NOT SYNCHRONIZED in Public Folder. Email Read/Unread count does not match."","STATUS"

End If

capture_desktop ( )

KillProcess("OUTLOOK.exe")

LogWrapr "ERROR - ","The last Email : 
"& dtSubject &" sent was Not Found in Inbox. Inbox Read/Unread count will not match."","STATUS"

LogWrapr "ERROR - ","Emails sent were NOT SYNCHRONIZED in Inbox. Email Read/Unread count does not match."","FAILURe"

ExitTest

End If

End Sub
Function SearchPFemail ( dtSubject)

    Set objOutlook = CreateObject("Outlook.Application")

    Set myNameSpace = objOutlook.GetNameSpace("MAPI")

    Set objPFALL = myNameSpace.GetDefaultFolder(18)

        flag = 0

        Datatable.GetSheet("Global").SetCurrentRow 1

        PF_FoldName = DataTable("PF_FolderName", dtGlobalSheet)

        'PF_FoldName = "testPF01"

        For each objPF in objPFALL.Folders

            If objPF.Name = PF_FoldName Then

                'msgbox "Found test folder"

                flag = 1

                Exit for

            End If

        Next

        found = 0

        If flag = 0 Then

            LogWRaper "ERROR - ","Public Folder does not exist. Create PF on Outlook client or check the dataset ","STATUS"

        End If

    End If

End Function
Set colItems = objPF.Items
loopcount =0

While found = 0 and loopcount <15
    Set colFilteredItems =colItems.Restrict("[Subject] =" & dtSubject &"")

    If colFilteredItems.Count > 0 Then
        Counter = 0
        For Each myItem In colFilteredItems
            Set objOutlookMsg  =
                CreateObject("Redemption.SafeMailItem")
                objOutlookMsg.item = myItem

                If objOutlookMsg.Subject = "" Then
                    Exit For
                End If

                If dtSubject =  objOutlookMsg.Subject
                    Then
                        'Msgbox "mail found" & objOutlookMsg.Subject
                        LogWrapper "INFO -
","FOUND email in Public Folder : " & dtSubject ,"STATUS"
                        found =1
                        SearchPFemail = 1
                        Exit For
                    End If
            End For
        End If
    End While
End If

Next

End If

Window("Microsoft Outlook").Type micF9

Wait 20

loopcount = loopcount +1

Wend

If found =1 Then

LogWraper "INFO - ","FOUND email in Public Folder. Continue to take PF unread/read counts","STATUS"

Else

LogWraper "ERROR - ","The last Email :" & dtSubject & " sent was Not Found in PF. PF Read/Unread count will not match.","STATUS"

SearchPFemail = 0

End If

Set objOutlook = Nothing

Set myNameSpace = Nothing

Set objPFALL = Nothing

Set colItems = Nothing

Set colFilteredItems = Nothing

End Function
Sub CreateInbxSubFold()

    Dim objName, objFolder, flag, j, ArrRuleNm, ErrorNum, row, Rowcount, Application

    Dim tbltext, fold, outlookfold, CheckFold, counter

    ErrorNum = 0

    Set Application = CreateObject("Outlook.Application")
    Set objName = Application.GetNamespace("MAPI")
    Set fold = objName.GetDefaultFolder(6)

    Rowcount = Datatable.GetSheet("Global").GetRowCount

End Sub
counter = 0
ReDim ArrRuleNm (Rowcout-1)
For row = 1 to Rowcout

Datatable.GetSheet("Global").SetCurrentRow row

  tbltext = TRIM(DataTable("Inbox_FolderName", dtGlobalSheet))
  If tbltext = "" Then
    Exit For
  End If
  ReDim preserve
  ArrRuleNm(counter)
  ArrRuleNm(counter) = tbltext
  counter = counter+1
Next
For j = Lbound(ArrRuleNm) to Ubound(ArrRuleNm)
  'msgbox ArrRuleNm(j) & " " & j
  flag = 0
  For each outlookfold in fold.Folders
    If outlookfold = ArrRuleNm(j) Then
      LogWrapper
      "INFO - "," Found Inbox sub folder : " & outlookfold.Name,"STATUS"
      Exit For
    End If
  Next
  For each outlookfold in fold.Folders
    If outlookfold = ArrRuleNm(j) Then
      LogWrapper
      "INFO - "," Found Inbox sub folder : " & outlookfold.Name,"STATUS"
      Exit For
    End If
  Next
Next
  For each outlookfold in fold.Folders
    If outlookfold = ArrRuleNm(j) Then
      LogWrapper
      "INFO - "," Found Inbox sub folder : " & outlookfold.Name,"STATUS"
      Exit For
    End If
  Next
Next
  For each outlookfold in fold.Folders
    If outlookfold = ArrRuleNm(j) Then
      LogWrapper
      "INFO - "," Found Inbox sub folder : " & outlookfold.Name,"STATUS"
flag = 1
Exit for
End If
Next

'Add Inbox subfolder if not found.
If flag = 0 Then

Set objFolder =
fold.Folders.Add(ArrRuleNm(j))
LogWrapper "INFO - "
Added Inbox sub folder : " & objFolder,"STATUS"
End If

Next

Set Application = Nothing
Set objName = Nothing
Set fold = Nothing
Set objFolder = Nothing
Set objFolder = Nothing
Set CheckFold = Nothing
'End Sub

'******************************************************************************
'******************************************************************************
' Function Name         :      DumbpEmail ( )
'Description          :       Dump Inbox email subject data into
Emaildump.txt

'******************************************************************************
'******************************************************************************

Sub DumbpEmail (filename )

    Dim Application, objNamespace, fso, MyFile, objInboxPF, Totalemail, email, myline

    Set Application = CreateObject("Outlook.Application")
    Set objNamespace = Application.GetNamespace("MAPI")
    Set fso = CreateObject("Scripting.FileSystemObject")
    Set MyFile = fso.CreateTextFile(Pathhash.item("TO_path") & "TO_O2K7_SendEmails_20016\Email_Cnt" & filename & ".txt", True)
    Set objInboxPF = objNamespace.GetDefaultFolder(6)
    Totalemail = objInboxPF.Items.Count

    'Get total number of emails in Inbox.
If Totalemail >= 0 Then

    LogWrapper "INFO - ", " Total number of emails in Inbox are: " & Totalemail, "STATUS"

End if

For each email in objInboxPF.Items

    myline = email.subject & "->"

    If email.unread = TRUE Then

        myline = myline & "unread"

    Else

        myline = myline & "read"

    End If

    MyFile.WriteLine(myline)

Next

MyFile.Close

LogWrapper "INFO - ", " Inbox email subjects are dumped into Emaildump.txt file. ", "STATUS"

Set Application = Nothing

Set objNamespace = Nothing

Set fso = Nothing

Set MyFile = Nothing

Set objInboxPF = Nothing
End Sub

Sub ReadDumpEmail(filename)

Dim EmailArr, filesys, filetxt, ArryCounter, TxtLine
Dim Application, objNamespace, objInboxPF, Totalemail, email, state, found, j, EmailVal
ReDim EmailArr(0)
Set filesys= CreateObject("Scripting.FileSystemObject")
If filesys.FileExists(Pathhash.item("TO_path") & "TO_O2K7_SendEmails_20001\Email_Cnt" & filename & ".txt") Then
Set filetxt =
filesys.OpenTextFile(Pathhash.item("TO_path") & "TO_O2K7_SendEmails_20001\" & filename & ".txt", 1)
ArryCounter = 0
Do While filetxt.AtEndOfStream <> True
    TxtLine = Trim(filetxt.ReadLine)
    ReDim preserve EmailArr(ArryCounter)
    EmailArr(ArryCounter) = TxtLine
    ArryCounter = ArryCounter + 1
Loop
Else
    LogWraper "ERROR - ", "Emaildump.txt file does not exits in path :
" & Pathhash.item("TO_path") & "TO_O2K7_SendEmails_20001" , "STATUS"
    ErrorNum = ErrorNum+1
End if

Set filesys = Nothing

Set Application = CreateObject("Outlook.Application")
Set objNamespace = Application.GetNamespace("MAPI")
Set objInboxPF = objNamespace.GetDefaultFolder(6)
    Totalemail = objInboxPF.Items.Count
    If Totalemail = (Ubound(EmailArr)+1) Then
LogWrapper "INFO - ", " Email count from Inbox matches with
Emaildump.txt. Email count is : " & Totalemail, "STATUS"

Else

LogWrapper "ERROR - ", " Inbox Email count
DOES NOT MATCH with Emaildump.txt file. Email count in Inbox is : " & Totalemail
& ". Email count in file is : " & Ubound(EmailArr)+1, "STATUS"

ErrorNum = ErrorNum+1

End If

For each email in objInboxPF.Items

state = ""

If email.Unread = TRUE Then

state = "unread"

Else

state = "read"

End If

found = 0

For j = Lbound(EmailArr) to Ubound(EmailArr)

If EmailArr(j) <> "" Then

EmailVal = Split

(EmailArr(j), "->")

' msgbox "Message : "

End If
If email.subject = EmailVal(0) Then
    found = 1
Else
    EmailVal(1) <> state Then
        LogWrapped "ERROR - ", " Email Status(Read/Unread) DOES NOT MATCH for : " & email.subject & ". Previous status is : " & EmailVal(1) & ". Current status is : " & state , "STATUS"
        ErrorNum = ErrorNum + 1
        EmailArr(j) = ""
    End If
End If

Next

If found = 0 Then
LogWrapper "ERROR - ", " Found new emails in Inbox : " & email.subject, "STATUS"

ErrorNum = ErrorNum+1

End If

NEXT

For j = Lbound(EmailArr) to Ubound(EmailArr)

If EmailArr(j) <> "" Then

LogWrapper "ERROR - ", " Email missing in Inbox : " & EmailArr(j), "STATUS"

ErrorNum = ErrorNum+1

End If

NEXT

SetGetError ErrorNum ' Sub call

Set Application = Nothing

Set objNamespace = Nothing

Set objInboxPF = Nothing

End Sub
'Function Name : Outlook_Launch ( )

'Description : Maximize Microsoft Outlook window if exist. If the Outlook does not exit then fail the test and stop test run.

