EIT Automation, Tools and Utilities

Submitted By

Jinal Dhruv

13MCEC04

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2015

EIT Automation, Tools and Utilities

Major Project

Submitted in partial fulfillment of the requirements

for the degree of

Master of Technology in Computer Science and Engineering

Submitted By

Jinal Dhruv

(13MCEC04)

Guided By

Prof. Tejal Upadhyay

(Assistant Professor, ITNU)

and

Mr. Punnet Saxena

(QA Manager, Oracle India Pvt. Ltd.)

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

INSTITUTE OF TECHNOLOGY

NIRMA UNIVERSITY

AHMEDABAD-382481

May 2015

Certificate

This is to certify that the major project entitled **"EIT Automation, Tools and Utilities"** submitted by **Jinal Dhruv (Roll No: 13MCEC04)**, towards the partial fulfillment of the requirements for the award of degree of Master of Technology in Computer Science and Engineering(CSE) of Institute of Technology, Nirma University, Ahmedabad, is the record of work carried out by him under my supervision and guidance. In my opinion, the submitted work has reached a level required for being accepted for examination. The results embodied in this project, to the best of my knowledge, haven't been submitted to any other university or institution for award of any degree or diploma.

Prof. Tejal Upadhyay
Guide & Assistant Professor,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Prof. Vijay Ukani
Associate Professor,
Coordinator M.Tech - CSE
Institute of Technology,
Nirma University, Ahmedabad

Dr. Sanjay Garg
Professor and Head,
CSE Department,
Institute of Technology,
Nirma University, Ahmedabad.

Dr. K. Kotecha
Director,
Institute of Technology,
Nirma University, Ahmedabad

Statement of Originality

I, **Jinal Dhruv**, Roll. No. **13MCEC04**, give undertaking that the Major Project entitled "**EIT Automation, Tools and Utilities**" submitted by me, towards the partial fulfillment of the requirements for the degree of Master of Technology in **Computer Science & Engineering** of Institute of Technology, Nirma University, Ahmedabad, contains no material that has been awarded for any degree or diploma in any university or school in any territory to the best of my knowledge. It is the original work carried out by me and I give assurance that no attempt of plagiarism has been made. It contains no material that is previously published or written, except where reference has been made. I understand that in the event of any similarity found subsequently with any published work or any dissertation work elsewhere; it will result in severe disciplinary action.

_____

Signature of Student

Date:

Place:

Endorsed by

Prof. Tejal Upadhyay

(Signature of Guide)

Acknowledgements

It gives me immense pleasure in expressing thanks and profound gratitude to **Prof. Tejal Upadhyay**, Assistant Professor, Computer Science Department, Institute of Technology, Nirma University, Ahmedabad and **Mr. Puneet Saxena**, QA manager, Oracle India Pvt. Ltd. for their valuable guidance and continual encouragement throughout this work. The appreciation and continual support they have imparted has been a great motivation to me in reaching a higher goal. their guidance has triggered and nourished my intellectual maturity that I will benefit from, for a long time to come.

It gives me an immense pleasure to thank **Dr. Sanjay Garg**, Hon'ble Head of Computer Science and Engineering Department, Institute of Technology, Nirma University, Ahmedabad for his kind support and providing basic infrastructure and healthy research environment.

A special thank you is expressed wholeheartedly to **Dr. K. Kotecha**, Hon'ble Director, Institute of Technology, Nirma University, Ahmedabad for the unmentionable motivation he has extended throughout course of this work.

I would also thank the Institution, all faculty members of Computer Engineering Department, Nirma University, Ahmedabad for their special attention and suggestions towards the project work.

I want to express my deep faith and love to my parents for their everlasting belief and support. At last, I cannot forget my friends,colleagues and all mighty.

- Jinal Dhruv

13MCEC04

Abstract

It is important in software industry to deliver the right product at the right time to customer with all the quality checks. Enterprise manual testing becomes difficult and time consuming if one tries various input combinations. Moreover, it is costly to manually analyze all results. This project is motivated by the drawbacks of manual regression testing and how to overcome those drawbacks using automated tools.

In case of manual testing, changes in source code, hardware configuration or operating system leads to repetition of whole process. These efforts can be reduced by means of an automation. Automation testing does the predefined actions written through code and it is also helpful in analyzing result. Automating this process also serves as a reporting solution which reports the success or failure of a particular test.

For doing it, we came up with an Enterprise Integration Testing [EIT] Tools and Utilities which are used by the EIT team to manage their day to day integration testing. It contains four major functionalities, EIT Dashboard, EIT Reports, EIT Tools, EIT Task Management with multiple utilities under each of them.

Abbreviations

EIT	Enterprise Integration Testing.
HPQC	Hewlett Packard Quality Center.
OTM	Oracle Test Manager.
FTR	Fault Tolerance Report.
RA	Retail Analytic.
PTM	Product Traceability Matrix.
RMS	Retail Merchandising System.
SIM	Store Inventory Management.
WMS	Warehouse Management Module.
POS	Point-of-Sale.
RIB	Retail Integration Bus.
BTV	Business Transaction Validation.
ADF	Application Development Framework.
GUI	Graphical User Interface.
RTV	Return To Vendor.
ReSA	Retail Sales Audit.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

1.1 Overview

Quality, time and cost are the main assets in the building of the any software system. There are many testing approaches available to improve the quality and efficiency of the software system and these approaches are rapidly adopted by the software organizations. If defects and faults in the software systems are still present after the testing phase it causes a huge loss of human resource, time and cost. Therefore it is very important for the organizations to perform the quality analysis carefully.

There are two well known methods majorly used in the industries: manual and automation testing. Every software product needs to be tested adequately, but as quickly and thoroughly as possible with minimal resources. To accomplish this goal automated testing is used. Automation testing satisfies the quality and time characteristics for the software system.

1.2 Motivation

Main goal of test automation is to arrange ideas, tools and assumptions to provide some core functionalities. These functionalities include logging, reporting, monitoring, functioning and error recovering. There are many approaches to do automation and all have different methodology. It mainly reduces the job of test engineers by assigning same job to computers.

Automating manual testing process includes compliance of requirements, database validations, load performance ...etc. With growing demand of support in testing, new features and capabilities are added. Regression testing is preferable to reduce risk and to find bug in the code in fastest way with less effort. We also try to achieve this by automation.

1.3 Oracle Retail Merchandise System

Oracle retail merchandising system (RMS) is the central repository that stores and manages all data virtually. It ensures data integrity by means of Retail Integration Bus. Key functionalities of RMS are replenishment, inventory management and item maintenance. Main goal of these functionalities is to provide easy access to the information which is important for the organization. It certainly helps to take critical decisions. It also helps in achieving targets related to sale and profits.[3]

Oracle offers many products as solutions to retailers. They all are categorized under one roof as Oracle Retail Merchandising Operations Management. Merchandize typically includes 4 levels i.e., Suppliers, Distribution Centers, Stores and then to customers. RMS includes applications that helps the merchandisers to maintain items of different types and their cost.

Data transfers from one application to other through Retail Integration Bus. RMS forms the central repository to hold the data. Point-of-Sales (POS), Store Inventory Management (SIM) and Warehouse Management System (WMS) forms store operations of Retail chain.

POS is a high end graphical retail solution to the customers. In todays world shoppers need fast and reliable tool to make their shopping experience memorable and convert them into customers for life. Retailers need access to real time customer, marketing data and product to achieve their retail objectives.

WMS defines individual warehouse types like high-rack storage, block storage, picking area and etc. It manages information regarding goods at storage bin level. In order to

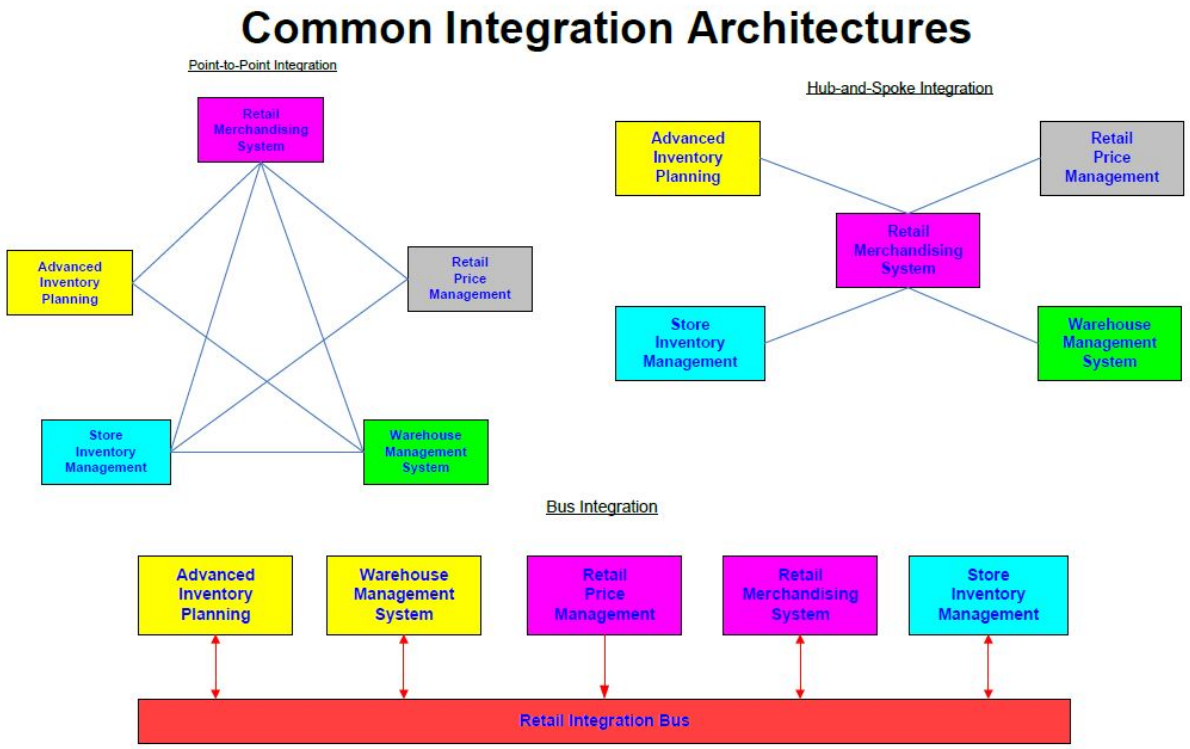


Figure 1.1: Integration Architectures[3]

associate the data with other applications, warehouse number is assigned to each storage location.

