Database editor

**Test plan**

**Version 1.**3

04/19/2020

**Document Control**

**Approval**

The Guidance Team and the reviewers shall approve this document.

**Document Change Control**

|  |  |
| --- | --- |
| Initial Release: | 04/05/2020 |
| Current Release: | 04/14/2020 |
| Indicator of Last Page in Document: | $ |
| Date of Last Review: | 04/19/2020 |
| Date of Next Review: |  |
| Target Date for Next Update: |  |

**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: Dr. Steve Roach

Producer: Ricardo Sanchez

Reviewers: Sebastian Nunez

Brian Cardiel

**Change Summary**

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 04/14/2020 | Ricardo Sanchez | Created Peer-Unrevised Version |
| 1.1 | 04/17/2020 | Brian Cardiel | Finished 1st Review |
| 1.2 | 04/19/2020 | Sebastian Nunez | Finished 1st Review |
| 1.3 | 04/19/2020 | Ricardo Sanchez | Created peer reviewed version |

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 I**](#_heading=h.30j0zll)

**Approval i**

**Document Change Control i**

**Distribution List i**

**Change Summary i**

[**1.**](#_heading=h.3dy6vkm) **INTRODUCTION 1**

**1.1.** **Purpose 1**

**1.2.** **Scope 1**

**1.3.** **System Overview 1**

**1.4.** **Suspension and Exit Criteri a 1**

**1.5.** **Document Overview 1**

**1.6.** **References 1**

[**2.**](#_heading=h.qsh70q) **TEST ITEMS AND FEATURES 2**

[**3.**](#_heading=h.35nkun2) **TESTING APPROACH 2**

[**4.**](#_heading=h.44sinio) **TEST G4 3**

**4.1.** **Test G4.C1 3**

**4.2.** **Test G4.C2 4**

**4.3. Test G4.D1 5**

**4.4. Test G4.D2 6**

**4.5. Test G4.D3 7**

[**6.**](#_heading=h.3j2qqm3) **TEST SCHEDULE 8**

[**8.**](#_heading=h.1pxezwc) **APPENDIX 8**

# 1. **Introduction**

The following section will introduce detailed information regarding the system and the test plan.

## **Purpose**

This purpose of the test plan is to provide documentation regarding the test items/features, testing approaches, and test schedule. This test plan is one of four different test plans for a simple database text editor. This test plan is a unit test plan that focuses on one of the four modules of the database text editor, which will be described in detail in the sections below.

## **Scope**

The software has a single version with no updates in the foreseeing future. This was provided by Dr. Roach and will be used by all the producers of the different modules of the system mentioned before.

## **System Overview**

The database text editor system is composed of a user interface and four different functional modules:

1. The first module contains file menu operations excluding comparing files to other versions, checking for duplicate rows, and filtering data. This includes the search menu option on the database window.
2. The second module contains filter and the search functions in the edit menu.
3. The third module contains edit menu operations.
4. The fourth module contains the functions: compare file to other version and check for duplicate rows.

The test plan will cover unit testing regarding the fourth module.

## **Suspension and Exit Criteria**

The suspension criteria of the test are the following: all critical tests must satisfy the testing specifications in order to suspend testing. Non-critical tests may or may not pass in order to suspend testing. Lastly each test must be executed carefully following the steps outlined.

## **Document Overview**

The test plan sections are outlined below:

Section 1 introduces the test plan and the system (modules)

Section 2 describes in detail the test items of the system

Section 3 contains the testing approach and a test suite table with description of test cases

Section 4 contains specific test procedures for test cases outlined in section 3

Section 5 contains logs the test schedule and responsibilities

Section 6 contains the appendix, used to illustrate test case outcomes

## **References**

<<List all the references applicable to the test plan. Generally, this includes project standards, SRS, SDD, and a product assurance plan.>>

TBD

# **Test Items and Features**

The items to be tested are the following functions of the dbEdit program:

* The File/Compare option allows a user to compare rows within two different versions of a database table. This compare can be restricted to selected columns by using the Options/Column Selection feature. The output is rather crude and is not intended for large change lists.
* File/Check for Duplicate Rows should list rows that have identical contents. This check is restricted to the columns selected in Options/Select Columns.

# **Testing Approach**

The following section describes the purpose of the tests to be carried out. These tests are designed to test program functions in sequences that approximate operational use of the system. Section 4 contains the steps and a more detailed view of each.

**Table 1: Test Plan**

|  |  |  |
| --- | --- | --- |
| **TEST SUITE <Identifier>** | | |
| **Description of Test Suite** | The test suite will focus on testing the ‘compare file to other version’ and ‘check for duplicate rows’ functions of the database table editor. | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| G4.C1 | Compare 2 different versions of a table | Critical |
| G4.C2 | Compare 2 different tables | Critical |
| G4.D1 | Check for duplicate rows in an empty table | Critical |
| G4.D2 | Check for duplicate rows in a table | Critical |
| G4.D3 | Check for duplicate rows in a table (special case) | Normal |

# **Test** G4

This section describes in detail the test methods, procedures and outcomes in specific labeled steps/

## **Test** G4.C1

**Objective:** The objective of this test case is to compare two different versions of the same table and confirm whether or not the program compares and displays the differences of the right fields.