'####################################################################
'####################################################################
'Sub Outlook_Launchs ( )

LogWraper "INFO - ", "Executing Sub : Outlook_Launch. Sub Checks Outlook is Active and running ", "STATUS"

Window("regexpwndtitle:=Microsoft Outlook").Activate

If Window("regexpwndtitle:=Microsoft Outlook").Exist(3) Then
    Window("regexpwndtitle:=Microsoft Outlook").Activate
    Window("regexpwndtitle:=Microsoft Outlook").Maximize
    Window("regexpwndtitle:=Microsoft Outlook").Type micF9

Else
    Wait 5
    capture_desktop( )
    KillProcess("OUTLOOK.exe")
LogWrapper "ERROR - ", "Microsoft Outlook is not launched. Check RDP connection is active or computer status.", "FAILURE"

ExitTest

End if

End Sub

'#######################################################################
'#######################################################################
'Function Name          :     Launch_QA_Automation
'Description          :      Open the test, configure run options and settings, 'runs the test, and then checks the results of the test run.

'When QuickTest opens, it loads the Libraries and add-ins required for the test.

'#######################################################################
'#######################################################################

Public Pathhash

'Launch "TO_O2K7_SendEmails_20016"

Sub Launch (testname)
Dim qtApp 'As QuickTest.Application ' Declare the Application object variable
Dim qtTest 'As QuickTest.Test ' Declare a Test object variable
Dim qtResultsOpt 'As QuickTest.RunResultsOptions ' Declare a Run Results Options object variable
Dim pDefColl 'As QuickTest.ParameterDefinitions ' Declare a Parameter Definitions collection
Dim pDef ' As QuickTest.ParameterDefinition ' Declare a ParameterDefinition object
Dim rtParams 'As QuickTest.Parameters ' Declare a Parameters collection
Dim rtParam ' As QuickTest.Parameter ' Declare a Parameter object
Dim qtLibraries 'As QuickTest.TestLibraries 'Declare a tests libraries Collection variable.
Dim blnNeedChangeAddins ' Declare a flag for indicating whether the test's associated add-ins are currently loaded
Dim arrTestAddins ' Declare the variable for storing the test's associated add-ins

LoadINIFiles ( )

Set qtApp = CreateObject("QuickTest.Application") ' Create the Application object
If qtApp.Launched Then
    qtApp.Quit
End If

' Verify required addins are loaded. If not load the required adding for the test
arrTestAddins =
qtApp.GetAssociatedAddinsForTest(Pathhash.item("TO_path")&testname) 'Create an array containing the list of addins associated with this test

' Check if all required add-ins are all already loaded

blnNeedChangeAddins = False ' Assume no change is necessary
For Each testAddin In arrTestAddins ' Iterate over the test's associated add-ins list
  If qtApp.Addins(testAddin).Status <> "Active" Then ' If an associated add-in is not loaded
    blnNeedChangeAddins = True ' Indicate that a change in the loaded add-ins is necessary
  Exit For ' Exit the loop
  End If
Next

If qtApp.Launched And blnNeedChangeAddins Then
  qtApp.Quit ' If a change is necessary, exit QuickTest to modify the loaded add-ins
End If

If blnNeedChangeAddins Then
  Dim blnActivateOK
blnActivateOK = qtApp.SetActiveAddins(arrTestAddins,
errorDescription) ' Load the add-ins associated with the test and check whether they load
successfully.

If Not blnActivateOK Then ' If a problem occurs while loading the add-ins
    MsgBox errorDescription ' Show a message containing the
error
    WScript.Quit ' And end the automation program.
End If

End If  ' End of loading Addin logic

If Not qtApp.Launched Then ' If QuickTest is not yet open
    qtApp.Launch ' Start QuickTest (with the correct add-ins loaded)
End If

'#################################################################
'Set run settings for the QTP test in Tools > Options
'#################################################################
qtApp.Visible = FALSE    ' Make the QuickTest application
invisible
qtApp.Options.DisableVORecognition = False
qtApp.Options.AutoGenerateWith = False
qtApp.Options.WithGenerationLevel = 2
qtApp.Options.TimeToActivateWinAfterPoint = 500
qtApp.Options.SaveLoadAndMonitorData = False
qtApp.Options.Run.RunMode = "Fast"
qtApp.Options.Run.ViewResults = False
qtApp.Options.Run.CaptureForTestResults = "OnError"
qtApp.Options.Run.StepExecutionDelay = 0
qtApp.Options.WindowsApps.AttachedTextArea = "TopLeft"
qtApp.Options.WindowsApps.ExpandMenuToRetrieveProperties = True
qtApp.Options.WindowsApps.NonUniqueListItemRecordMode = "ByName"
qtApp.Options.WindowsApps.RecordOwnerDrawnButtonAs = "PushButtons"
qtApp.Options.WindowsApps.ClickEditBeforeSetText = 0
qtApp.Options.WindowsApps.VerifyMenuInitEvent = 0
qtApp.Folders.RemoveAll
qtApp.Folders.Add(Pathhash.item("AO_path"))
qtApp.Folders.Add(Pathhash.item("Rec_path")) ' End of Tools > Options settings in QTP

'#################################################################################################################################

'File > Test Settings

'#################################################################################################################################

qtApp.Test.Settings.Launchers("Windows Applications").Active = False
qtApp.Test.Settings.Launchers("Windows Applications").RecordOnQTDescendants = True
qtApp.Test.Settings.Launchers("Windows Applications").RecordOnExplorerDescendants = False
qtApp.Test.Settings.Launchers("Windows Applications").RecordOnSpecifiedApplications = False
qtApp.Test.Settings.Resources.DataTablePath = "<Default>"
"O2K7_RecoveryScenario", 1

End of File > Test Settings in QTP application

qtApp.Open Pathhash.item("TO_path")&testname, True ' Open the test in read-only mode

' Add Utilities.vbs if it's not in the collection
If qtLibraries.Find(Pathhash.item("Lib_path") & "logs.vbs") = -1 Then ' If the library cannot be found in the collection
    qtLibraries.Add Pathhash.item("Lib_path") & "logs.vbs", 1 ' Add the library to the collection in first position
End If

If qtLibraries.Find(Pathhash.item("Lib_path") & "Recovery.vbs") = -1 Then ' If the library cannot be found in the collection
    qtLibraries.Add Pathhash.item("Lib_path") & "Recovery.vbs", 2 ' Add the library to the collection in Second Position
End If

If qtLibraries.Find(Pathhash.item("Lib_path") & "DtImport.vbs") = -1 Then
    qtLibraries.Add Pathhash.item("Lib_path") & "DtImport.vbs", 3
End If

If qtLibraries.Find(Pathhash.item("Lib_path") & "Miscellaneous.vbs") = -1 Then
    qtLibraries.Add Pathhash.item("Lib_path") & "Miscellaneous.vbs", 4
End If

' Retrieve the parameters collection defined for the test.
Set pDefColl = qtApp.Test.ParameterDefinitions
Set rtParams = pDefColl.GetParameters() ' Retrieve the Parameters collection defined
Set qtTest = qtApp.Test
Set qtResultsOpt = CreateObject("QuickTest.RunResultsOptions") ' Create the Run Results Options object
qtResultsOpt.ResultsLocation = Pathhash.item("TR_path")&testname&_"_TestResult" ' Set the results location
qtTest.Run qtResultsOpt ' Run the test
qtTest.Close ' Close the test
qtApp.Quit

Set qtResultsOpt = Nothing ' Release the Run Results Options object
Set qtTest = Nothing ' Release the Test object
Set qtApp = Nothing ' Release the Application object
Set pDefColl = Nothing
Set rtParams = Nothing
Set qtLibraries = Nothing
Set filesys = Nothing
Set filesystxt = Nothing
Set Pathhash = Nothing
Set filetxt = Nothing
End Sub
Sub LoadINIFiles()

Set Pathhash= CreateObject("Scripting.Dictionary")
Dim fileys, filetxt, filesystxt, INIPath, INIpathAbs, TxtLine, INIVal
INIPath = "C:\Automator\Lib\Client\O2K7_Lib\"
Set fileys = CreateObject("Scripting.FileSystemObject")
Set filesystxt =
fileys.CreateTextFile("C:\Automator\Lib\Client\O2K7_Lib\Txtfile\INIVar_Paths.txt", True)
INIpathAbs = INIPath & "O2K7_paths.ini"
If fileys.FileExists(INIpathAbs) Then
    Set filetxt = fileys.OpenTextFile(INIpathAbs, 1)
    Do While filetxt.AtEndOfStream <> True
        TxtLine = Trim(filetxt.ReadLine)

- 151 -
If TxtLine <> "" AND Left(TxtLine,1) <> "#" Then

If InStr(TxtLine, "#") > 0 Then

TxtLine = Left(TxtLine, InStr(TxtLine, ">#") - 1)

End If

INIVal = Split (TxtLine, "=")

If TRIM(INIVAL(0)) <> "" AND TRIM(INIVAL(1)) <> "" Then

Pathhash.Add TRIM(INIVAL(0)), TRIM(INIVAL(1))

End If

End If

End If

LOOP

Else

Msgbox "O2K3_path.ini file does not exits in path " & INIpathAbs &" Check you files"

End If

For Each Key1 IN Pathhash.Keys

filesystxt.WriteLine(key1 & "=" & Pathhash.item(key1))
Next

filetxt.close
filesystxt.Close
Set filesys = nothing
Set filetxt = nothing
Set filesystxt = nothing

End Sub

'###################################################################################
'###################################################################################

'Function Name          :     CloseRunningProcesses.
'Description          :      Kills OUTLOOK,EXE processes.
'###################################################################################
'###################################################################################

'Function used by Recovery Scenario to kill the below exe
Public Function CloseRunningProcesses(Object, Method, Arguments, retVal)
    'Append Error to log file
    HandleRunError ()
    ' ErObjPtr (Err)
capture_desktop() ' Caputre Desktop Image

If NOT SystemUtil.CloseProcessByName("WINWORD.exe") Then

    SystemUtil.CloseProcessByName("WINWORD.exe")

End If

If Window("Microsoft Outlook").Exist Then

    hwnd = Window("Microsoft Outlook").GetROProperty("hwnd")

End If

If NOT SystemUtil.CloseProcessByHwnd(hwnd) Then

    SystemUtil.CloseProcessByHwnd(hwnd)

End If

SystemUtil.CloseProcessByName("OUTLOOK.exe")

KillWord()

LogWraper "ERROR - ", "UNKNOWN ERROR : Test Run Stopped by Recovery Scenario", "FAILURE"

End Function
'Function Name : KillWord ( )

'Description : Kills WINWORD.EXE processes.