Inventory present in a store is tracked by Store Inventory Management (SIM). SIM receives, tracks, and transfers merchandize accurately and efficiently. SIM operates at three tiers namely at Server tier, Client tier and Data tier. The server tier is developed and deployed as J2EE application inside Web logic application server and which mostly works as standalone application for handheld devices. The data tier consists of Oracle 11g database and LDAP directory.

1.4 Problem Definition

Design, Development and Implementation of Automation Tool and Utilities which are used by Enterprise Integration Testing team for reducing manual efforts.

1.5 Objectives

Goal of this dissertation is to automate functionalities which are regularly used in test execution in order to . . .

- Reduce cycle time of testing by automating functionalities and reducing efforts.
- Improving consistency and quality in execution of test.
- Increasing productivity if manual testing process.
- Making a report by tracking and analyzing work of team members.
- Developing an easy to understand interface to manage the process.

1.6 Organization of Dissertation

This Chapter describes problem definition in abstract manner and gives an overview of retail process. Chapter 2 is about literature survey which also includes different types of testing. Chapter 3 gives an insight to Automated Software Testing. Chapter 4 is all about Software Requirement Specification. Chapter 5 describes high level design with main features and functionalities. Chapter 6 gives insight of some utilities and tools developed. Chapter 7 sums up details and concludes thesis.

Chapter 2

Literature Survey

2.1 Requirement of Literature Survey

Applying different techniques and evaluating results is useless until you know what is actually happening behind the screen. So, it was very important to understand characteristics of retail cycle and automation as well as how it handles data by understanding methodology. This greatly helps to analyze results effectively.

2.2 Literature Review

Literature review helped in identifying and measuring effectiveness of a test automation framework and technique through various factors such as Scalability, Test- Reusability, Maintainability and Learning Curve for non technical users.[1]

1. Bach. J says that automates tests automated process requires very less human intervention. It eliminates human error and give faster results. Thus, it saves labor cost.

2. C. Nagle points out problems with test automation. he describes tools which are not strong enough to handle all different applications. Different test automation strategies and automation methodologies are also discussed.

3. J. Kent explains code structures for automation with related merits. Test frameworks and review progress are also considered. To provide effective automation, maturity model for automation is described.

4. E. H. Kim et al. discusses some new test automation framework architectures. they are easy to manage and also includes different commanding structures. Reusability is the main aspect which is taken care of in this approach.

5. J. Hinz et al. categorizes test automation process in different generations.

2.3 Need for Testing

The best way to ensure a quality of a product is testing and it is crucial phase in software development life cycle. It finds out information related to quality of the product by investigating in a well defined manner. it also takes care of the context with which product needs to work. Overall, it gives better vision for product and avoids failures.

Main objectives of testing are finding bugs, errors and defects which are not discovered in the development process. They are not desirable. There are different techniques available for testing. Some of them are Unit testing, Integration testing, Quality Acceptance testing . . . etc.

2.4 Regression Testing

Regression testing forms major part of integration testing. Whenever a minor change has been made to some part of the system, the whole system is supposed to undergo testing to find any chances of failure. In case of complex application, regression testing becomes costly in terms of time and money.

The main goal of regression testing is to make sure that all changes are appropriate. There should not be any adverse effect on features or performance. It is performed with the help of test cases in previous versions of software. Number of test cases should not grow with software. For this, many techniques have been evolved. [2]

Test suite selection techniques try to reduce the cost of testing by running a subset of the tests, such as those that execute the modified source code, in order to ensure that the updated program still operates correctly.

Regression Testing is most tedious, error prone and time consuming testing type as it

requires complete execution of all set of test cases whenever the system undergoes some kind of change. Hence, there is a need of term called as Test Automation which eases the task of Regression Testing.

Chapter 3

Automated Software Testing

3.1 Type Of Software Testing

Software can be effectively tested by two testing techniques which are described below:
Manual testing is more traditional approach,In which manually execute the testcases which relate with all functional area of product to identify any unexpected behavior or bug and at the end submit the report with the defects which are found during testing process.
Automation testing is very popular technique to reduce time and cost relate with testing activity.This technique take support of automation testing tool for testcase execution and validation process.

3.2 Requirement of Automation in Testing

Importance of software is increasing and mean while complexity as well as size of software is also increased.Because of this its very important to maintain quality of software.If quality of software is not upto satisfying level than it is going to increase human effort for maintenance which leads to increase overall cost of software.If system contain faults than its going to decrease quality with increasing complexity and size of software. Software testing assesses the quality of the product.Demand for good quality software makes more pressure on testing process.[4]

3.2.1 What is automated software testing?

Automation software testing is writing the code which in turns test the code. it performs actions as defined and compares results. Based on behaviors expected, success or failure

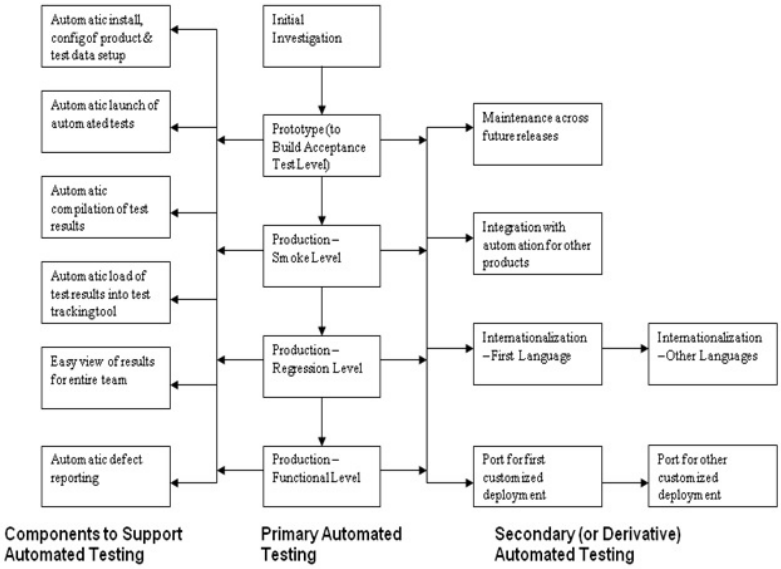


Figure 3.1: Development Flow for Test Automation

of test execution is decided.

3.2.2 What makes automated software testing so important?

1. **Perform tests which would be difficult**

   Some software testing activities are not time and cost effective if we perform it manually.Manual testing is also not able to find some type of bugs which are present in software in effective manner.This tasks can be very well handle by test automation.

2. **Regression tests**

   When ever new patch is ready to support new feature or remove old bugs,its very probable that this changes can affect previous functionality of software.To overcome this we need to test all functionality after producing new patches for system.This testing is known as regression testing.For this type of task test automation can provide very effective solution because we can re-run each test scenario as many time as we want.

3. **Consistency & Repeatability**

   Automation testing provide consistency because every time its going to run same way and produce consistence output.If we doing same work with manual testing than it might not be as consistence as automation because manual testing involves

human error.Human can miss out one testcase or validation rule which might lead to unable detecting bug.Due to consistence performance of test automation,we can able to re-run every test scenario as many time as we want.

4. **Speed**

   Testcasaes can run concurrently on different environments which have different configuration.

5. **Reusability**

   Less time require to run automated testcase than manual run.Because of this we are able to get faster feedback.

3.3 Key aspects of Test Automation

This section explains required capabilities of automation Framework.These properties can be seen as functional requirements for test automation framework.

- **Executing Tests**

  Framework should set the environment,check all precondition and start execute testscript after execute command is given to it.

- **Control Over Test Execution**

  Framework should be able to support scheduling testscript at specific time or after certain events/precondition are met.

- **Error Handling**

  Test framework should be able to recover from error occur during execution and complete its given task without any manual intervention.

- **Result Verification**

  Framework should able to provide way which identify deviation between actual and expected output.

- **Debugging**

  Framework should be able to support Debugging by providing pause functionality while testscript are running so that tester can see the state of testscript.

- **Status Of Test**

  Framework should able to provide status after test execution complete.If all the verification step are clear than its should give pass otherwise fail status with proper reason behind failure.Various status with there significant meaning are as listed below:

  - **Pass:**This indicates that testcase is passed.
  - **Fail:**This status indicates that testcase is failed.
  - **Warning:** After executing recover scenario if testcase able to pass than we need to log message that after recovering this failure state its passed.These messages are known warning.
  - **Info:**These type of message are needed to describe various checkpoint in test-cases like start testcase,validate values,mismatch.

- **Reporting**

  Detail test report contain all the above message.but some time this type of report when we need to glance result like how many testcase are failed and how many are passed.We are not looking for reason behind failure but just statistics than we need to generate another report which summarize the all testcases results.

Chapter 4

Software Requirement Specification

4.1 Automation Agile Delivery

Agile development methodology followed and details are as follows ...

Product Backlogs: Collection of requirements from manual test teams and external teams who use the Enterprise Integration Testing utilities and tracking them as product backlogs.

Scrum Sprints: Two sprints in a month planned and delivery of sprint on 15th and 30th of every month. Sprint meetings are done daily. Every sprint the code developed will be tested and then released into production website.

Demo: Each sprint will have weekly demo in our case two demos in each sprint one for design review demo and another for actual feature completed.

4.2 Software System Attributes

Portability: The test automation done is portable amongst different operating systems and different browsers.

Reliability: The Run results are reported in an appropriate manner after running every test case. Whether the test case is failed or passed can be checked and whether the failure reason is an application issue or an automation issue can also be analyzed from the results.

Correctness: All the automated test cases for a particular version of a product are delivered together as a codebase after unit testing and batch testing. So there are negligible

possibilities of automation failures.

Reusability: Libraries, Object repositories, Recovery scenarios are developed as a part of framework development for test automation of a product. These can be reused across the different versions of the same product with minimal changes. Minimized effort is required for repeated regression cycles.

Maintenance: Test automation provides complete support for application's new enhancements and existing features modification since framework developed once can be reused with minimal changes.

Performance:

- Less time for development
- Easy to understand.
- Easy to train new personnel.

4.3 Functional Requirements

- Utilities must be executed unattended. That is verification of test result, reporting errors, analyzing test execution and generating report of test execution.
- Utilities should be executed with minimal effort and there should be a provision to execute the same at pre scheduled time.
- Errors caused while executing must be handled properly and should be notified accordingly.
- Some status should be assigned to all test executions which leads to effective reporting. One message describing result of execution is desirable.
- To complete execution in short time, it should be possible to perform same task from different machine at a same time and it must not lead to incosistancy.
- Report should be customized to make an analysis easier.

4.4 Nonfunctional Requirements

- All tools and utilities should be easy to use without the need of any technical or porgramming skills.

- Maintenance must be easy for code developed.
- Modular approach should be followed because some of utilities are very big.
- Optimization is the key factor while developing utility or tool.
- Proper naming and coding conventions are equally important and it must be documented well.

4.5 Automation Infrastructure

4.5.1 Tools and Technologies

- JSP as a User Interface.
- Servlet as a controller.
- Oracle WebLogic as a Web server.
- Oracle database to store EIT utilities' specific information.
- JDBC Connections to connect RMS(Retail Merchandising System), WMS(Warehous Management System), SIM(Store Inventory Management), POS(Point of Service).
- Apache SVN(Subversion) for source code configuration management.
- d3.js for dynamic graphs.
- Cache mechanism to reduce response time.

4.5.2 QA Automation Environment

Virtual automation box which hosts the jsp website, weblogic server, oracle database and batch job scheduler. The automation box details are ...

- Msp52458.us.oracle.com
- Processor: intel xeon 3.00ghz
- RAM: 16gb
- OS: windows server 2008 r2 enterprise

4.5.3 Directory

Enterprise Integration Testing Utility:this directory holds all the jsp and related files to make war file.

Script: this directory holds all the shell scripts for batches, ftp scripts.

4.5.4 Automation Infrastructure

- Primary Enterprise Integration Testing utilities which is our main source and it will be used by Enterprise Integration Testing manual test team to reduce their effort on testing.
- Secondary Enterprise Integration Testing utilities which will used by automation engineers to develop new features.

4.6 Risk Management

4.6.1 Open Issues

- SVN check-in are not getting followed.
- ADF training for doing complex tasks.
- Manual test teams are not aware of EIT new features to use them, We can plan to have regular demo to manual test teams.

4.6.2 Open Questions

- Currently all the developers working on EIT utilities are interns, Is there a plan to induce FTE?
- What is the process of getting requirements from other teams if they would like to use this utility?

4.6.3 Assumptions

- The utilities are developed for EIT specific needs.
- If other team likes to use this Utility, they will have to host separate website and we can think of bundling the software.

Chapter 5

High Level Design

5.1 Abstract View

Enterprise Integration Testing [EIT] Tools, Utilities is used by the EIT team to manage their day to day integration testing. It contains four major functionalities, EIT Dashboard, EIT Reports, EIT Tools, EIT Task Management with multiple utilities under each of them. EIT Dashboard contains graphical representation of analysis of the foundation data. EIT Reports contains tools for the generating reports for messages covered, foundation data available, system options and its comparison report across environments. EIT Tools is the major functionality which contains tools for Data verification, Transaction validation and Configuration management. EIT Task Management contains tools for updating the daily status and getting reports, tracking any issues faced by the team.

Business Validation tool will be used to reduce the execution cycle time for each test cast and decreases manual testing of individual tables. Configuration will be done as per the availability of test environments, products and databases. Task management is a tool for frequent internal usage. It will be having a provision for secure login of the authorized uses.

EIT tools and utilities will be having capability to be configured with any retail product. Automation scripts can be extended for verification and validation of database will be performed with a single click drastically reducing manual effort and test case execution time.

5.2 Features and Functionalities

5.2.1 Visual Representations

EIT Dashboard: EIT Dashboard contains graphical representation of analysis of the foundation data.

EIT Reports: EIT Reports contains tools for the generating reports for messages covered, foundation data available, system options and its comparison report across environments.

EIT Tools: EIT Tools is the major functionality which contains tools for Data verification, Transaction validation and Configuration management.

EIT Task Management: EIT Task Management contains tools for updating the daily status and getting reports, tracking any issues faced by the team.

5.2.2 Functionalities

- Integration Coverage across EIT applications.
- Message Coverage.
- Product Traceability Matrix.
- Business Transaction Validation.
- HPQC-OTM Migration
- RA Integration
- Team Hand Off Report

5.2.3 Optimization and Generalization

- Configuration will be done as per the availability of test environments, products and databases.
- Business Validation will be used to reduce the execution cycle time for each test cast and decreases manual testing of individual tables.
- Verification and validation of database will be performed with a single click drastically reducing manual report and test case execution time.

- Once developed for EIT team, capability will be enhanced to be configured with other retail products and teams.
- It is assured that tools and utilities work in expected way irrespective of platform or browser.

Chapter 6

Tools and Utilities

6.1 Connection Configuration

6.1.1 Objective

This Utility can be used to add a new connection, modify an existing connection and also view the connection details.

6.1.2 Functional Requirements

- Administrator should be able to log in through single sign-on.
- Administrator should be able to perform 3 tasks: Add environment ,Modify environment and Delete environment.
- Add Environment: admin should be able to add the following environment details ...
  - Environment name
  - Product name
  - Url/hostname
  - Database name/service name
  - Username
  - Password
  - Port number

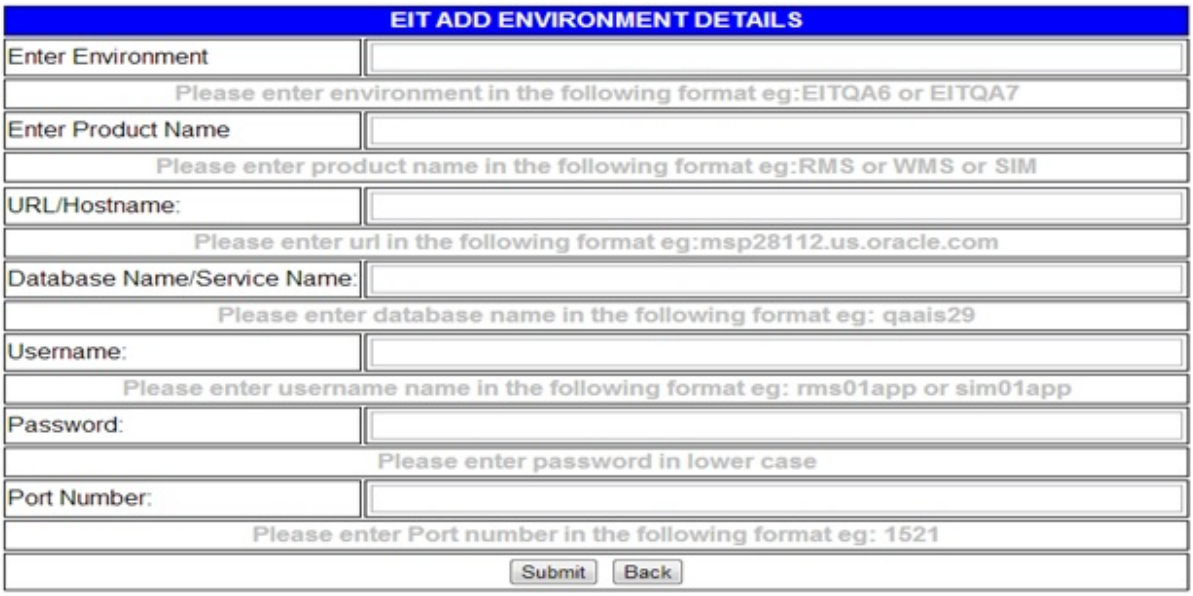


Figure 6.1: Add Environment

- Modify environment: admin should be able to modify the environment by selecting the environment and product for which the details needs to be modified.
- View environment: admin should be able to view the environment details by selecting the environment and product whose details need to be viewed.

6.1.3 GUI Design

- Add Environment: administrator enters the details and clicks submit to add the details. First time environment setup for a product requires the new environment name to be input. For second time the environment can be selected from dropdown for a different product.
- Modify Environment: Administrator selects the environment and product whose details needs to be modified. Administrator edits the fields that needs to be modified and clicks on submit to save the changes.
- View Environment: Administrator selects the environment and the product and clicks on submit in order to view the corresponding environment details.

6.1.4 Database Design

Refer Table 6.1, 6.2 and 6.3 given below.

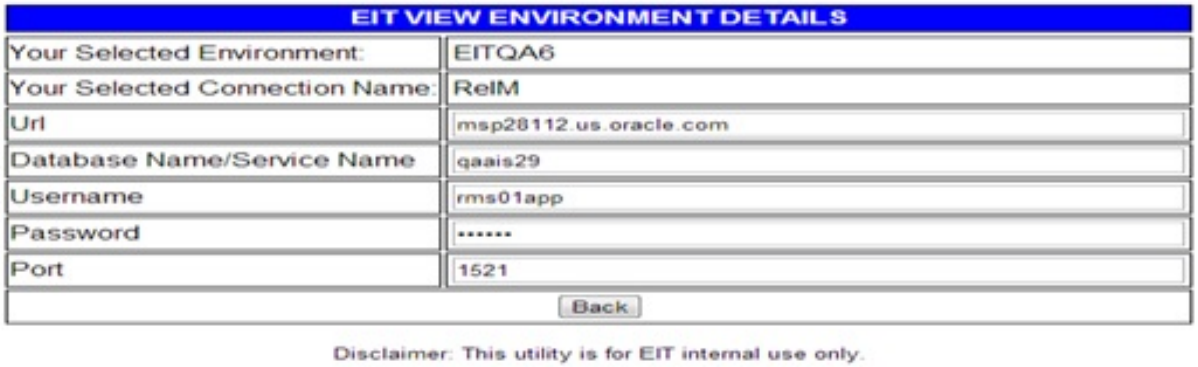


Figure 6.2: View Environment

Field Name	Data Type	Description
Environmentnm	Varchar(40)	Primary key
Productnm	Varchar(40)	Primary key
Url	Varchar(100)	
Dbname	Varchar(40)	
Username	Varchar(40)	
Password	Varchar(40)	
Port	Varchar(40)	
Databasedriver	Varchar(60)	
Commission	Varchar(2)	

Table 6.1: Connection_Config Table

Field Name	Data Type	Description
Environmentnm	Varchar(40)	Primary key

Table 6.2: Environment Table

6.1.5 Implementation Logic

1. Administrator enters the utility after entering the username and password.

2. Administrator can perform the following operations: add environment, modify environment and view environment.

3. Once the environment details are added, then connection can be made dynamically to access the data present in that environment.

4. Connection details can be modified at any given point of time by using the modify environment option.

5. Environment details can also be viewed using the view environment option.

Field Name	Data Type	Description
Productnm	Varchar(40)	Primary key

Table 6.3: Product Table

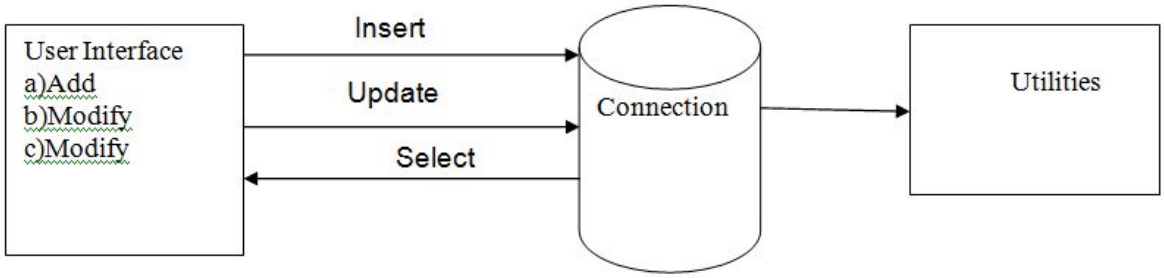


Figure 6.3: Design of Connection Configuration

6.1.6 High Level Design

Figure on next page represents high level design for connection configuration.

6.2 Single Sign On

6.2.1 Objective

Single Sign On is used for user authentication to certain utilities of the portal so that only authorized users can login into the utility and access or modify the data.

6.2.2 Functional Requirements

- The Module Consists of two roles: User and Administrator.
- Administrator should be able to add users and delete users.
- Add user details should contain the following fields...
  - First name
  - Last name
  - Email
  - Password
- Delete User Details: user details should be deleted based on the email id displayed as a list of values.
- User should be able to ...

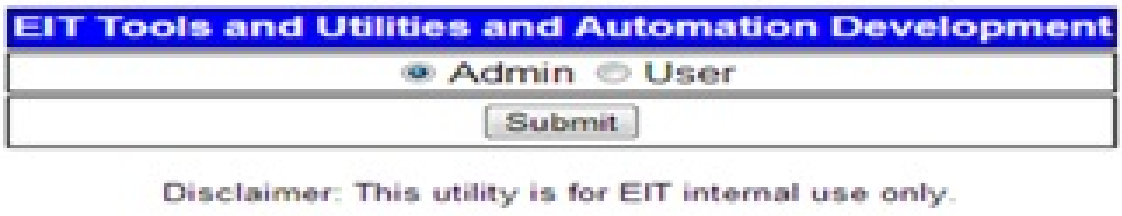


Figure 6.4: User Type Selection

Figure 6.5: Add User

- Change password assigned by administrator.
- Login to the utility using the username and password.

6.2.3 GUI Design

- One can continue as an administrator or user as shown in figure 6.4.
- Admin can add or delete new users based on the email id as shown in figure 6.5.
- User can reset the password that has been issued by the administrator and once the password has been reset, user can login into the portal by providing the username and the respective password.

6.2.4 Database Design

Table Name: User

Field Name	Data Type	Description
Firstname	Varchar(30)	
Lastname	Varchar(30)	
Email	Varchar(50)	Primary Key
Password	Varchar(50)	

Table 6.4: User Table

6.2.5 Implementation Logic

- The utility has been implemented for 2 types of users: Administrator and User.
- Administrator will log into the administrator utility and will be able to add users and also will be able to delete existing or registered users.
- User can change the password that would be preset by the administrator if required and using the valid username and password user can access the utility.

6.2.6 High Level Design

- Administrator

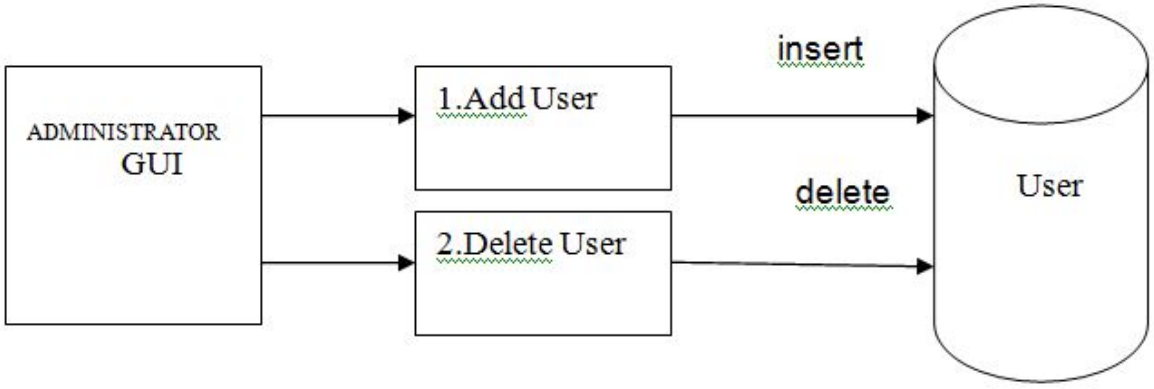


Figure 6.6: Administrator application flow design

- User

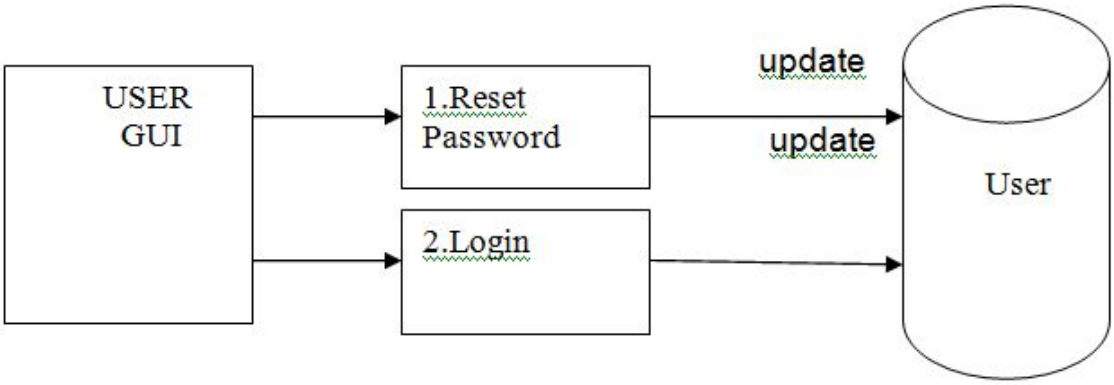


Figure 6.7: User application flow design

6.3 RA Integration

6.3.1 Objective

Main objective of this utility is to generate csv files from templates of different natures. CSV files having bulky data are stored in cloud which is used to populate RA tables whenever required.

6.3.2 Functional Requirements

- Nature of template should be changed by editing keywords in properties file without changing the code.
- All keywords holding data of source file must be unique and related keywords should have common prefix.
- There should be no limit on no. of csv files generated at a time though one should be able to restrict this using properties file.
- Functionality should be implemented to copy data directly from database and also to generate empty csv files whenever required.
- There should be no restriction on depth of dependency levels.
- In case of no primary key or ambiguous primary key, there must be a provision to define pseudo primary key to handle specific group of data.

6.3.3 GUI Design

Refer figure 6.8 and 6.9 given below.

6.3.4 Implementation Logic

1. First, Template needs to be prepared having exactly 1 row.
2. User should fill dynamic values in source file.
3. Proper template names and keywords will be specified in properties file for which csv files are required.

```
W_RTL_PRODUCT_BRAND_DS_LEVEL1=B
W_RTL_PRODUCT_BRAND_DS_LEVEL2=SB
##################################################
Org. Hierarchy
##################################################
Company=1
Chain=11
Area=22
Region=33
District1=444
District2=555
##################################################
Merc. Hierarchy for Fashion Items
##################################################
Company=1
Division=555
Group=555
Department=1112
Class=1112,1113
Subclass=1112,1113
##################################################
Merc. Hierarchy for Grocery Items
##################################################
Company=1
Division=11
Group=111
Department=1111
Class=1111
Subclass=1111
##################################################
Location Details
##################################################
LOC_TYPE|LOC_ID|LOC_STATE|LOC_CL_ID|LOC_CL|BANNERD|CHANEL_ID|CHANNELD|LOC_ZIP|LOC_T
ST|1111|MN|01|Business|EIT_BANNER|1|B & M|55402|Seasonal Pop-UP|1|C|1000|111|MPLS|United States
```

Figure 6.8: Source file

4. As a preconfiguration, Data will be read first from all files and nature of data will also identified.

5. Once data is stored appropriately, dynamic values should be propagated followed by loading dependent values. Static and Fixed values are loaded at last.

6. Code should handle tables which is having more than 256 columns and not contained in template. Such columns should also be loaded if specified in properties file.

7. Finally, csv files should be generated in format as specified in properties file.

```
LEVEL_NAME=dependent=LVL4ANC_PRODCAT_ID:LVL5ANC_PRODCAT_ID:LVL6ANC_PRODCAT_ID:LVL7ANC_PRODCA
END
################################################
tablename=W_PRODUCT_DS
PRODUCT_TYPE_CODE=fixed=ITEM
UOM_CODE=fixed=EA
SRC_EFF_FROM_DT=fixed=todaydate
INTEGRATION_ID=dependent=PROD_NUM,copyvalue
PROD_CAT5=dependent2=Division:Group:dummy3PROD_CAT5:dummy4PROD_CAT5:dummy5PROD_CAT5,valuefrommultiplecolumns
Division=dependent=PROD_NUM,groupfromDV
Group=dependent=PROD_NUM,groupfromGR
dummy3PROD_CAT5=dependent=PROD_NUM,groupfromDP
dummy4PROD_CAT5=dependent=PROD_NUM,groupfromCL
dummy5PROD_CAT5=dependent=PROD_NUM,groupfromSC
PACK_FLG=fixed=N
PROD_CAT5_AS_WAS=dependent3=PROD_CAT5,copyvalue
PROD_NUM=dynamic=ITEM
END
################################################
tablename=W_INT_ORG_ATTR_DS
ORG_NUM=dynamic=Store,PWH,VWH
W_CATEGORY=fixed=RETAIL
ORG_ATTR1_NAME=fixed=Business
ORG_ATTR2_NAME=fixed=01
ORG_ATTR3_NAME=dependent=ORG_NUM,emptyorfixedasPWH:EIT Banner
ORG_ATTR4_NAME=dependent=ORG_NUM,emptyorfixedasPWH:1
ORG_ATTR5_NAME=dependent=ORG_NUM,emptyorfixedasPWH:B & M
ORG_ATTR7_NAME=dependent=ORG_NUM,emptyorfixedasPWH:BANDM
ORG_ATTR8_NAME=dependent=ORG_NUM,emptyorfixedasStore:Y
ORG_ATTR13_NAME=dependent=ORG_NUM,fixedoremptyasStore:9999999999
ORG_ATTR14_NAME=dependent=ORG_NUM,specificreplacement1111::2222::3333::4444::5555::10:-1:20:-1:11:10:12:10:21:20:22:20
ORG_ATTR15_NAME=dependent=ORG_NUM,groupfromZIP
ORG_ATTR19_NAME=dependent=ORG_NUM,firstorsecondasPWH:N:Y
ORG_ATTR21_NAME=dependent=ORG_NUM,mappedvalue2
ORG_ATTR22_NAME=dependent2=ORG_ATTR21_NAME,mappedvalue
```

Figure 6.9: Properties file

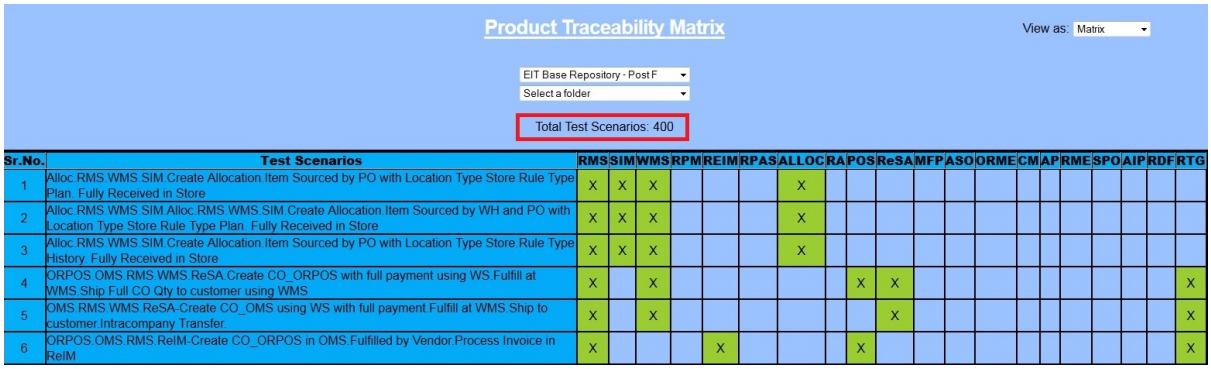
6.4 Product Traceability Matrix

6.4.1 Objective

This utility helps stack holders to identify coverage of test cases as well as products which eventually leads to better test scenarios.

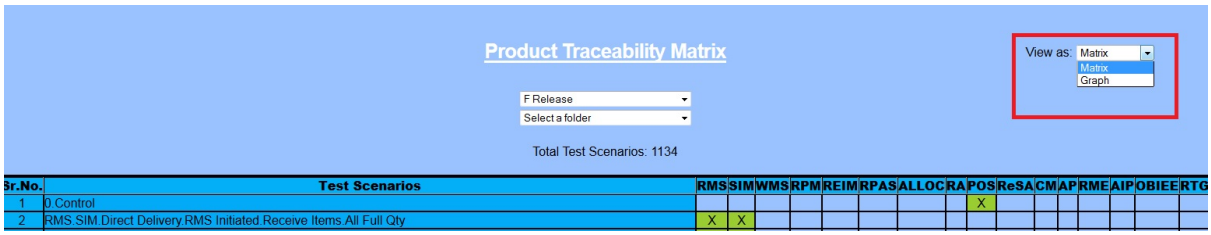
6.4.2 Functional Requirements

- Utility should be developed for HPQC as well as OTM database.
- It should be easy to switch between matrix and graph view.
- Graph should be updated on real time.
- Cache mechanism is preferable to reduce time.
- There should be no restriction on folder level hierarchy.



Sr.No.	Test Scenarios	RMS	SIM	WMS	RPM	REIM	RPAS	ALLOC	RA	POS	ReSA	MFP	ASO	ORME	CM	AP	RME	SPO	AIP	RDF	RTG
1	Alloc.RMS.WMS.SIM.Create Allocation.Item Sourced by PO with Location Type Store.Rule Type Plan..Fully Received in Store	X	X	X				X													
2	Alloc.RMS.WMS.SIM.Alloc.RMS.WMS.SIM.Create Allocation.Item Sourced by WH and PO with Location Type Store.Rule Type Plan..Fully Received in Store	X	X	X				X													
3	Alloc.RMS.WMS.SIM.Create Allocation.Item Sourced by PO with Location Type Store.Rule Type History..Fully Received in Store	X	X	X				X													
4	ORPOS.OMS.RMS.WMS.ReSA.Create CO_ORPOS with full payment using WS.Fulfill at WMS.Ship.Full CO Qty to customer using WMS	X		X						X	X										X
5	OMS.RMS.WMS.ReSA-Create CO_OMS using WS with full payment.Fulfill at WMS.Ship to customer.Intracompany Transfer	X		X							X										X
6	ORPOS.OMS.RMS.ReIM-Create CO_ORPOS in OMS.Fulfilled by Vendor.Process Invoice in ReIM	X				X				X											X

Figure 6.10: Matrix view



Sr.No.	Test Scenarios	RMS	SIM	WMS	RPM	REIM	RPAS	ALLOC	RA	POS	ReSA	CM	AP	RME	AIP	OBIEE	RTG
1	0.Control									X							
2	RMS.SIM.Direct Delivery.RMS Initiated.Receive Items.All Full Qty	X	X														

Figure 6.11: Switching View

- User should be able to save matrix as well as graph.

6.4.3 GUI Design

Figure 6.10, 6.11 and 6.12 given here show GUI of this utility.

6.4.4 Database Design

Refer table 6.5, 6.6 and 6.7 given below.

Field Name	Data Type	Description
AL_ITEM_ID	NUMBER(10)	Primary key
AL_FATHER_ID	NUMBER(10)	Not Null
AL_DESCRIPTION	VARCHAR2(255)	
AL_NO_OF_SONS	NUMBER(10)	

Table 6.5: ALL_LISTS Table

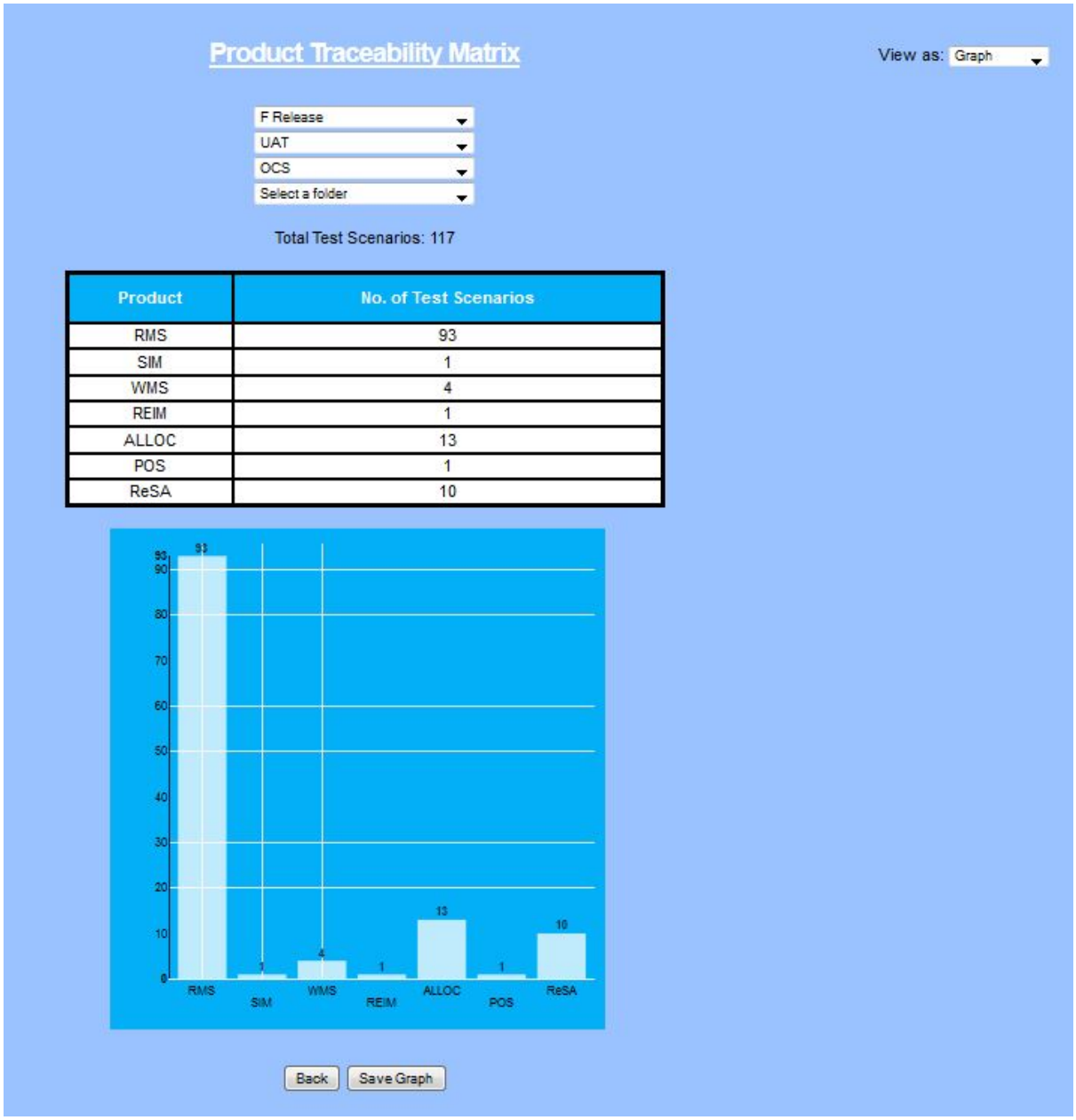


Figure 6.12: Graph View

Field Name	Data Type	Description
TS_TEST_ID	NUMBER(10)	Primary key
TS_NAME	VARCHAR2(255)	
TS_STEPS	NUMBER(5)	Not Null
TS_SUBJECT	NUMBER(10)	
TS_CREATION_DATE	DATE	Not Null

Table 6.6: TEST Table

Field Name	Data Type	Description
TESTCASEID	NUMBER(38)	Primary key
NAME	NVARCHAR2(255)	Not Null
Deleted	NUMBER(3)	Not Null
USERID	NUMBER(10)	
CREATIONDATE	DATE	
PARENTID	NUMBER(10)	
PROJECTID	NUMBER(10)	
TESTTYPEID	NUMBER(10)	
MODIFICATIONDATE	DATE	
DELETIONDATE	DATE	

Table 6.7: TO_TestCase Table

6.4.5 Implementation Logic

1. As a starting point, one root folder is defined for particular database.

2. Irrespective of no. of levels in hierarchy, another select menu should appear till it reaches to folder at last level.

3. One function jots down all sub folders under selected folder where each of folders will be given to another function as an input which does the job of collecting test cases under particular folder.

4. Once all test cases are retrieved, it filters as per different product based on criteria decided with the help of test case description.

5. There is a functionality to download report in excel format.

6. User is allowed to switch between graph and matrix view at any point without disturbing folder selection.

7. Cache mechanism is used to reduce graph retrieval time and cache is refreshed at regular intervals.

8. Graphs are displayed at run time using d3.js. In case, browser does not support d3.js, it is detected by java script and for such cases, graphs will be prepared using Jfree Chart library.

9. Functionality is developed to download graph at client side as well.

6.5 Automation Issue Log

6.5.1 Objective

Automation Issue log is to put the all the issues which everyone is getting while doing automation. So that all the issues will be together and can be tracked and it can also be closed afterwords.

6.5.2 Functional Requirements

This utility should be used by the tester to update any issues faced in automation work. The tester should be able to perform the following tasks...

- Add new issue
- View issue
- Modify issue

Add new issue:Tester should be able to add new issue and it must contain the following fields...

- Application
- Short description
- Long description
- Priority
- Status
- Responsible/accountable

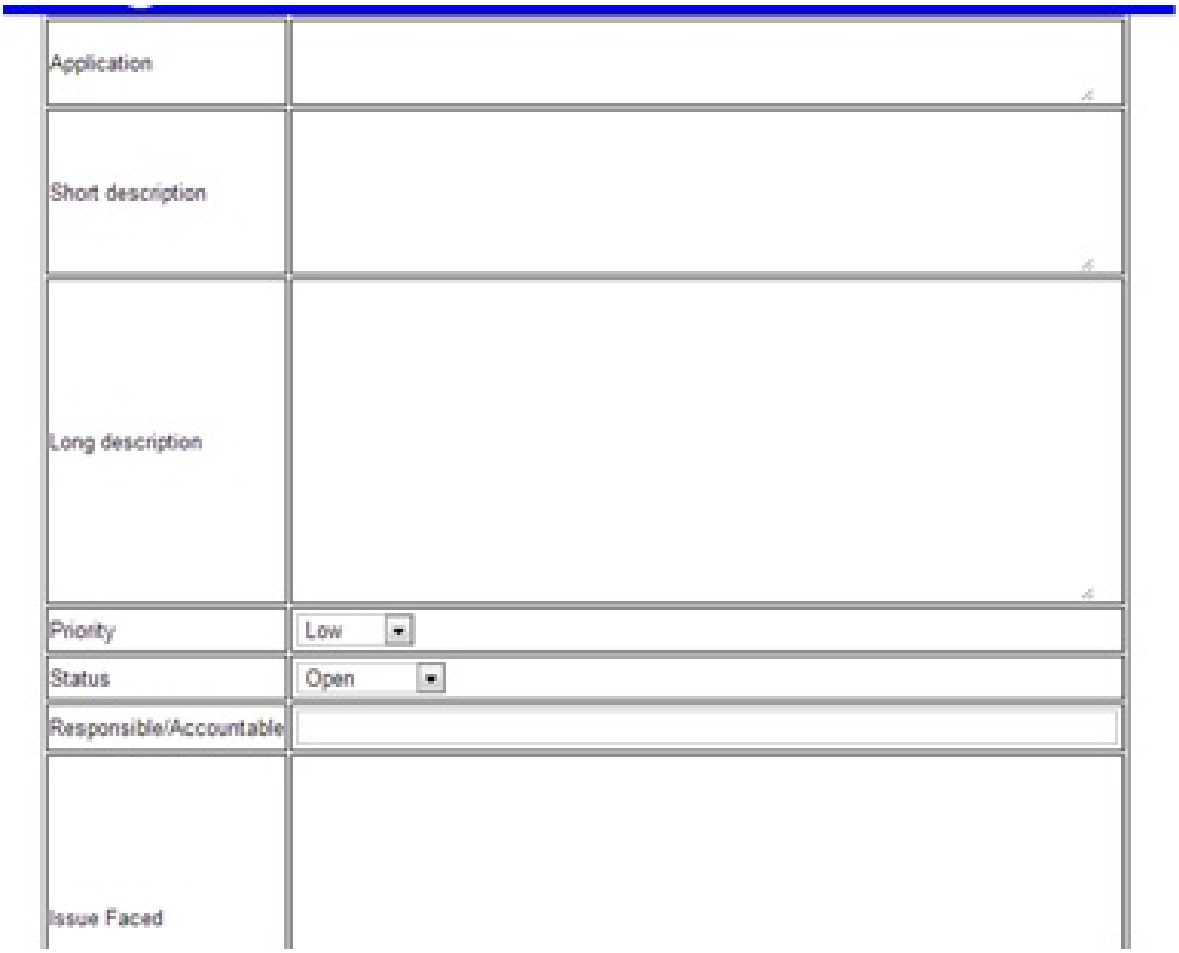


Figure 6.13: Automation Issue Log Front Page

- Issue faced
- Resolution

View issue: Tester should be able to view the issue that has been filed.

Modify issue tester should be able to modify the issue that has been filed.

6.5.3 GUI Design

Refer figure 6.13, 6.14 and 6.15 given here.

6.5.4 High Level Design

Refer figure 6.16 given below.

Application	Short Description	Long Description	Priority	Status	Responsible/Accountab
Synergy	Facing the issue for Logout on RelM as it fails to record the Logout	I reinstalled synergy and restarted several times to allow it to pick updates.but still it fails to recognize RPM.Made a workflow recording for RelM.for the same during playback when tool encounters a drop down,i get an exception exceeded the number of arguments.	High	Open	Mithilesh/Puneet
Automation Utilities	Contact Us page in EIT Page	have to create a page to contact.but I am not able to send the email from that page.I tried a lot to send the email.but I don't know how to setup the server.So I was not able to do that.i tried to set the server but unable to procede.	High	Open	Parag/Krishna
Automation Utilities.	System Option for Allocation.	I cant get the details regarding for database connection for allocation in wiki page. So I was not able to connect to Allocation database. Database for allocation is different then RMS.	High	Open	Parag/Krishna
OATS	OATS recording of RMS and WMS	The work flows in RMS and WMS have to be recorded in OATS.	Low	Open	Jayanth/Krishna

Figure 6.14: Automation Issue Log Result Page

Priority	Low
Status	Open
Responsible/Accountable	Jayanth/Krishna
Issue Faced	I am not able to playback the recorded workflow. Spoke to Ashok from automation team. Re-installed OATS in custom version.
Resolution	

Submit Back

Disclaimer: This utility is for EIT internal use only.

Figure 6.15: Automation Issue Log Modify Page

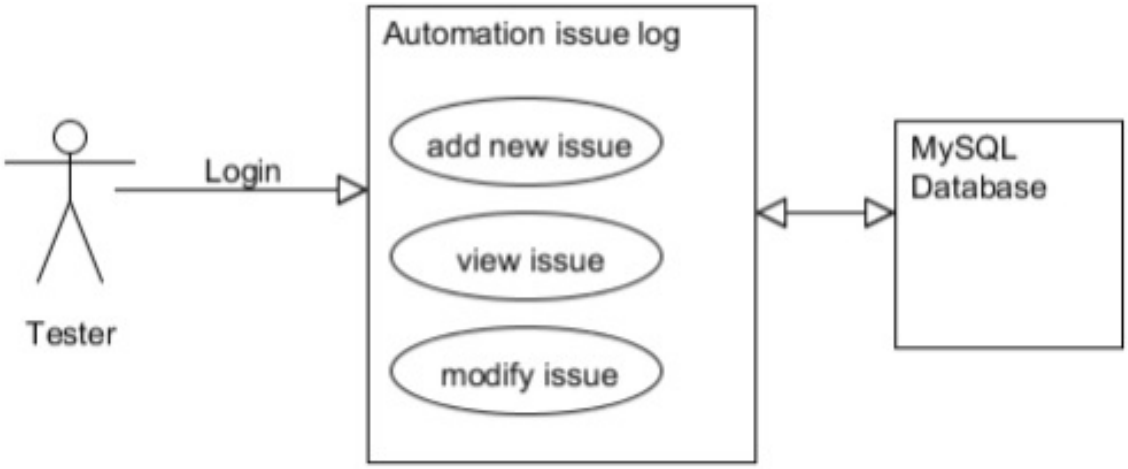


Figure 6.16: Automation Issue Log Design

6.6 RMS Batch Scheduler

6.6.1 Objective

This Utility is used to generate the batch report on daily basis. We are running the batch schedulers on different environments every day. User is able to see the report to check which batches ran successfully and which are failed.

6.6.2 Functional Requirements

- When user will click on RMS batch scheduler, it should display the reports that as to which batches ran what day.
- Daily a file is generated into automation box which shows all the batches which are running and not running and there is a need to track the report of all the days.
- The report needs to be consolidated such that the consolidated reports should be stored and should be accessible through the user interface, so that on one click, user should be able to see the report of all days and see the overall status that report is failed or passed.
- It is required that user can also see which batches got success, which are failed and which did not run.

Archived Env - 14.0 Env – QA6

RMS Batch Scheduler Report - QA7

Current DATE	VDate	dlyprg	salstage	saldly	fifgldn1	fifgldn2	salapnd	stkdly	salweek pre	salweek	salweek post	salmth	salmth post	dtesys	Overall Status
DEC-23-2013	DEC-23-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-23-2013	DEC-23-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-23-2013	DEC-23-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-23-2013	DEC-23-2013	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	Y	Success
DEC-23-2013	DEC-23-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-17-2013	DEC-17-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-16-2013	DEC-16-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-15-2013	DEC-15-2013	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	DNR	DNR	DNR	Y	Success
DEC-14-2013	DEC-14-2013	Y	Y	Y	Y	Y	Y	Y	Y	Y	Y	DNR	DNR	Y	Success
DEC-12-															

Figure 6.17: RMS Batch Scheduler

6.6.3 GUI Design

RMS batch scheduler graph displays percentage of reports that are passed or failed across any environments. The report link directs the user to the consolidated reports of all days and it shows whether reports have passed or failed.

6.6.4 Implementation Logic

- Whenever we click on link of rms batch scheduler, then all the files which are in directory are read and it checks the name in table scheduler_data_file and verifies file entry, if it is there then it will not open the file else it will open the file.
- If file is not read, then it will open the file and we can read the file. Read occurs line by line and lines are divided by comma separated and stored in array. Read of each batch success and failure is converted into a report and make the report entry in an array. The date is taken from the file and the data is entered into oracle scheduler_data table with y or n as a status indicator.
- Thus, all the files are read which are present in the batch scheduler and also making sure that each file has an entry in scheduler report. After making all entries to oracle database. We display the table of oracle database as a report.

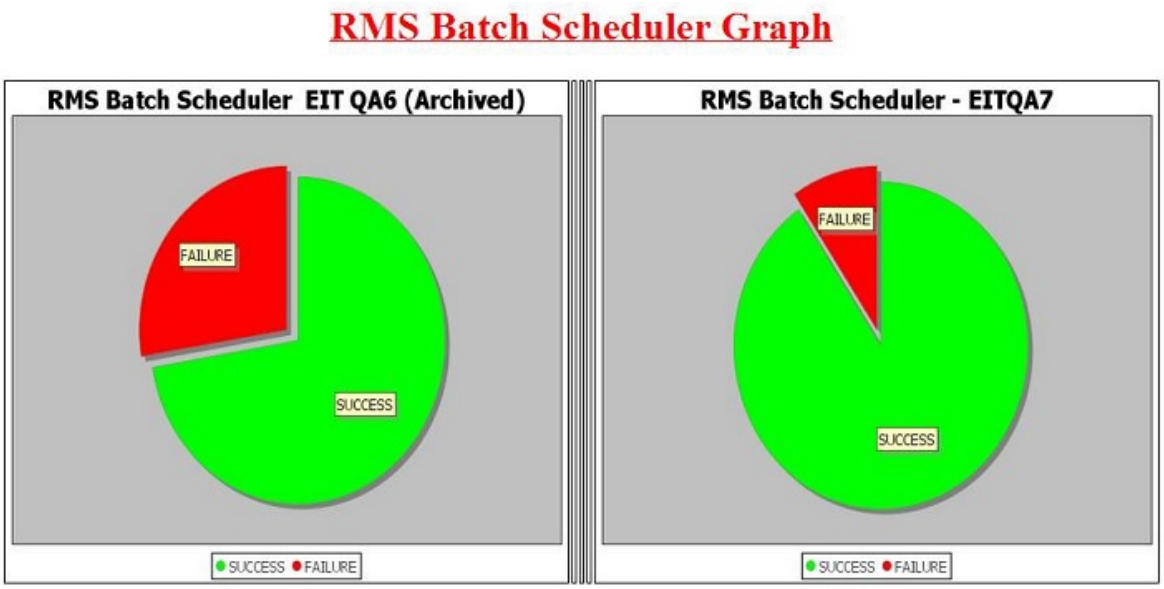


Figure 6.18: RMS Batch Scheduler Graph

6.7 Message Coverage

6.7.1 Objective

As a part of EIT testing team, we will test multiple applications which includes RIB , Interfaces and Batch Programs. So to keep track this utility,we will give a snapshot of what are all artifacts covered as a part of our testing and also it provides the user (admin) to update the Coverage details. It also provide a GUI (graphs) on percentage of coverage.

6.7.2 Functional Requirements

The section is divided into four parts...

- 'F' coverage report: This report shows the report for overall coverage that contains product coverage and interface coverage. It shows graph of overall coverage and interface coverage.
- Interfaces:
  - Covered: This shows that what all messages have been covered in table format.
  - Not Covered: This shows that what all messages has not been covered in table format.

  - All: This shows that all messages which are covered or not covered in table format.

- Administration: This is admin page. It asks for admin login and then admin can change message coverage from coverage to non coverage or vice versa.

- Advanced search: This gives option to display result based on following criteria:

  - entity: you have to select the entity and it give result of all the data of the entity.

  - entity and subentity: you have to select the entity and sub entity and it give result of all the data of the entity and sub entity.

  - source and destination: you have to select the source and destination and it give result of all source and destination.

  - interface: you have to select the interface and it gives result of all the data of the interface.

6.7.3 GUI Design

The utility should contain the following sub parameters as the front page of the utility. The links leads to the respective reports that are provided in the functional requirements. Refer figure 6.19.

6.8 Business Transaction Validation

6.8.1 Functional Requirements

- User should be able to select the following...

  - Test Environment

  - Entity

  - Sub Entity

  - Choice for selection of entity ID

    * Pick form List

    * Enter Value

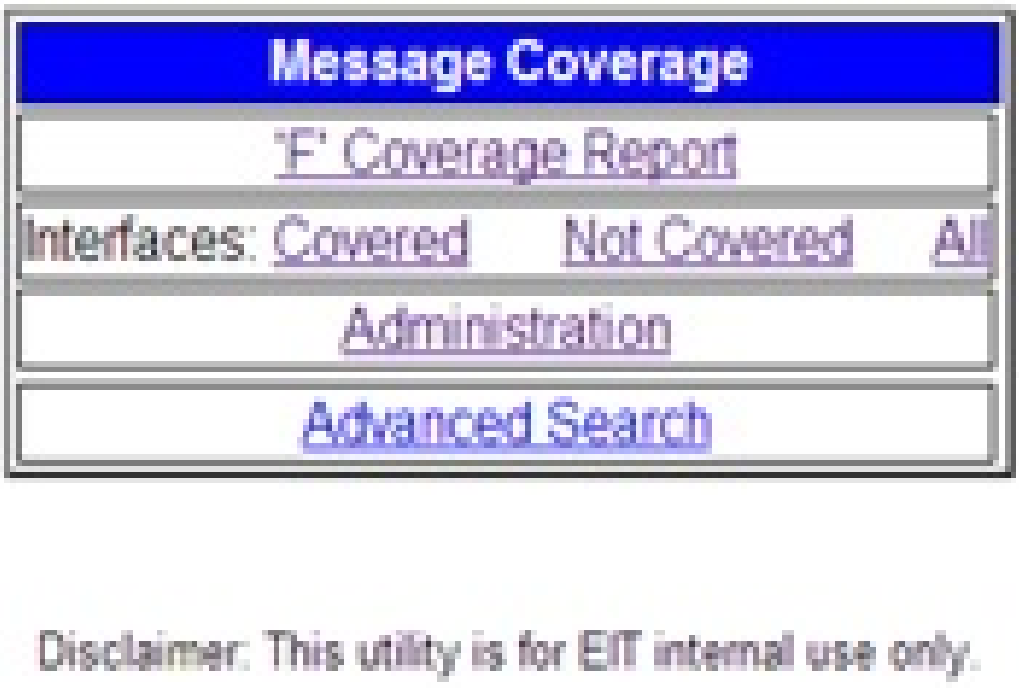


Figure 6.19: Message Coverage Home Page

- User should be able to give the entity ID either by selecting from a list or by entering manually.
- User must be given access to a flat file containing the list of databases, tables and respective queries for validation of the selected entity.
- Any modification of the flat file must be immediately reflected in the result page.
- Once the user submits the entity ID, the result page should validate the business transaction across the databases and the tables specified in the flat file across the entity.

6.8.2 GUI Design

Spread Sheet

An excel sheet is maintained to store the respective queries for any entity. The format is given below.

Home Page

Refer figure 6.21

	A	B	C	D	E
1	ENTITY	SUB ENTITY	DATABASE	TABLE	QUERY
2	PO	PO for SIM	Entity Selection Query	Select Purchase Order Number	select DISTINCT ORDER_NO as NUM from ORDLOC where LOC_TYPE = 'S' ORDER BY ORDER_NO
3	PO	PO for SIM	RMS	ORDHEAD	select * from ORDHEAD where ORDER_NO = '"+id_num+"'
4	PO	PO for SIM	RMS	ORDLOC	select * from ORDLOC where ORDER_NO = '"+id_num+"'
5	PO	PO for SIM	RMS	ORDSKU	select * from ORDSKU where ORDER_NO = '"+id_num+"'
6	PO	PO for SIM	RMS	SHIPMENT	select * from SHIPMENT where ORDER_NO = '"+id_num+"'
7	PO	PO for SIM	RMS	SHIPSKU	select * from SHIPSKU where SHIPMENT in(select SHIPMENT from SHIPMENT where ORDER_NO = '"+id_num+"')
8	PO	PO for SIM	RMS	ITEM_LOC_SOH	select * from ITEM_LOC_SOH where ITEM in (select ITEM from ORDLOC where ORDER_NO = '"+id_num+"')and LOC in(select LOCATION from ORDLOC where ORDER_NO='"+id_num+"')
					SELECT * FROM STORE_ITEM_STOCK WHERE ITEM_ID IN (SELECT ITEM_ID FROM purchase_order_line_item WHERE purchase_order_ID in (SELECT ID FROM purchase_order WHERE external_id IN (SELECT RMS01.ORDLOC.ORDER_NO FROM

Figure 6.20: Excel Sheet with queries for Business Transaction Validation

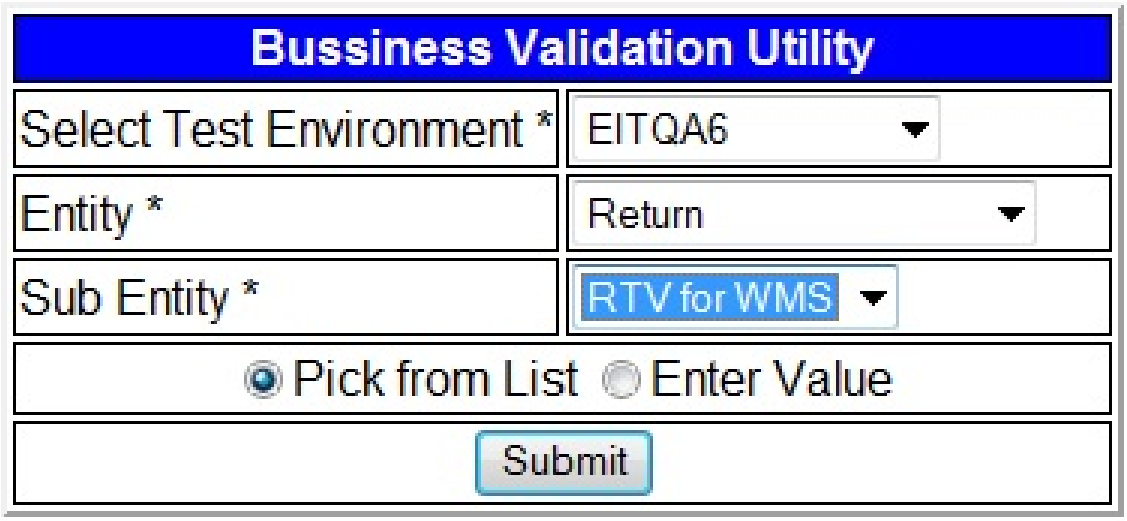


Figure 6.21: Home Page for Business Transaction Validation

Entity Selection Page

Refer figure 6.22

Result Page

Refer figure 6.23

6.8.3 Implementation Logic

Home Page

The first column of the spreadsheet are entities which are repeated. In a single iteration over the spreadsheet the entities are fetched and stored in a Set. Since a Set does not store repeated values, it contains the unique entities.

