Edit Menu Operations

Test plan

Version <1.0>

<date>

Document Control

Approval

The Guidance Team and the customer shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 4/6/20 |
| Current Release: | 4/7/20 |
| Indicator of Last Page in Document: | $$ |
| Date of Last Review: |  |
| 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: Juan A. Gaucin, Carlos I. Vargas, Valeria Macias

Customer: Dr. Steven Roach

Software Team Members: Dr. Steven Roach

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 4/6/20 | Valeria Macias | Edited Sections 1,2,3 and 4 |
| 1.1 | 4/7/20 | Valeria Macias | Edited every section of the document |
|  |  |  |  |

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 XX 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 created for the purposes of understanding Test Plans and Reviews for CS 5387 Software Verification and Validation at the University of Texas at El Paso. This Test Plan is designed to test the edit menu operations of a database editing software.

## Purpose

The purpose of this Test Plan is to test the functionality of the edit menu operations of the database editing software. The edit menu operations include: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. This Test Plan follows a unit test plan approach for two functions, then an integration of units for two functions, then an integration of units for four functions. The details of this test plan system are elaborated in the 1.3 System Overview section below.

## Scope

This Test Plan is designed to test the latest version of the software released by Dr. Steven Roach on the Github repository created on April 5, 2020.

## System Overview

This Test Plan follows a unit test plan approach for the edit menu operations: “Insert” and “Delete” because these two operations can be performed without the other edit menu operations. However, the edit menu operation “Copy” is not testable without the edit menu operation “Paste” and vice versa. Thus, this Test Plan then follows an integration of units approach for “Copy” and “Paste” as these two functions necessarily work together. Finally, the edit menu operation “Undo” can only be performed after other edit menu operations are performed. Thus, this Test Plan takes an integration of units approach for “Undo” as the operation undo is tested after all other operations.

This Test Plan approach of unit test, unit test, integration test between two functions, integration test between all functions, must follow a sequential order to ensure that each operation works correctly before being tested as an integration of units.

## Suspension and Exit Criteria

The test cases in this Test Plan are meant to be tested sequentially in so far as the final test plan is dependent on the pass rate of the first three test plans. