Database Editor

Test plan

Version 1.0

04/06/2020

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 04/05/2020 |
| Current Release: | 04/05/2020 |
| Indicator of Last Page in Document: |  |
| Date of Last Review: |  |
| Date of Next Review: |  |
| Target Date for Next Update: | 05/10/2020 |

Distribution List

This following list of people shall receive a copy of this document every time a new version of this document becomes available:

Guidance Team Members:

Customer:

Software Team Members:

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 04/05/2020 | Carlos Vargas | Test plan first release |
|  |  |  |  |
|  |  |  |  |

Note: The template presented in this document was taken from:

Donaldson, S., and S. Siegel, *Successful Software Development*. Upper Saddle River, NJ: Prentice Hall, 2001, pp. 321-323.

Note: The template presented in this document was taken from: Donaldson, S., and S. Siegel, *Successful Software Development*. Upper Saddle River, NJ: Prentice Hall, 2001, pp. 321-323 and modified by Humberto Mendoza and Steve Roach.

Supplementary information is from:

Pfleeger, S. *Software Engineering, Theory and Practice*. Upper Saddle River, NJ: Prentice Hall, 1998, p. 365.

Table of Contents

[Document Control ii](#_Toc22915465)

[Approval ii](#_Toc22915466)

[Document Change Control ii](#_Toc22915467)

[Distribution List ii](#_Toc22915468)

[Change Summary ii](#_Toc22915469)

[1. Introduction 1](#_Toc22915470)

[1.1. Purpose 1](#_Toc22915471)

[1.2. Scope 1](#_Toc22915472)

[1.3. System Overview 1](#_Toc22915473)

[1.4. Suspension and Exit Criteria 1](#_Toc22915474)

[1.5. Document Overview 1](#_Toc22915475)

[1.6. References 1](#_Toc22915476)

[2. Test Items and Features 2](#_Toc22915477)

[3. Testing Approach 3](#_Toc22915478)

[4. Test SUITE A 4](#_Toc22915479)

[4.1. Test <<test id>> 4](#_Toc22915480)

[5. User Interface Testing 5](#_Toc22915481)

[6. Test Schedule 6](#_Toc22915482)

[7. Other Sections 7](#_Toc22915483)

[8. Appendix 8](#_Toc22915484)

# Introduction

This test plan is intended to cover some functionality of a basic table editor application. This plan contain detailed steps that must be followed in order to test the given application.

## Purpose

This test plan would ensure that the given application works properly or if it has to be modified in case some errors are found. By running the specified tests, bugs could be identified for future fixes.

## Scope

The test covers the functionality for comparing different versions of the same table, and the verification of duplicated data in different rows in a table.

## System Overview

This is a database table editor. It contains some basic functionality. It modifies the content in an XML file by saving the file, and it creates a file for the previous version of the table. Rows and columns can be duplicated, removed, inserted.

## Suspension and Exit Criteria

All test must be performed before stopping.

## Document Overview

The document contains the test suit with the tests to be ran.

## References

<https://piazza.com/class_profile/get_resource/k55rbh37ri6be/k8hjj7sn7hu7n0>

# Test Items and Features

This test plan covers the following features of the database editor:

1. Compare file to other version feature
2. Check for duplicate rows feature.

# Testing Approach

For performing tests, the application must be launched before. It also has to load a database with tables. The setup instructions are provided in next section.

Table 1: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE A** | | |
| **Description of Test Suite** | Test that focuses to the previously mentioned features: comparison of table versions and checking for duplicated rows. | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| 1 | First test for the feature that compares version. In this test the same table is compared with the exact same version of table 1. |  |
| 2 | In this test, the comparison is between 2 different tables. |  |
| 3 | In this test, a change in table is compared with the same table with no changes to get a mismatch. |  |
| 4 | This test uses the column selection feature to disable a column and change the content of this column to compare the same table. |  |
| 5 | This test checks for duplicate rows by using the feature and finds matches. |  |
| 6 | This test checks for duplicate rows by using the feature and does not find matches |  |
| 7 | This test checks for duplicate rows by using the feature and it also uses the column selection feature. |  |

# Test Suite A

## Initial Set Up

1. Launch dbEdit.jar file. This is the initial view of the application:

![Una captura de pantalla de una computadora

Descripción generada automáticamente]()

1. Click on the upper-left menu “File”, select “Open” and finally choose the main DB file named “Test\_DB.xml”

![Captura de pantalla de un celular con letras

Descripción generada automáticamente]()



1. The next menu illustrates the three available tables. Select any of those three.

![Imagen que contiene captura de pantalla

Descripción generada automáticamente]()

1. Tables are represented as shown in image below.

![Captura de pantalla de un celular con letras

Descripción generada automáticamente]()

## Test 1

**Objective:** Compare table 1 with exact same table.

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 1 | | | | Current Status: Pending | | |
| Test title: Compare table to exact same table. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to be compared with itself. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened | COMMENTS |
| 2 | Click on the File menu item. And then “Compare table to another version” | Opening folder with databases files. | | | Opening the file system with current directory. |  |
| 3 | Select the CONSTRAINT\_TABLE (XML file) to be compared with table 1 (opened table). | Compare the same table. | | | 0 mismatches: |  |
| Concluding Remarks: Test is successful if there are no mismatches. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

## Test 2

**Objective:** In this test, the comparison is between 2 different tables.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 2 | | | | Current Status: Pending | | |
| Test title: Compare table to different table. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to be compared with TYPE\_TABLE. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened | COMMENTS |
| 2 | Click on the File menu item. And then “Compare table to another version” | Opening folder with databases files. | | | Opening the file system with current directory. |  |
| 3 | Select the TYPE TABLE (XML file) to be compared with table 1 (opened table). | Compare the same table. | | | Any column found in other table: |  |
| Concluding Remarks:  Test successful if the error pops up. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