In the second iteration, for every entity, a Set is prepared with the sub-entities and later an entry is made in the Hash table with the key as entity and the value as the Set of

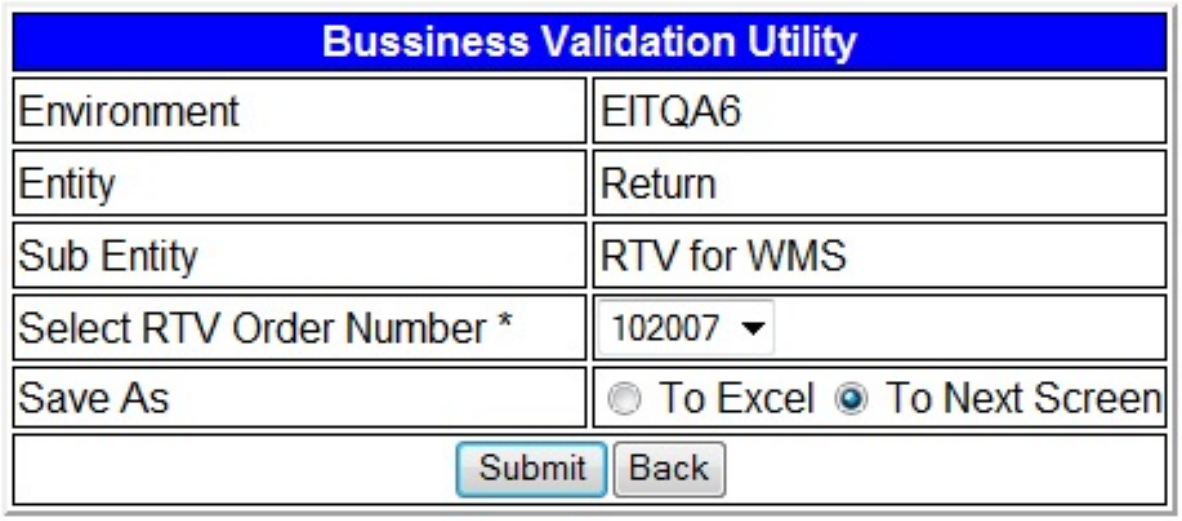


Figure 6.22: Entity Selection Page for Business Transaction Validation

Return
RTV for WMS

RMS DATABASE

ITEM_LOC_SOH

ITEM	ITEM_PARENT	ITEM_GRANDPARENT	LOC	LOC_TYPE	AV_COST	UNIT_COST	STOCK
100016296	null	null	4241	S	10	10	
100016296	null	null	20	W	10	10	
100016296	null	null	123	S	10	10	
100016296	null	null	19	W	10	10	
100016296	null	null	22	W	10	10	

RTV_HEAD

RTV_ORDER_NO	SUPPLIER	STATUS_IND	STORE	WH	TOTAL_ORDER_AMT	SHIP_TO_ADD_1	SHIP
102007	11	10	1	22	50	900 2nd ave s	

RTV_DETAIL

RTV_ORDER_NO	SEQ_NO	ITEM	SHIPMENT	INV_STATUS	QTY_REQUESTED	QTY_RETURNED	QTY_C
102007	1	100016296	null	null	5	null	

TRAN_DATA

Figure 6.23: Business Transaction Validation Result

```
Algorithm 1
  while entity set has an entity do
    for each row in the spreadsheet do
      if entity from set equals entity from sheet then
        if !sub_entity_set.contains(sub_entity) then
          sub_entity_set.add(sub_entity);
          entity_table.put(entity, sub_entity_set);
        end if
      end if
    end for
  end while
```

Figure 6.24: Business Transaction Validation: Home Page Algorithm

```
Algorithm 2
  if entity from set equals entity from sheet then
    if Sub entity from set equals Sub entity from sheet then
      if Third column contains "Entity Selection Query" then
        Execute the Query in the QUERY column.
      end if
    end if
  end if
```

Figure 6.25: Business Transaction Validation: Selection Page Algorithm

sub-entities.

Hence a Hash table is obtained with entities as keys and values as sets of corresponding sub-entities. The procedure is done in O(n) time complexity during worst case scenario and O(1) in the best case scenario; where n is the number of entries in the spreadsheet.

Entity Selection Page

For every sub-entity, there is an entry in the spreadsheet with Entity Selection Query in the third column. The corresponding query is then fetched, executed and the result is displayed in the dropdown box for the user to select an entity.

The procedure is done in O(n) time complexity during worst case scenario and O(1) in

the best case scenario; where n is the number of entries in the spreadsheet.

Result Page

Selected entity is searched in the first column of the spreadsheet, once found the sub-entity is searched in the second column of that particular entity only. In the spreadsheet the third column contains the database name, fourth column contains the table name and the fifth column contains the query to be executed. Hence the query is executed based on the database name.

The display of the result for every table is performed using **print_table** method of **Entity_Details** class.

Hence, the overall process is done in O(n) time complexity during worst case scenario and O(1) in the best case scenario; where n is the number of entries in the spreadsheet.

6.8.4 Guidelines to Handle Spreadsheet

(1) Understand the format precisely.

(2) Do not change the format of the excel sheet.

(3) Do not modify the existing names of entities, sub-entities and databases. They are case sensitive.

(4) Do not use any unnecessary spaces before or after an entry in the sheet.

(5) Do not use any extra space unless mandatory within the queries.

(6) All the entries for a entity must be placed together.

(7) Do not add data unless its not already present in the spreadsheet. Please reuse[by Copying] all possible columns for new entries.

(8) All the entries must be made in the same format and in the same worksheet.

(9) Every single extra space in any query will throw an exception.

6.8.5 Architecture

Figure 6.26 describes architecture for Business Transaction Validation.

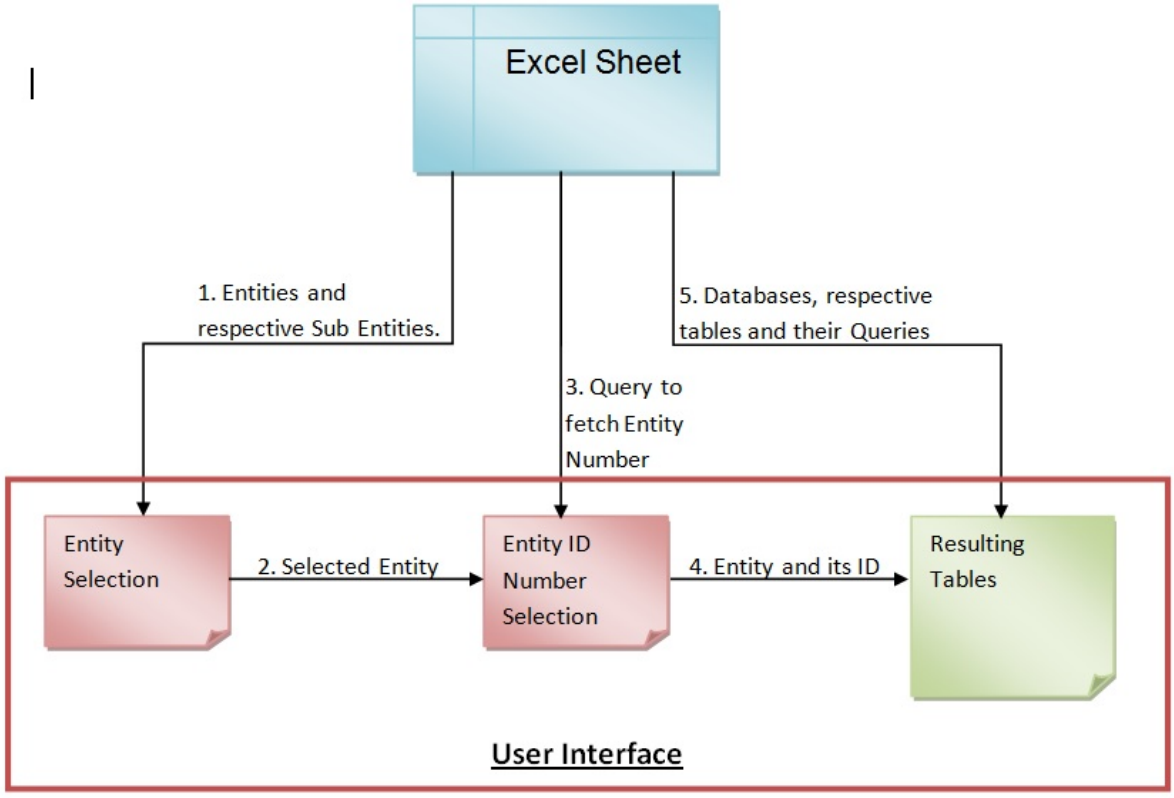


Figure 6.26: Business Transaction Validation Architecture

Chapter 7

Conclusion

7.1 Automation

7.1.1 Advantages of Automation

- Reliable: eliminating human error.
- Reusable.
- Better Quality Software.
- Fast.
- Cost Reduction.

7.1.2 Disadvantages of Automation

- High investment is needed in the tools and training.
- High man power requirement for test preparations.
- A lot of testing areas left uncovered.

7.2 Final words

- Automation is an ongoing commitment, not a one-time step.
- Automation is coding, not testing and requires specialized skills.
- Automation has no short term benefit, and instead incurs an immediate cost in terms of resources, time, and money.

References

[1] Elfriede Dustin. *Implementing Automated Software Testing.* Addison-Wesely, 2009.

[2] Oracle Retail. Oracle functional testing, 2010.

[3] Oracle Retail. Oracle retial merchandising system, 2012.

[4] Wikipedia. Test automation — wikipedia, the free encyclopedia, 2014. [Online; accessed 12-September-2014].