Sub KillWord ( )

'If Microsoft Word Application crashes, then kill it

    If Dialog("text:=Microsoft Office Word","nativeclass:=#32770","is owned window:=True","is child window:=False").Exist(3) Then

        Dialog("text:=Microsoft Office Word","nativeclass:=#32770","is owned window:=True","is child window:=False").Activate

        Dialog("text:=Microsoft Office Word","nativeclass:=#32770","is owned window:=True","is child window:=False").WinButton("text:=&Don't Send","nativeclass:=Button").Click

        LogWrapper "ERROR - ", "Word crashed. QTP logic handled crash and clicked on button Don't Send", "STATUS"

        Wait 3

        SystemUtil.CloseProcessByName("WINWORD.exe")

    End If
Sub HandleOutlook()
  If Dialog("text:=Microsoft Office Outlook",
  "nativeclass:=#32770").Exist(2) Then
    Dialog("text:=Microsoft Office Outlook",
  "nativeclass:=#32770").Activate
    Dialog("text:=Microsoft Office Outlook",
  "nativeclass:=#32770").WinButton("text:=&Don't Send",
  "nativeclass:=Button").Click
    LogWraper "ERROR - ", "Outlook crashed. QTP logic handled crash and clicked on button Don't Send", "STATUS"
  End If
End Sub

'#######################################################################
'Function Name :     HandleRunError
'Description : Captures the latest Run Error and logs it into logs.txt.
Dim ErrorNum, ErrorDes

'msgbox ErrorObject.Number

'LogWrapr "WARNING - ", "Test Run Error on Object : " & objtxt, "STATUS"

If Err.Number <> 0 Then
    ErrorNum = CStr(Err.Number)
    ErrorDes = Err.Description
    LogWrapr "ERROR - ", "Test Run Error # : " & ErrorNum & " Description : " & ErrorDes, "," & Status
    Err.Clear ' Clear the error.
End if
End Sub

'Function ErObjPtr (Object)
'    ErObjPtry = GetRef(Object)
'    HandleRunError (ErObjPtry )
'End Function

'#########################################################################
'#########################################################################
'Function Name          :     capture_desktop
'Description          :      Captures the desktop screenshot when Run Error occurs

Error occurs
'Capture Bitmap Image

Public Sub capture_desktop()
    Dim datestamp
    Dim filename
    datestamp = Now()
    filename = Environment("TestName")&"_"&datestamp&".png"
    filename = Replace(filename,"/",""")
    filename = Replace(filename,":\",""")
    filename = Pathhash.item("TR_path")&filename
    Desktop.CaptureBitmap filename, true
    Reporter.ReportEvent micFail,"ScreenShot of Error","<img src=" & filename & ">
    Add screenshot to Test Result file
    LogWrapper "INFO - ", "Screenshot of error is captured ", "STATUS"
End Sub

'Function Name : ExportTimeStp ()
'Description : If a Test Operation fails, then the data in timestamp filed (which is output to Global sheet) will be exported to external excel Subject_TimeStamp.xls file

Sub ExportTimeStp (filename )

    If filename = "SendMail" Then

        Datatable.ExportSheet

        Pathhash.item("AO_path")&"AO_O2K3_SendMail\Subject_TimeStamp.xls", 1 ' Export TimeStamp to external file

        LogWraper "INFO - ","Script is incomplete. Subject time stamp of all mails are exported to external excel file Subject_TimeStamp.xls" 

    Elseif filename = "SendCalendar" Then

        Datatable.ExportSheet

        Pathhash.item("AO_path")&"AO_O2K3_SendCalendar\Subject_TimeStamp.xls", 1 ' Export TimeStamp to external file

        LogWraper "INFO - ","Script is incomplete. Subject time stamp of all mails are exported to external excel file Subject_TimeStamp.xls" 

    Elseif filename = "" Then

        ' dont do anything.
Else

LogWrapper "WARNING - ", "ExportTimeStp function call argument is not valid. ", "STATUS"

End If

End Sub

'###########################################################################
'###########################################################################
'Function Name : StopRun ( )
'Description : If there is force of test failure by scipt, SopRun sub will be executed.

'###########################################################################
'###########################################################################

Sub StopRun (filename )

   ExportTimeStp (filename )

   Recovery.SetScenarioStatus 1, FALSE 'Disable recovery scearion

   CloseRunningProcesses Object, Method, Arguments, retVal 'Recovery Scenario function call

   ExitTest
18 Appendix B – Sample EAF Automation Code

'#########################################################################
'#########################################################################

'File Name           :     TO_O2K3_CheckCalendar_20007

'Description          :     Check email in Run multiple iteration of
"Login to Profile - Send email to the list of users email address in local datasheet -
Logout"

'Input Parameter    :     'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )

'Output Parameter     :     None

'Last Modified Date    :

'#########################################################################
'#########################################################################

Dim logrowcou, logitercou, TotalErrorNum

LogWraper "INFO - ", "Started execution of script TO_O2K3_CheckCalendar_20007",
"STATUS"

Recovery.Activate
PrepRun "TO_O2K3_CheckCalendar_20007" 'Clean's up running process and check all required AO scripts files exist

' LoadINIFiles ( )

'Import datatable Sheet

XLFFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
XLFFileValidation "AO_O2K3_CheckCalendar", Pathhash.item("DT_path"), "Verify"

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K3_CheckCalendar", "AO", "2",
"AO_O2K3_CheckCalendar", "AO_O2K3_CheckCalendar"

logrowcou = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]" ).GetRowCount

For logitercou = 1 to logrowcou

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou

    RunAction "AO_O2K3_CheckCalendar [AO_O2K3_CheckCalendar]",
oneIteration

    OutlookLogout( )

Next
If GlobalErrorNum <> 0 Then

    LogWrapper "ERROR - ", "Grand total # of erros in AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum , "FAILURE"

    Reporter.ReportEvent micFail,

    "AO_O2K3_CheckMail","Grand total # of erros in AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum

Else

    Reporter.ReportEvent micPass,

    "TO_O2K3_CheckCalendar_20007", "Successfully completed running script TO_O2K3_CheckCalendar_20007"

    LogWrapper "INFO - ", "Successfully completed running script TO_O2K3_CheckCalendar_20007", "SUCCEED"

End If

########################################################################
########################################################################

'File Name           :     TO_O2K3_CheckMails_20006
'Description          :     Check email in Run multiple iteration of
                          "Login to Profile - Send email to the list of users email address in local datasheet -
                          Logout"
'Input Parameter    :  'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )

'Output Parameter     :     None

'#######################################################################
'#######################################################################

Dim logrowcou, logitercou, TotalErrorNum

Recovery.Activate

  PrepRun "TO_O2K3_CheckMails_20006" 'Clean's up running process and check all
required AO scripts files exist

  XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"),  "Verify"
  XLFileValidation "AO_O2K3_CheckMail", Pathhash.item("DT_path"),  "Verify"

'Import datatable Sheet
  ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
  "AO_Check-Mail-Cal"
  ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2", "AO_O2K3_CheckMail",
  "AO_O2K3_CheckMail"
logrowcou = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]").GetRowCount

On Error Resume Next

For logitercou = 1 to logrowcou

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou

    ErrorCheck "AO_O2K3_Login", Err.Description, Err.Number

    RunAction "AO_O2K3_CheckMail [AO_O2K3_CheckMail]", oneIteration, logitercou

    ErrorCheck "AO_O2K3_CheckMail", Err.Description, Err.Number

OutlookLogout( )

Next

If GlobalErrorNum <> 0 Then

    LogWraper "ERROR - "," Grand toltal # of erros in AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum ,"FAILURE"

- 165 -
Reporter.ReportEvent micFail,
"AO_O2K3_CheckMail","Grand total # of errors in AO_O2K3_CheckMail script for all
the user are : " & GlobalErrorNum

Else
Reporter.ReportEvent micPass,
"TO_O2K3_CheckMails_20006", "Successfully completed running script
TO_O2K3_CheckMails_20006"
LogWrapper "INFO - ", "Successfully completed running
script TO_O2K3_CheckMails_20006", "SUCCEEDED"
End If

########################################################################
####################################################################
'File Name           :     TO_O2K3_CheckPost_toPF_20015
'Description          :     Check email in Run multiple iteration of
"Login to Profile - Send email to the list of users email address in local datasheet -
Logout"
'Input Parameter    :     'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )
'Output Parameter     :     None

- 166 -
Dim logrowcou, logitercou, TotalErrorNum

Recovery.Activate

PrepRun "TO_O2K3_CheckPost_toPF_20015" 'Clean's up running process and check all required AO scripts files exist

    XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
    XLFileValidation "AO_O2K3_SendCheckPost_toPF", Pathhash.item("DT_path"), "Verify"

    'Import datatable Sheet
    ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login", "AO_Check-Mail-Cal"
    ImpDiffXLTOAO "AO_O2K3_SendCheckPost_toPF", "AO", "2", "AO_O2K3_SendCheckPost_toPF"

    logrowcou = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]").GetRowCount

    For logitercou = 1 to logrowcou
RunAction "AO_O2K3_Login [AO_O2K3_Login]",
oneIteration, logitercou

RunAction "AO_O2K3_SendCheckPost_toPF [AO_O2K3_SendCheckPost_toPF]", oneIteration, "CHECKPOST"

OutlookLogout( )

Next

If GlobalErrorNum <> 0 Then

    LogWrapper "ERROR - ", "Grand total # of errors in
    TO_O2K3_CheckPost_toPF_20015 script for all the user are : " & GlobalErrorNum
    ,"FAILURE"

    Reporter_ReportEvent micFail,

    "TO_O2K3_CheckPost_toPF_20015", "Grand total # of errors in
    TO_O2K3_CheckPost_toPF_20015 script for all the user are : " & GlobalErrorNum

Else

    Reporter_ReportEvent micPass,

    "TO_O2K3_CheckPost_toPF_20015", "Successfully completed running script
    TO_O2K3_CheckPost_toPF_20015"

    LogWrapper "INFO - ", "Successfully completed running
    script TO_O2K3_CheckPost_toPF_20015", "SUCCEED"

End If
Dim logrowcou, logitercou, RowCt
LogWrapper "INFO - ", "Started execution of script TO_O2K3_CreateRule_20010",
"STATUS"
Recovery.Activate

PrepRun "TO_O2K3_CreateRule_20010" 'Clean's up running process and check all
required AO scripts files exist

XLFFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"
XLFFileValidation "AO_O2K3_Rules", Pathhash.item("AO_path"), "Verify"
'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login", "AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K3_Rules", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]" ).GetRowCount

ScriptName = "AO_O2K3_CreateRules"

For logitercou = 1 to rowct

   RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
   Datatable.SetCurrentRow logitercou
   ProfName = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]" ).GetParameter("ProfileName")
   RunAction "AO_O2K3_Rules [AO_O2K3_Rules]", oneIteration, ScriptName, ProfName

   RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", oneIteration

   Next

LogWraper "INFO - ", "Successfully completed running script TO_O2K3_CreateRule_20010", "SUCCEED"

'##########################################################################
'##########################################################################
'File Name : TO_O2K3_DeleteRule_20012.
'Description :
'Input Parameter : logitercou (contains the row number to be set on AO_O2K3_Login datasheet )
'Output Parameter : None
'##########################################################################
'##########################################################################

Dim logrowcou, logitercou, RowCt
LogWraper "INFO - ", "Started execution of script TO_O2K3_DeleteRule_20012", "STATUS"
Recovery.Activate

PrepRun "TO_O2K3_DeleteRule_20012" 'Clean's up running process and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"
XLFileValidation "AO_O2K3_Rules", Pathhash.item("AO_path"), "Verify"
'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K3_Rules", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

ScriptName = "AO_O2K3_DeleteRules"

For logitercou = 1 to rowct

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou

    Datatable.SetCurrentRow logitercou

    ProfName = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetParameter("ProfileName")

    RunAction "AO_O2K3_Rules [AO_O2K3_Rules]", oneIteration, ScriptName, ProfName

    RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", oneIteration

Next

If GlobalErrorNum <> 0 Then

    LogWrapper "ERROR - ", " Grand total # of errors in
TO_O2K3_DeleteRule_20012 script for all the users are : " & GlobalErrorNum

,"FAILURE"
Reporter.ReportEvent micFail, "AO_O2K3_CheckMail","Grand total # of errors in AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum
Else
LogWrapper "INFO - ", "Successfully completed running script TO_O2K3_DeleteRule_20012", "SUCCEEDED"
End If

'#######################################################################
'FILE NAME           :     TO_O2K3_MsgOutofOfficePopUp_20004
'DESCRIPTION          :     Script verifies that Out of Office message appears or not when a user is logged into Outlook. This script should be run after TO_O2K3_OutofOfficeON_20002
'INPUT PARAMETER    :      None
'OUTPUT PARAMETER     :     MsgVal (Verify the message box display)
'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )

'#######################################################################
Dim MsgVal, logrowcou, logitercou

LogWraper "INFO - ", "Started Execution of
TO_O2K3_MsgOutOfOfficePopUp_20004", "STATUS"

Recovery.Activate

PrepRun "TO_O2K3_MsgOutOfOfficePopUp_20004" 'Clean's up running process
and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"

'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_OutofOffice_ON-OFF"

logrowcou = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

MsgVal = "VERIFY"

For logitercou = 1 to logrowcou

RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration,

logitercou

RunAction "AO_O2K3_MsgOutOfOfficePopUp
[AO_O2K3_MsgOutOfOfficePopUp]", oneIteration, MsgVal
RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", onelIteration

Next

"Successfully completed running script TO_O2K3_MsgOutofOfficePopUp_20004"
LogWraper "INFO - ", "Successfully completed running script
O_O2K3_MsgOutofOfficePopUp_20004", "SUCCEED"

'*******************************************************************************

TO_O2K3_O2K7_MailboxCleanUp_20030.

Description : Delete emails in Mailbox from Outlook client.

Input Parameter : 'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )

Output Parameter : None

'*******************************************************************************

Dim logrowcou, logitercou, rowct, rownum, dtSubjectval
LogWraper "INFO - ", "Started execution of script
TO_O2K3_O2K7_MailboxCleanUp_20030", "STATUS"
Recovery.Activate

PrepRun "TO_O2K3_O2K7_MailboxCleanUp_20030" 'Clean's up running process and check all required AO scripts files exist

XLFValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_O2K3_O2K7_MailboxCleanUp"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

On Error Resume Next

For logitercou = 1 to rowct

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration , logitercou

    ErrorCheck "AO_O2K3_Login", Err.Description, Err.Number

RunAction "AO_O2K3_O2K7_MailboxCleanUp
[AO_O2K3_O2K7_MailboxCleanUp]", oneIteration

RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", oneIteration

- 176 -
ErrorCheck "AO_O2K3_Logout", Err.Description, Err.Number

Next


LogWraper "INFO - ", "Successfully completed running script TO_O2K3_O2K7_MailboxCleanUp_20030", "SUCCEED"

'#########################################################################
'#########################################################################
'
'File Name           :     TO_O2K3_OutofOfficeOFF_20003. 
'Description          :     Script sets Out of Office option ON
'Input Parameter    :      None
'Output Parameter     :     MsgVal (Value to be selected on Out of office message box)

'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )

'#########################################################################
'#########################################################################

Dim MsgVal, logrowcou, logitercou

LogWraper "INFO - ", "Started Execution of TO_O2K3_OutofOfficeOFF_20003", "STATUS"
Recovery.Activate

PrepRun "TO_O2K3_OutofOfficeOFF_20003" 'Clean's up running process and check all required AO scripts files exist

'Check all the required AO's exist before running the script.
AONames = Array ("AO_O2K3_Login", "TO_O2K3_OutofOfficeOFF_20003", "AO_O2K3_Logout")
For i= Lbound(AONames) to Ubound(AONames)
  AOFileExists (AONames(i) )
Next

XLFFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login", "AO_OutofOffice_ON-OFF"

logrowcou = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]").GetRowCount

MsgVal = "NO"

For logitercou = 1 to logrowcou
RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration,
logitercoup

RunAction "AO_O2K3_MsgOutofOfficePopUp
[AO_O2K3_MsgOutofOfficePopUp]", oneIteration, MsgVal

RunAction "AO_O2K3_OutofOfficeOFF
[AO_O2K3_OutofOfficeOFF]", oneIteration

RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", oneIteration

Next

"Successfully completed running script TO_O2K3_OutofOfficeOFF_20003"

LogWraper "INFO - ", "Successfully completed running script
TO_O2K3_OutofOfficeOFF_20003", "SUCCEED"

'#######################################################################
###############################
# File Name           :     TO_O2K3_OutofOfficeON_20002.
# Description          :     Script sets Out of Office option ON
# Output Parameter    :       MsgVal (Value to be selected on Out of office message box)
logitercou (contains the row number to be set on AO_O2K3_Login datasheet )

'#######################################################################
'#######################################################################
'#######################################################################

Dim MsgVal, logitercou, logrowcou

LogWraper "INFO - ", "Started Execution of TO_O2K3_OutofOfficeON_20002", "STATUS"

Recovery.Activate

PrepRun "TO_O2K3_OutofOfficeON_20002" 'Clean's up running process and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"

'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login", "AO_OutofOffice_ON-OFF"

logrowcou = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]").GetRowCount

MsgVal = "YES"
For logitercou = 1 to logrowcou

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration,
    logitercou

    RunAction "AO_O2K3_MsgOutOfOfficePopUp
    [AO_O2K3_MsgOutOfOfficePopUp]", oneIteration, MsgVal

    RunAction "AO_O2K3_OutofOfficeON
    [AO_O2K3_OutofOfficeON]", oneIteration

    RunAction "AO_O2K3_Logout [AO_O2K3_Logout]",
    oneIteration

Next

"Successfully completed running script TO_O2K3_OutofOfficeON_20002"

LogWrapper "INFO - ", "Successfully completed running script
TO_O2K3_OutofOfficeON_20002", "SUCCEED"

'#######################################################################
'######################################################################_
'File Name           :     TO_O2K3_Rules_CheckMails_20009
'Description : Check email in Run multiple iteration of "Login to Profile - Send email to the list of users email address in local datasheet - Logout"

'Input Parameter : 'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )

'Output Parameter : None

#######################################################################
#######################################################################

Dim logrowcou, logitercou, TotalErrorNum

Recovery.Activate

PrepRun "TO_O2K3_Rules_CheckMails_20009" 'Clean's up running process and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
XLFileValidation "AO_O2K3_CheckMail", Pathhash.item("DT_path"), "Verify"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2", "AO_O2K3_CheckMail",
"AO_O2K3_CheckMail"
ImpDiffXLTOAO "AO_O2K3_Rules", "AO", "1",
"TO_O2K3_Rules_CheckMails_20009", "Global"

logrowcou = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

For logitercou = 1 to logrowcou

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration,
logitercou

    'Verify Rule 1

    Outlook_FoldName = "From_Someone"

    ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2",
"AO_O2K3_CheckMail", "From_Someone"

    RunAction "AO_O2K3_CheckMail [AO_O2K3_CheckMail]", oneIteration,
logitercou, Outlook_FoldName

    'Verify Rule 2

    Outlook_FoldName = "Flag_Emails"

    ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2",
"AO_O2K3_CheckMail", "Flag_Emails"

    RunAction "AO_O2K3_CheckMail [AO_O2K3_CheckMail]", oneIteration,
logitercou, Outlook_FoldName

    'Verify Rule 3

    Outlook_FoldName = "Body_Emails"
ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2",
"AO_O2K3_CheckMail", "Body_E-mails"

RunAction "AO_O2K3_CheckMail [AO_O2K3_CheckMail]", oneIteration,
logitercou, Outlook_FoldName

'Verify Rule 4

Outlook_FoldName = "HighImp_E-mails"

ImpDiffXLTOAO "AO_O2K3_CheckMail", "AO", "2",
"AO_O2K3_CheckMail", "HighImp_E-mails"

RunAction "AO_O2K3_CheckMail [AO_O2K3_CheckMail]",
oneIteration, logitercou, Outlook_FoldName

OutlookLogout( )

Next

If GlobalErrorNum <> 0 Then

LogWraper "ERROR - ", "Grand toltal # of errors in
AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum , "FAILURE"

Reporter.ReportEvent micFail,

"AO_O2K3_CheckMail", "Grand toltal # of errors in AO_O2K3_CheckMail script for all
the user are : " & GlobalErrorNum

Else
Reporter.ReportEvent micPass,

"TO_O2K3_Rules_CheckMails_20009", "Successfully completed running script

TO_O2K3_Rules_CheckMails_20009"

LogWraper "INFO - ", "Successfully completed running

script TO_O2K3_Rules_CheckMails_20009", "SUCCEEDED"

End If

'#######################################################################
'########################################################################
'File Name           :     TO_O2K3_Rules_SendMails_20008.
'Description          :     Run multiple iteration of "Login to Profile -
Send email to the list of users email address in local datasheet - Logout"
'Input Parameter    :     'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )
'Output Parameter     :     None
'########################################################################
'########################################################################

Dim logrowcou, logitercou, RowCt
LogWraper "INFO - ", "Started execution of script

TO_O2K3_Rules_SendMails_20008", "STATUS"
Recovery.Activate

PrepRun "TO_O2K3_Rules_SendMails_20008" 'Clean's up running process and check all required AO scripts files exist

  XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
  XLFileValidation "AO_O2K3_SendMail", Pathhash.item("DT_path"), "Verify"
  'Import datatable Sheet for Rule 1
  ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
  "From_Someone"
  ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "2", "AO_O2K3_SendMail",
  "From_Someone"
  ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "1", "Global", "Global"

  rowct = Datatable.GetSheet("AO_O2K3_Login
  [AO_O2K3_Login]").GetRowCount
  FolderName = "From_Someone"
  For logitercou = 1 to rowct
    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
    RunAction "AO_O2K3_SendMail [AO_O2K3_SendMail]",
    oneIteration, FolderName
  OutlookLogout( )
  Next
'Import datatable Sheet for Rule 2

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"Flag_Emails"

ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "2", "AO_O2K3_SendMail",
"Flag_Emails"

ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

FolderPath = "Flag_Emails"

For logitercou = 1 to rowct
    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
        RunAction "AO_O2K3_SendMail [AO_O2K3_SendMail]",
        oneIteration, FolderName
        OutlookLogout( )
    Next

'Import datatable Sheet for Rule 3

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"Body_HighImp_Emails"

ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "2", "AO_O2K3_SendMail",
"Body_Emails"
ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "1", "Global", "Global"

    rowct = Datatable.GetSheet("AO_O2K3_Login [AO_O2K3_Login]").GetRowCount

For logitercou = 1 to rowct
    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
        FolderName = "Body_Emails"
        RunAction "AO_O2K3_SendMail [AO_O2K3_SendMail]",
            oneIteration, FolderName

    'Import datatable Sheet for Rule 4
    ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "2", "AO_O2K3_SendMail",
        "HighImp_Emails"
        FolderName = "HighImp_Emails"
        RunAction "AO_O2K3_SendMail [AO_O2K3_SendMail]",
            oneIteration, FolderName

    OutlookLogout()  

    Next

Reporter.ReportEvent micPass, "TO_O2K3_Rules_SendMails_20008", "Successfully completed running script TO_O2K3_Rules_SendMails_20008"
LogWrapper "INFO - ", "Successfully completed running script TO_O2K3_Rules_SendMails_20008", "SUCCEED"

'##############################################################################
'##############################################################################
'File Name : TO_O2K3_SendMeeting_20005
'Description : Script sets Out of Office option ON
'Input Parameter : None
'Output Parameter :
'##############################################################################
'##############################################################################
Dim logrowcou, logitercou, RowCt

Recovery.Activate

PrepRun "TO_O2K3_SendCalendar_20005" 'Clean's up running process and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
XLFileValidation "AO_O2K3_SendCalendar", Pathhash.item("DT_path"), "Verify"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_O2K3_Login"

ImpDiffXLTOAO "AO_O2K3_SendCalendar", "AO", "2",
"AO_O2K3_SendCalendar", "AO_O2K3_SendCalendar"

ImpDiffXLTOAO "AO_O2K3_SendCalendar", "AO", "1", "Global", "Global"

logrowcou = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

For logitercou = 1 to logrowcou
    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
    RunAction "AO_O2K3_SendCalendar [AO_O2K3_SendCalendar]", oneIteration
        OutlookLogout( )
    Next

Reporter.ReportEvent micPass, "TO_O2K3_SendCalendar_20005", "Successfully
completed running script TO_O2K3_SendCalendar_20005"

LogWrapper "INFO - ", "Successfully completed running script
TO_O2K3_SendCalendar_20005", "SUCCEED"
'#-----------------------------------------------------------------------
'######################################################################_
'File Name                     : TO_O2K3_SendMails_20001.
'Description                   : Run multiple iteration of "Login to Profile -
Send email to the list of users email address in local datasheet - Logout"
'Input Parameter              : logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )
'Output Parameter             : None
'#-----------------------------------------------------------------------
'######################################################################_
Option Explicit
Dim logrowcou, logitercou, rowct, rownum, dtSubjectval, filename
LogWraper "INFO - ", "Started execution of script TO_O2K3_SendMails_20001",
"STATUS"
Recovery.Activate

PrepRun "TO_O2K3_SendMails_20001" 'Clean's up running process and check
all required AO scripts files exist
XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
XLFileValidation "AO_O2K3_SendMail", Pathhash.item("DT_path"), "Verify"

'Import datatable Sheet

ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_O2K3_Login"
ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "2", "AO_O2K3_SendMail",
"AO_O2K3_SendMail"
ImpDiffXLTOAO "AO_O2K3_SendMail", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

On Error Resume Next

For logitercou = 1 to rowct
     RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logitercou
     ErrorCheck "AO_O2K3_Login", Err.Description, Err.Number

     RunAction "AO_O2K3_SendMail [AO_O2K3_SendMail]",
oneIteration
     ErrorCheck "AO_O2K3_SendMail", Err.Description, Err.Number

     OutlookLogout( )

Next

'Load login data to check for Read/Unread email counts.
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_Check-Mail-Cal"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]`).GetRowCount

Datatable.ImportSheet

Pathhash.item("AO_path")&"AO_O2K3_SendMail\TimeStamp\Inbox.xls", "Global",
"Global"

For logrowcou = 1 to rowct

   RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration, logrowcou
   Window("Microsoft Outlook").Type micF9
   rownum = Datatable.GetSheet("Global").GetRowCount
   Datatable.GetSheet("Global").setCurrentRow rownum
   Do While rownum <= 1

      If DataTable("Subject_TimeStamp", dtGlobalSheet) = "" Then

         rownum = rownum - 1

      Else

         Exit For

      End If

      If rownum <= 1 AND
      DataTable("Subject_TimeStamp", dtGlobalSheet) = "" Then

         -- Additional code

   End Do

- 193 -
LogWrapper "ERROR - ", "Datatable Subject timestamp is blank. Import subject timestamp to datatable and continue.", "FAILURE"

ExitTest

End If

Loop

Datatable.GetSheet("Global").SetCurrentRow rownum

dtSubjectval = DataTable("Subject_TimeStamp", dtGlobalSheet)

SearchEmail (dtSubjectval ) 'Sub call

Get_RUEmailCount "INBOXPF" ' Sub call

DumbpEmail "Emaildump"&logrowcou

filename = "EmailCnt"&logrowcou

PrintEmailCnt_ToFile unreadIN, readIN, "INBOXPF", unreadPF, readPF, filename, Totalemail

OutlookLogout( )

Next

Reporter.ReportEvent micPass, "TO_O2K3_SendMails_20001", "Successfully completed running script TO_O2K3_SendMails_20001"

LogWrapper "INFO - ", "Successfully completed running script TO_O2K3_SendMails_20001", "SUCCEED"
'File Name :   TO_O2K3_SendPost_toPF_20014.
'Description :  Run multiple iteration of "Login to Profile - Send email to the list of users email address in local datasheet - Logout"