**Notes:** N/A

## Test 3

**Objective:** Compare table 1 with exact same table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 3 | | | | Current Status: Pending | | |
| Test title: A change in table is compared with the same table with no changes to get a mismatch. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to be compared with itself. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened | COMMENTS |
| 2 | Modify one cell in one column | The comparison has to identify this change | | | A cell is modified |  |
| 3 | Click on the File menu item. And then “Compare table to another version” | Opening folder with databases files | | | Opening the file system with current directory |  |
| 4 | Select the CONSTRAINT\_TABLE (XML file) to be compared with table 1 (opened table) | Compare the same table | | | 1 mismatch: |  |
| Concluding Remarks: Test should be considered successful if different changes are tested. Test should be performed with more than 1 mismatch. This feature does not work with many changes. It should be identified the “max amount” of mismatches that this application can find. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

**Notes:** N/A

## Test 4

**Objective:** Compare table 1 with exact same table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.:4 | | | | Current Status: Pending | | |
| Test title: This test uses the column selection feature to disable a column and change the content of this column to compare the same table. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to be compared with itself. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened | COMMENTS |
| 2 | Modify first cell in column “DESCRIPTION”.  The content should not be “1”. | The cell should not be counted in the comparison. | | | Cell is modified. |  |
| 3 | Disable column “DESCRIPTION” by clicking on the options menu time -> Column selection. Then disable the description checkbox. | Disable one column for comparison. | | | The column is disabled. |  |
| 4 | Click on the File menu item. And then “Compare table to another version” | Opening folder with databases files. | | | Opening the file system with current directory |  |
| 5 | Select the CONSTRAINT\_TABLE (XML file) to be compared with table 1 (opened table). | Compare the same table. | | | 0 mismatches:  Captura de pantalla de un celular  Descripción generada automáticamente | Even though the change is made, the comparison did not include the disabled column. |
| Concluding Remarks: | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

**Notes:**

## Test 5

**Objective:** Compare table 1 with exact same table.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 5 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows by using the feature and finds matches. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to search for duplicate rows. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened. | COMMENTS |
| 2 | Click on the File menu item and then select the option “Check For Duplicate Rows”. | Check for rows. | | | 3 Duplicate rows: | 3 duplicate rows are found. There are 3 rows with same information on the bottom and 2 with the same information on the top. This is why we get 3 duplicates |
| Concluding Remarks:  Test successfully passes if 3 is the result for duplicate rows. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

**Notes:**

## Test 6

**Objective:** Compare table 1 with exact same table.

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 6 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows by using the feature and does not finds matches. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to search for duplicate rows in a table with no duplicates. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened. | COMMENTS |
| 2 | Modify rows so at least one column is different. | To expect 0 as the final result. | | | Every row is different: |  |
| 3 | Click on the File menu item and then select the option “Check For Duplicate Rows”. | Check for rows. | | | 0 Duplicate rows: |  |
| Concluding Remarks:  Test successfully passes if 0 is the result for duplicate rows. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

## Test 7

**Objective:** Compare table 1 with exact same table.

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: 7 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows by using the feature and finds matches. Use feature “Column selection” to get more matches. | | | | | | |
| Testing approach: Select CONSTRAINT\_TABLE to search for duplicate rows. | | | | | | |
| STEP  1 | OPERATOR ACTION  Open CONSTRAINT\_TABLE from main menu after loading TEST\_DB.xml | PURPOSE  Selecting table to compare | | | EXPECTED RESULTS  The table should be opened. | COMMENTS |
| 2 | Disable column “RESTRINCTION\_TYPE” by clicking on the options menu time -> Column selection. Then disable the description checkbox. | Remove a column, to find more duplicates in | | | Captura de pantalla de un celular  Descripción generada automáticamente |  |
| 3 | Click on the File menu item and then select the option “Check For Duplicate Rows”. | Check for rows. | | | 4 Duplicate rows: | 4 duplicate rows are found. There are 4 rows with same information on the bottom and 3 with the same information on the top (after removing one of the columns). This is why we get 3 duplicates. |
| Concluding Remarks:  Test successfully passes if 4 is the result for duplicate rows. | | | | | | |
| Testing Team:  N/A | | | Date Completed: | | | |

# User Interface Testing

N/A

# Test Schedule

<< Specify the schedule for testing activities. A table with the order and completion dates of the tests is useful. The table below might be useful.>>

|  |  |  |
| --- | --- | --- |
| **Task and date** | **People** | **Description** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# Other Sections

<< Other sections that may appear in a test plan (but not required for this course) are:

* Test Management Requirements: how testing is to be managed; a delineation of responsibilities of each project organization involved with testing
* Staffing and training needs: delineate the responsibilities of those individuals who are to perform the testing, level of skill required, and training to be provided
* Environmental Requirements: describe the hardware (including communication and network equipment) needed to support testing; describe configuration of hardware components on which software and database to be tested are to operate.
* Software Requirements: describe the software needed to support testing; include the software code and databases that are object of the testing. Also include software tools such as compilers, CASE instruments and simulators that are needed to model the user’s operational environment.
* Risk and contingencies
* Cost: include an estimate of costs.
* Approvals
* Test Deliverables

>>

# Appendix

<< possibly more readable to put the expected output here and refer to it in the previous sections. Might also provide explicit directions for analysis of output, if it’s easier to read as an appendix or if analysis is post execution. >>