**Notes:** This test case should take less than 5 minutes to successfully execute. The resources needed to successfully run this test are the dbEdit.jar executable, and the input files: G4\_\_TEST\_TABLE.XML, G4\_\_TEST\_TABLE~.XML provided in the G4\_TestFiles folder provided over on the Group 6 Team 4 GitHub repository. This test case tests 2 versions are of the same table, with all columns fields selected (that is, without manually selecting any specific columns on the options menu tab, also named the default option).

Note that the two files: G4\_\_TEST\_TABLE.XML and G4\_\_TEST\_TABLE~.XML are two different files. Also, the screenshots provided below were resized to fit in the table, actual results might vary on size but not on content. E.g. the name of the window, and the actual content inside of the window should be the same as below.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: G4.C1 | | | | Current Status: Pending | | |
| Test title: Compare 2 different table versions | | | | | | |
| Testing approach: This test will be conducted using the dbEdit.jar file on either windows or macOS. Behavior is observed on the simulation screen once the program is running. The sample screenshots on the expected results column should match after performing the corresponding operator action. | | | | | | |
| STEP | OPERATOR ACTION | PURPOSE | | | EXPECTED RESULTS | COMMENTS |
| 1 | Run the dbEdit executable  (right click the dbEdit file and click Open). | Initial condition | | |  |  |
| 2 | Click “File” on the menu bar, then click the “Open” option.  After the file search path window opens, select and open the file: G4\_\_TEST\_TABLE.XML | Open the newest version of the table. | | |  | Output: A window with the newest version named: G4\_TEST\_TABLE.XML  \*Ignore the red highlighted field |
| 3 | Click “File” on the menu bar, then click the “Compare table to another version” option.  After the file search path window opens, select the file: G4\_\_TEST\_TABLE~.XML | Compare the old version (G4\_\_TEST\_TABLE~.XML) to the newest one (G4\_\_TEST\_TABLE.XML) | | |  | First output: the Compare Result window.  Second: a window with the old version named: G4\_TEST\_TABLE~.XML  Third: a window listing the differences of the two files. |
| Concluding Remarks: | | | | | | |
| Testing Team:  Producer: Ricardo Sanchez  Evaluators: Sebastian Nunez, Brian Cardiel | | | Date Completed: TBD | | | |

## Test G4.C2

**Objective:** The objective of this test case is to compare two different table files and observe how the system handles this operation.

**Notes:** This test case should take less than 5 minutes to successfully execute. This test case compares 2 different tables, all columns fields are used to compare. The key element to note is how both tables share only 1 common column field.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: G4.C2 | | | | Current Status: Pending | | |
| Test title: Compare 2 different tables (that is with completely different column fields), with only 1 column in common (the selected column field appears in both tables). | | | | | | |
| Testing approach: This test will be conducted using the dbEdit.jar file on either windows or macOS. Behavior is observed on the simulation screen once the program is running. The sample screenshots on the expected results column should match after performing the corresponding operator action. | | | | | | |
| STEP | OPERATOR ACTION | PURPOSE | | | EXPECTED RESULTS | COMMENTS |
| 1 | Run the dbEdit executable (right click the dbEdit file and click Open). | Initial condition | | |  |  |
| 2 | Click “File” on the menu bar, then click the “Open” option.  After the file search path window opens, select and open the file: G4\_\_TEST\_TABLE.XML | Open the first of 2 tables to compare. | | |  | Output: A window with the newest version named: G4\_TEST\_TABLE.XML  \*Ignore the red highlighted field. |
| 3 | Click “File” on the menu bar, then click the “Compare table to another version” option.  After the file search path window opens, select the file: G4\_\_CONSTRAINT\_TABLE.XML | Compare: G4\_\_TEST\_TABLE.XML to a completely different format of table: G4\_\_CONSTRAINT\_TABLE.XML | | |  | First output: a window with the columns that are not contained in both tables.  Second: window with the number of total mismatches.  Third: window listing the different rows of both table 1 and table 2.  Fourth: the table G4\_\_CONSTRAINT\_TABLE.XML. |
| Concluding Remarks: | | | | | | |
| Testing Team:  Producer: Ricardo Sanchez  Evaluators: Sebastian Nunez, Brian Cardiel | | | Date Completed: TBD | | | |