'Input Parameter :  'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )
'Output Parameter :  None

Dim logrowcou, logitercou, rowct, rownum, dtSubjectval
Recovery.Activate

PrepRun "TO_O2K3_SendPost_toPF_20014" 'Clean's up running process and check all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("DT_path"), "Verify"
XLFileValidation "AO_O2K3_SendCheckPost_toPF",
Pathhash.item("DT_path"), "Verify"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_O2K3_Login"

ImpDiffXLTOAO "AO_O2K3_SendCheckPost_toPF", "AO", "2",
"AO_O2K3_SendCheckPost_toPF", "AO_O2K3_SendCheckPost_toPF"

ImpDiffXLTOAO "AO_O2K3_SendCheckPost_toPF", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

For logitercou = 1 to rowct

    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration

    RunAction "AO_O2K3_SendCheckPost_toPF
[AO_O2K3_SendCheckPost_toPF]", oneIteration, "SENDPOST"

    OutlookLogout( )

Next

Reporter.ReportEvent micPass, "TO_O2K3_SendPost_toPF_20014", "Successfully
completed running script TO_O2K3_SendPost_toPF_20014"

LogWraper "INFO - ", "Successfully completed running script
TO_O2K3_SendPost_toPF_20014", "SUCCEED"
'File Name           :     TO_O2K3_VerifyRule_20011.
'Description          :
'Input Parameter    :     'logitercou (contains the row number to be set on
AO_O2K3_Login datasheet )

Dim logrowcou, logitercou, RowCt
LogWraper "INFO - ", "Started execution of script TO_O2K3_VerifyRule_20011",
"STATUS"
Recovery.Activate

PrepRun "TO_O2K3_VerifyRule_20011" 'Clean's up running process and check
all required AO scripts files exist

XLFileValidation "AO_O2K3_Login", Pathhash.item("AO_path"), "Verify"
XLFileValidation "AO_O2K3_Rules", Pathhash.item("AO_path"), "Verify"
'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K3_Login", "AO", "2", "AO_O2K3_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K3_Rules", "AO", "1", "Global", "Global"

rowct = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetRowCount

ScriptName = "AO_O2K3_VerifyRules"

For logitercou = 1 to rowct
    RunAction "AO_O2K3_Login [AO_O2K3_Login]", oneIteration , logitercou
        Datatable.SetCurrentRow logitercou
        ProfName = Datatable.GetSheet("AO_O2K3_Login
[AO_O2K3_Login]").GetParameter("ProfileName")
    RunAction "AO_O2K3_Rules [AO_O2K3_Rules]", oneIteration, ScriptName, ProfName
    RunAction "AO_O2K3_Logout [AO_O2K3_Logout]", oneIteration

Next

If GlobalErrorNum <> 0 Then
    LogWraper "ERROR - "," Grand total # of erros in TO_O2K3_VerifyRule_20011 script for all the user are : " & GlobalErrorNum
    ","FAILURE"
Reporter.ReportEvent micFail,
"AO_O2K3_CheckMail","Grand total # of errors in AO_O2K3_CheckMail script for all the user are : " & GlobalErrorNum

Else

Reporter.ReportEvent micPass,
"TO_O2K3_VerifyRule_20011", "Successfully completed running script TO_O2K3_VerifyRule_20011"

LogWrapper "INFO - ", "Successfully completed running script TO_O2K3_VerifyRule_20011", "SUCCEED"

End If

File Name           :     TO_O2K7_CheckCalendar_20019
Description          :     Check email in Run multiple iteration of "Login to Profile - Send email to the list of users email address in local datasheet - "Input Parameter    :     'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )
Dim logrowcou, logitercou, TotalErrorNum

LogWrapper "INFO - ", "Started execution of script TO_O2K7_CheckCalendar_20019", "STATUS"
Recovery.Activate

    PrepRun "TO_O2K7_CheckCalendar_20019" 'Clean's up running process and check all required AO scripts files exist
    ' LoadINIFiles ()

    CreateDelFile "AO_O2K7_Login", Pathhash.item("DT_path") , "VerifyXLSfile"
    CreateDelFile "AO_O2K7_SendCheckCalendar", Pathhash.item("DT_path") , "VerifyXLSfile"
    'Import datatable Sheet
    ImpDiffXLTOAO "AO_O2K7_Login", "AO", "2", "AO_O2K7_Login", "AO_Check-Mail-Cal"
    ImpDiffXLTOAO "AO_O2K7_SendCheckCalendar", "AO", "2", "AO_O2K7_SendCheckCalendar"

    logrowcou = Datatable.GetSheet("AO_O2K7_Login
[AO_O2K7_Login]").GetRowCount
    CalendarPara = "CheckCalendar"
    For logitercou = 1 to logrowcou
RunAction "AO_O2K7_Login [AO_O2K7_Login]", oneIteration
RunAction "AO_O2K7_SendCheckCalendar [AO_O2K7_SendCheckCalendar]",
oneIteration, "CheckCalendar"
        Logout ( )
        Next

If GlobalErrorNum <> 0 Then
    LogWraper "ERROR - ", "Grand toltal # of erros in
TO_O2K7_CheckCalendar_20019 script for all the user are : " & GlobalErrorNum
,"FAILURE"
    Reporter.ReportEvent micFail,
    "TO_O2K7_CheckCalendar_20019","Grand toltal # of erros in
TO_O2K7_CheckCalendar_20019 script for all the user are : " & GlobalErrorNum
Else
    Reporter.ReportEvent micPass,
    "TO_O2K7_CheckCalendar_20019", "Successfully completed running script
TO_O2K7_CheckCalendar_20019"
    LogWraper "INFO - ", "Successfully completed running
script TO_O2K7_CheckCalendar_20019", "SUCCEEDED"
End If
Dim logrowcou, logitercou

Recovery.Activate

PrepRun "TO_O2K7_CheckEmails_20017" 'Clean's up running process and check all required AO scripts files exist

CreateDelFile "AO_O2K7_Login", Pathhash.item("DT_path") , "VerifyXLSfile"
CreateDelFile "AO_O2K7_SendCheckEmail", Pathhash.item("DT_path") , "VerifyXLSfile"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K7_Login", "AO", "2", "AO_O2K7_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K7_SendCheckEmail", "AO", "2",
"AO_O2K7_SendCheckEmail", "AO_O2K7_SendCheckEmail"

logrowcou = Datatable.GetSheet("AO_O2K7_Login
[AO_O2K7_Login]").GetRowCount

SendCheckEmail = "CheckEmail"

'On Error Resume Next

For logitercou = 1 to logrowcou

RunAction "AO_O2K7_Login [AO_O2K7_Login]", oneIteration, logitercou

RunAction "AO_O2K7_SendCheckEmail [AO_O2K7_SendCheckEmail]", oneIteration, "Inbox", "CheckEmail"

Logout ()

Next

If GlobalErrorNum <> 0 Then

LogWraper "ERROR - "," Grand total # of erros in TO_O2K7_CheckEmails_20017 script for all the user are : " & GlobalErrorNum
,"FAILURE"
Reporter.ReportEvent micFail,
"TO_O2K7_CheckEmails_20017","Grand total # of errors in TO_O2K7_CheckEmails_20017 script for all the user are : " & GlobalErrorNum

Else

Reporter.ReportEvent micPass,
"TO_O2K7_CheckEmails_20017", "Successfully completed running script TO_O2K7_CheckEmails_20017"

LogWrapper "INFO - ", "Successfully completed running script TO_O2K7_CheckEmails_20017", "SUCCEED"

End If

'#######################################################################
'#######################################################################
'File Name           :     TO_O2K7_CheckPost_toPF_20023
'Description          :     Check email in Run multiple iteration of "Login to Profile - Send email to the list of users email address in local datasheet - Logout"
'Input Parameter : 'logitercou (contains the row number to be set on AO_O2K3_Login datasheet )

'#######################################################################
'########################################################################

Dim logrowcou, logitercou, TotalErrorNum

Recovery.Activate

PrepRun "TO_O2K7_CheckPost_toPF_20023" 'Clean's up running process and check all required AO scripts files exist

'LoadINIFiles ( )

CreateDelFile "AO_O2K7_Login", Pathhash.item("DT_path") , "VerifyXLSfile"
CreateDelFile "AO_O2K7_SendCheckPost_toPF", Pathhash.item("DT_path") , "VerifyXLSfile"

'Import datatable Sheet
ImpDiffXLTOAO "AO_O2K7_Login", "AO", "2", "AO_O2K7_Login",
"AO_Check-Mail-Cal"

ImpDiffXLTOAO "AO_O2K7_SendCheckPost_toPF", "AO", "2", "AO_O2K7_SendCheckPost_toPF"

ImpDiffXLTOAO "AO_O2K7_SendCheckPost_toPF", "AO", "1", "Global", "Global"
logrowcou = Datatable.GetSheet("AO_O2K7_Login[AO_O2K7_Login]").GetRowCount

For logitercou = 1 to logrowcou
    RunAction "AO_O2K7_Login [AO_O2K7_Login]", oneIteration, logitercou
    RunAction "AO_O2K7_SendCheckPost_toPF [AO_O2K7_SendCheckPost_toPF]", oneIteration, "CHECKPOST"
    Logout()
Next

If GlobalErrorNum <> 0 Then
    LogWrapper "ERROR - ", "Grand total # of erroes in TO_O2K7_CheckPost_toPF_20023 script for all the user are : " & GlobalErrorNum,"FAILURE"
    Reporter.ReportEvent micFail,
    "TO_O2K7_CheckPost_toPF_20023", "Grand total # of erroes in TO_O2K7_CheckPost_toPF_20023 script for all the user are : " & GlobalErrorNum
Else
Reporter.ReportEvent micPass,
"TO_O2K7_CheckPost_toPF_20023", "Successfully completed running script
TO_O2K7_CheckPost_toPF_20023"

LogWrapper "INFO - ", "Successfully completed running
script TO_O2K7_CheckPost_toPF_20023", "SUCCEEDED"

End If

19 Appendix B – Output files Automation Code

LOGS:

2009.04.2.15.42.48-INFO - Atomic Operation: AO_IAP_StartPuTTY_151 - Start
PuTTY PuTTY started--

2009.04.2.15.59.16-INFO - Atomic Operation: Test - Start PuTTY

[AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.2.15.59.21-INFO - Atomic Operation: Test - Login to PuTTY

[AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.2.15.59.31-INFO - Atomic Operation: Test - Login to PuTTY

[AO_IAP_LoginPuTTY_152] Logged into PuTTY--
2009.04.2.15.59.32-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.2.15.59.42-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.2.16.15.04-INFO - Atomic Operation: Test - Start PuTTY
[AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.2.16.15.10-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.2.16.15.19-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.2.16.15.21-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.2.16.15.31-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.2.18.31.03-INFO - Atomic Operation: Test - Start PuTTY
[AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.2.18.31.08-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.2.18.31.18-INFO - Atomic Operation: Test - Login to PuTTY
[AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.3.10.16.10-INFO - Atomic Operation: Test1 - Start PuTTY
[AO_IAP_StartPuTTY_151] PuTTY
2009.04.3.10.24.45-ERROR - Atomic Operation: Test1 - Start PuTTY

[AO_IAP_StartPuTTY_151] PuTTY

2009.04.3.11.46.58-INFO - Atomic Operation: Test1 - Copy of Enter the command path in the cmd prompt Entered the command-

2009.04.3.24.21.48-INFO - Atomic Operation: Test1 - Copy of Enter the command path in the cmd prompt Entered the command-./collectLogs

2009.04.3.13.03.35-INFO - Atomic Operation: Test1 - Copy 2 of Enter the command path in the cmd prompt Entered the command-vl command

2009.04.8.24.01.44-INFO - Atomic Operation: AO_IAP_StartPuTTY_151 - Start PuTTY PuTTY started--

2009.04.9.14.42.53-ERROR - Atomic Operation: AO_IAP_ChkForProcessCompletion_3 - Enter the command path in the cmd prompt Failed to Enter the command to grep if the process is running-ps -ef | grep '/usr/bin/perl /opt/bin/collectLog*' | grep -v grep > /root/temp/rlog

2009.04.9.17.44.35-INFO - Atomic Operation: AO_IAP_viFilePuTTY_4 - Enter the command path in the cmd prompt Entered the command to grep if the process is running-:q!

2009.04.9.17.56.02-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - Enter the command path in the cmd prompt [AO_IAP_ChkForProcessCompletion_3] Entered the command to grep if the process is running-ls >
2009.04.17.56.08-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - Enter the command path in the cmd prompt [AO_IAP_viFilePuTTY_4] Entered the command to grep if the process is running-vi /

2009.04.10.24.20.00-INFO - Atomic Operation: AO_IAP_DelContentNotepad_06 - SaveResultsInNotepadFromCLUI Deleting content of Notepad—

2009.04.10.24.20.01-ERROR - Atomic Operation: AO_IAP_DelContentNotepad_06 - SaveResultsInNotepadFromCLUI Failed to identify Notepad to delete contents--

2009.04.10.17.51.31-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is running &

2009.04.10.17.51.47-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - Enter the command path in the cmd prompt [AO_IAP_ChkForProcessCompletion_3] Entered the command to grep if the process is running-ps -ef | grep '/usr/bin/perl /opt/bin/collectLog*' | grep -v grep >

2009.04.10.17.51.52-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - Enter the command path in the cmd prompt [AO_IAP_viFilePuTTY_4] Entered the command to grep if the process is running-

2009.04.13.24.49.10-INFO - Atomic Operation: AO_IAP_SaveZipFileNameToExcel_11 - SaveResultsInNotepadFromCLUI Copying zip file names to excel –C: \Putty\TO_IAP_PuTTYCmds.xls

2009.04.13.15.54.31-INFO - Atomic Operation: AO_IAP_CompareZipFileNames_12 - SaveResultsInNotepadFromCLUI Checking for the zip files, all zipfiles present with proper names--
2009.04.13.17.10.17-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_1 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying search command output to notepad--temp1.txt

check for process status [AO_IAP_ChkForProcessCompletion_03] Entered the command to grep if the process is running-ps -ef | grep '/usr/bin/perl /opt/bin/collectLog*' | grep -v grep > /root/temp/rlog

2009.04.14.14.39.09-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command to vi the file [AO_IAP_viFilePuTTY_04] Entered the command to grep if the process is running-vi /root/temp/rlog

2009.04.14.14.40.29-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command path in the cmd prompt [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running-:q!

2009.04.14.14.40.36-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - SaveResultsInNotepadFromCLUI [AO_IAP_DelContentNotepad_06] Deleting content of Notepad--C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

2009.04.14.14.40.47-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter command to check for process status [AO_IAP_ChkForProcessCompletion_03] Entered the command to grep if the process is running-ps -ef | grep '/usr/bin/perl /opt/bin/collectLog*' | grep -v grep > /root/temp/rlog

2009.04.14.14.52-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command to vi the file [AO_IAP_viFilePuTTY_04] Entered the command to grep if the process is running-vi /root/temp/rlog
2009.04.14.14.41.04-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command path in the cmd prompt [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running--:q!


2009.04.14.17.54.59-INFO - Atomic Operation: AO_IAP_cd_applianceLogs_08 - Enter command - cd applianceLogs Entered the command to exit PuTTY-exit
2009.04.14.17.55.29-ERROR - Atomic Operation: AO_IAP_cd_applianceLogs_08 - Enter command - cd applianceLogsFailed to Enter the command to exit PuTTY-exit


2009.04.14.18.08.40-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.14.18.08.50-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.14.18.08.57-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is

2009.04.14.18.09.01-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter command to exit the PuTTY session [AO_IAP_ExitPuTTYSession_15] Entered the command to exit PuTTY-exit


2009.04.14.18.10.38-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--
2009.04.14.18.11.04-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.14.18.11.06-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered
the command to grep if the process is running

2009.04.14.18.11.08-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Enter command to exit the PuTTY session [AO_IAP.ExitPuTTYSession_15] Entered
the command to exit PuTTY-exit

- Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY failed to start--

Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.14.18.13.30-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered
the command to grep if the process is running-/opt/bin/collectCRLogs >
/root/temp/log.out

2009.04.14.18.13.36-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Enter command to exit the PuTTY session [AO_IAP.ExitPuTTYSession_15] Entered
the command to exit PuTTY-exit
Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is running-/opt/bin/collectCRLogs >
/root/temp/log.out

Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY failed to start--

Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is
2009.04.15.9.51.16-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter command to exit the PuTTY session [AO_IAP_ExitPuTTYSession_15] Entered the command to exit PuTTY-exit

2009.04.15.9.51.20-ERROR -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY failed to start--

2009.04.15.9.52.48-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.15.9.53.14-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.15.9.53.24-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.15.9.53.30-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is running-../opt/bin/collectCRLogs&

2009.04.15.9.53.34-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter command to exit the PuTTY session [AO_IAP_ExitPuTTYSession_15] Entered the command to exit PuTTY-exit

2009.04.15.9.53.39-ERROR -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY failed to start--

2009.04.15.9.55.14-INFO -  Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter the command path in the cmd prompt [AO_IAP_CollLogsBackgrnd_09] Entered the command to grep if the process is running-../opt/bin/collectCRLogs&
2009.04.15.9.55.21-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Enter command to exit the PuTTY session [AO_IAP_ExitPuTTYSession_15] Entered the command to exit PuTTY-exit

2009.04.15.9.55.25-ERROR - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY failed to start--

2009.04.15.10.14.25-INFO - Atomic Operation: AO_IAP_CollLogsBackgrnd_09 - Enter the command to CollectLogs Entered the command to grep if the process is running-/opt/bin/collectCRLogs > /root/temp/log.out

2009.04.15.10.19.56-INFO - Atomic Operation: AO_IAP_CollLogsBackgrnd_09 - Enter the command to CollectLogs Entered the command to Collect Logs-/opt/bin/collectLogs > /root/temp/log.out

2009.04.15.15.17.48-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running-cd /var/log/applianceLogs/

2009.04.15.15.20.17-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running-cd /var/log/applianceLogs/

2009.04.15.15.20.56-INFO - Atomic Operation: AO_IAP_grepPatternsInZip_14 - Enter command to grep patterns in zip file Entered the command to grep for common patterns in Zip file-egrep 'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' > /root/temp/verifyzip
2009.04.15.15.21.29-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running - cd /var/log/applianceLogs/

2009.04.15.15.21.36-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the command to grep for common patterns in Zip file - egrep 'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' > /root/temp/verifyzip

2009.04.15.15.21.41-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command to vi the file - vi /root/temp/verifyzip

2009.04.15.15.21.53-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running - :q!

2009.04.15.15.22.01-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Delete contents of Notepad [AO_IAP_DelContentNotepad_06] Deleting content of Notepad -- C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

2009.04.15.15.22.07-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the command to grep for common patterns in Zip file - egrep 'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' > /root/temp/verifyzip

2009.04.15.15.22.12-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command to vi the file - vi /root/temp/verifyzip
2009.04.15.15.22.15-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad - output.txt--

2009.04.15.15.22.20-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad--temp1.txt

2009.04.15.15.22.24-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running:-q!

2009.04.15.15.23.39-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running-cd /var/log/applianceLogs/

2009.04.15.15.23.46-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the command to grep for common patterns in Zip file-egrep 'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' 10.0.96.3 > /root/temp/verifyzip

2009.04.15.15.23.50-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command to vi the file - vi /root/temp/verifyzip

2009.04.15.15.23.54-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad - output.txt--
2009.04.15.15.23.58-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad--temp1.txt

2009.04.15.15.24.02-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running-:q!

2009.04.15.15.24.17-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the command to grep for common patterns in Zip file-egrep

'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' 10.0.96.4 >
/root/temp/verifyzip

2009.04.15.15.24.33-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running-:q!


2009.04.15.15.24.43-ERROR - Atomic Operation:
STO_IAP_ChkForZipFileValidity_03 - Check for Zip file validity A few files in zip format have readable contents.

2009.04.15.15.26.08-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running-cd /var/log/applianceLogs/
2009.04.15.15.26.14-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the
command to grep for common patterns in Zip file - egrep
'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' 10.0.96.3 >
/root/temp/verifyzip

2009.04.15.15.26.19-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command
to vi the file - vi /root/temp/verifyzip

2009.04.15.15.28.20-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the
command to grep if the process is running - :q!

2009.04.15.15.28.28-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Delete contents of Notepad [AO_IAP_DelContentNotepad_06] Deleting content of
Notepad--C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

2009.04.15.15.35.31-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Enter command - cd applianceLogs [AO_IAP_cd_applianceLogs_08] Entered the
command to grep if the process is running - cd /var/log/applianceLogs/

2009.04.15.15.35.37-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 -
Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the
command to grep for common patterns in Zip file - egrep
'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' 10.0.96.3 >
/root/temp/verifyzip
2009.04.15.15.35.42-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command to vi the file - vi /root/temp/verifyzip

2009.04.15.15.35.45-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad - output.txt--

2009.04.15.15.35.49-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad--temp1.txt

2009.04.15.15.35.54-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running:-q!

2009.04.15.15.36.03-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Delete contents of Notepad [AO_IAP_DelContentNotepad_06] Deleting content of Notepad--C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

2009.04.15.15.36.09-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter command to grep patterns in zip file [AO_IAP_grepPatternsInZip_14] Entered the command to grep for common patterns in Zip file-egrep

'INFO|ERROR|WARN|FATAL|DEBUG|com|IP ADDRESS' 10.0.96.4 >
/root/temp/verifyzip

2009.04.15.15.36.14-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to vi the file [AO_IAP_viPuttyVerifyZip_13] Entered the command to vi the file - vi /root/temp/verifyzip
2009.04.15.15.36.17-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad - output.txt--

2009.04.15.15.36.22-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad--temp1.txt

2009.04.15.15.36.27-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running:-:q!

2009.04.15.15.36.35-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Delete contents of Notepad [AO_IAP_DelContentNotepad_06] Deleting content of Notepad--C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

2009.04.15.15.36.36-INFO - Atomic Operation: STO_IAP_ChkForZipFileValidity_03 - Check for Zip file validity Verifying for valid zip files--

2009.04.16.16.36.54-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.16.16.37.20-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.16.16.37.30-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 - Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.16.16.37.41-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Start PuTTY [AO_IAP_StartPuTTY_151] PuTTY started--

2009.04.16.16.38.06-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Login to PuTTY [AO_IAP_LoginPuTTY_152] Logging in to PuTTY--

2009.04.16.16.38.16-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Login to PuTTY [AO_IAP_LoginPuTTY_152] Logged into PuTTY--

2009.04.16.16.38.27-INFO - Atomic Operation: STO_IAP_WaitTillCollLogOver_01 -
Enter command to check for process status [AO_IAP_ChkForProcessCompletion_03]
Entered the command to grep if the process is running-ps -ef | grep '/usr/bin/perl
/opt/bin/collLo*\| grep -v grep > /root/temp/rlog
running-cd /var/log/applianceLogs/

2009.04.16.19.28.26-INFO - Atomic Operation:
STO_IAP_ChkCollLogsOpFileNames_02 - Enter the command to collect the Zip file
names [AO_IAP_GetZipFileNames_10] Entered the command to grep if the process is running-
ls > /root/temp/rlog

2009.04.16.19.28.31-INFO - Atomic Operation:
STO_IAP_ChkCollLogsOpFileNames_02 - Enter the command to vi the file
[AO_IAP_viFilePuTTY_04] Entered the command to grep if the process is running-
vi /root/temp/rlog

2009.04.16.19.28.34-INFO - Atomic Operation:
STO_IAP_ChkCollLogsOpFileNames_02 - SaveResultsInNotepadFromPutty
[AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad -
output.txt--
2009.04.16.19.28.38-INFO - Atomic Operation:

STO_IAP_ChkCollLogsOpFileNames_02 - SaveResultsInNotepadFromPutty

[AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad-temp1.txt

2009.04.16.19.28.43-INFO - Atomic Operation:

STO_IAP_ChkCollLogsOpFileNames_02 - Enter the command to exit the vi editor

[AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running--:q!

2009.04.16.19.31.06-INFO - Atomic Operation:

STO_IAP_ChkCollLogsOpFileNames_02 - Enter command - cd applianceLogs

[AO_IAP_cd_applianceLogs_08] Entered the command to grep if the process is running-

cd /var/log/applianceLogs/

2009.04.16.19.31.11-INFO - Atomic Operation:

STO_IAP_ChkCollLogsOpFileNames_02 - Enter the command to collect the Zip file names

[AO_IAP_GetZipFileNames_10] Entered the command to grep if the process is running-

ls > /root/temp/rlog

2009.04.16.19.35.08-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Populate the Excel from Notepad

[AO_IAP_RdLine#NotepadSaveToExcel_07] Reading line#1 from Notepad

2009.04.16.19.35.11-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Delete contents of Notepad

[AO_IAP_DelContentNotepad_06] Deleting content of Notepad--
2009.04.16.19.35.13-ERROR - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Check for Zip file validity File missing for command--find aaa/ -name 'bbbb' > /root/temp/findFiles

2009.04.16.19.35.26-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Enter the command to find the file names [AO_IAP_RunFindCmd_18] Entered the command to find if the file exists-find

2009.04.16.19.35.31-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Enter the command to vi the file [AO_IAP_viPuttyFindFiles_20] Entered the command to vi the file - vi /root/temp/findFiles

2009.04.16.19.35.34-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad - output.txt--

2009.04.16.19.35.39-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - SaveResultsInNotepadFromPutty [AO_IAP_SavePuttyTextToNotepad_01] Copying fgsearch command output to notepad--temp1.txt

2009.04.16.19.35.43-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running:-q!

2009.04.16.19.35.49-INFO - Atomic Operation: STO_IAP_ChkForFilesPresent_04 - Populate the Excel from Notepad [AO_IAP_RdLine#NotepadSaveToExcel_07] Reading line#1fromNotepad
Delete contents of Notepad [AO_IAP_DelContentNotepad_06] Deleting content of Notepad--C:\Automation\IM\PlatformFiles\2.10\Logging\Documents\temp1.txt

Check for Zip file validity File missing for command--find /var/log/zebra/ -name '*' > /root/temp/findFiles

Enter the command to find the file names [AO_IAP_RunFindCmd_18] Entered the command to find if the file exists-find ddd -name 'eee' > /root/temp/findFiles

Enter the command to vi the file [AO_IAP_viPuttyFindFiles_20] Entered the command to vi the file - vi /root/temp/findFiles

Copying fgsearch command output to notepad - output.txt--

Copying fgsearch command output to notepad--temp1.txt

Enter the command to exit the vi editor [AO_IAP_ExitFilePuTTY_05] Entered the command to grep if the process is running:-:q!
Populate the Excel from Notepad [AO_IAP_RdLine#NotepadSaveToExcel_07] Reading line#1 from Notepad

Atomic Operation screenshot:

Figure 22: Failed Screenshot of Atomic Operation

SubTestOperation:
Appendix B – Survey Questions

The QA questionnaire will take 15 minutes of your time. Please click the submit button once all questions are answered. The questionnaire can be taken anonymously. Thank you for your inputs.

Question 1: Does your project have a test plan specification?

a) Yes

b) No
**Question 2:** Does your project have a test design specification?
   a) Yes  
   b) No

**Question 3:** If “Yes” for question 2, then is the test design specification developed for each major feature?
   a) Yes  
   b) No

**Question 4:** Does your project have a test case specification?
   a) Yes  
   b) No

**Question 5:** If “Yes” for question 4, then is the test design specification developed for each major feature?
   a) Yes  
   b) No

**Question 6:** Are the test cases for each major feature designed to cover the customer specific use cases?
   a) Yes  
   b) No
c) Partially

**Question 7:** Are the test cases for each major feature designed to cover the state based scenarios?

a) Yes

b) No

c) Partially

**Question 8:** Are the test cases reviewed by engineering?

a) Yes

b) No

**Question 9:** How do you rate your quality process in terms of completeness and clarity in definition of the process?

a) 0-25(Not up to standards)

b) 25-50(Just started)

c) 50-75(Meets Expectations)

d) 75-100(Great, No Improvement Needed)

**Question 10:** Are the automation test cases prioritized?

a) Yes
b) No

c) Partially

**Question 11:** Does the automation team execute the test cases before development of the automation scripts?

a) Yes

b) No

c) Partially

**Question 12:** Does your QA team follow agile process?

a) Yes

b) No

**Question 13:** How many customer escalations do you get per 2 weeks?

a) 1-4

b) 5-9

c) 10-15

d) More than 15

e) Not applicable as the product is not shipped to customer

**Question 14:** Did you lose any customer due to defects found by customers?
Question 15: What is your quality level? Quality level is defined based on the defects found at various phases of project which should be bell curve.

   a) 0-25(Not up to standards)
   b) 25-50(Average)
   c) 50-75(Meets Expectations)
   d) 75-100(Great, No Improvement Needed)