## Test G4.D1

**Objective:** The objective of this test case is to check for duplicate rows in an empty table (0 rows, 0 columns).

**Notes:** This test case should take less than 5 minutes to successfully execute. This case uses the default configuration of selecting columns, in other words, all columns are taken into consideration.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: G4.D1 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows in an empty table. | | | | | | |
| Testing approach: This test will be conducted using the dbEdit.jar file using either windows or macOS. Behavior is observed on the simulation screen. The sample screenshots on the expected results column should match the operator’s after performing the corresponding operator action. | | | | | | |
| STEP | OPERATOR ACTION | PURPOSE | | | EXPECTED RESULTS | COMMENTS |
| 1 | Run the dbEdit executable (right click the dbEdit file and click Open). | Initial condition | | |  |  |
| 2 | Click “File” on the menu bar, then click the “Open” option.  After the file search path window open, select and open the file: G4\_\_EMPTY\_TABLE.XML | Open and display the empty table: G4\_\_EMPTY\_TABLE.XML | | |  | Output: window with the table: G4\_EMPTY\_TABLE.XML |
| 3 | Click “File” on the menu bar, then click the “Check for duplicate rows”. | Check for duplicates in a table with no column fields or row fields. | | |  | Output: window with the number of duplicates. In this case, since the table is empty, it found 0 duplicates. |
| Concluding Remarks: | | | | | | |
| Testing Team:  Producer: Ricardo Sanchez  Evaluators: Sebastian Nunez, Brian Cardiel | | | Date Completed: TBD | | | |

## Test G4.D2

**Objective:** The objective of this test case is to compare two different versions of the same table file.

**Notes:** This test case should take less than 5 minutes to successfully execute.

The only difference in both files is the row with index 2,

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: G4.D2 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows in a table with 1 duplicate. | | | | | | |
| Testing approach: This test will be conducted using the dbEdit.jar file using either windows or macOS. Behavior is observed on the simulation screen. The sample screenshots on the expected results column should match after performing the corresponding operator action. | | | | | | |
| STEP | OPERATOR ACTION | PURPOSE | | | EXPECTED RESULTS | COMMENTS |
| 1 | Run the dbEdit executable (right click the dbEdit file and click Open). | Initial condition | | |  |  |
| 2 | Click “File” on the menu bar, then click the “Open” option.  After the file search path window open, select and open the file: G4\_\_TEST\_TABLE~.XML | Open and display a table with repeated column fields. | | |  | Output: window with the table: G4\_\_TEST\_TABLE~.XML |
| 3 | Click “Options” on the menu bar, then click the “Column selection” option. | Open the Enable/Disable columns window. | | |  | Output: window with the column fields of the table. |
| 4 | Once the Enable/Disable window is open, unselect the fields: INDEX and DESCRIPTION. | Leave only one field to compare. | | |  | This window can be left open or closed after unselecting the fields. |
| 5 | On the table window (the one on step 2), click “File” on the menu bar, then click the “Check for duplicate rows”. | Check for duplicates with only one column selected | | |  | First output: window with the number of duplicates found  Second: window with value of the duplicate key.. |
| Concluding Remarks: | | | | | | |
| Testing Team:  Producer: Ricardo Sanchez  Evaluators: Sebastian Nunez, Brian Cardiel | | | Date Completed: TBD | | | |

## Test G4.D3

**Objective:** The objective of this test case is to observe how the program handles a boundary when checking for duplicate rows.

**Notes:** This test case should take less than 5 minutes to successfully execute. The test consists of checking for duplicates on a table with no duplicates while at the same time deselecting all columns with the column selection feature.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: G4.D3 | | | | Current Status: Pending | | |
| Test title: Check for duplicate rows in a table with no duplicates | | | | | | |
| Testing approach: This test will be conducted using the dbEdit.jar file using either windows or macOS. Behavior is observed on the simulation screen. | | | | | | |
| STEP | OPERATOR ACTION | PURPOSE | | | EXPECTED RESULTS | COMMENTS |
| 1 | Run the dbEdit executable (right click the dbEdit file and click Open). | Initial condition | | |  |  |
| 2 | Click “File” on the menu bar, then click the “Open” option.  After the file search path window open, select and open the file: G4\_\_TEST\_TABLE.XML | Open a table with 0 row repetitions, both individually and as a whole.. | | |  | Output: window with the table: G4\_\_TEST\_TABLE.XML |
| 3 | Click “Options” on the menu bar, then click the “Column selection” option. | Open the Enable/Disablecolumns window. | | |  | Output: window with the column fields of the table. |
| 4 | Once the Enable/Disable window is open, unselect the fields: INDEX and DESCRIPTION. | Disable all column fields when comparing. | | |  | This window can be left open or closed after unselecting the fields. |
| 5 | On the table window (the one on step 2), click “File” on the menu bar, then click the “Check for duplicate rows”. | See if the program handles this operation correctly. | | |  | First output: window with the number of duplicates found.  Second: 2 windows with the values of the duplicate keys found.  Third: Changes on the window with the table (from step 2), the highlights in yellow indicate duplicates. |
| Concluding Remarks: | | | | | | |
| Testing Team:  Producer: Ricardo Sanchez  Evaluators: Sebastian Nunez, Brian Cardiel | | | Date Completed: TBD | | | |

# **Test Schedule**

Below is the test schedule for reviews and adjusts.

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

# **Appendix**

TBD

$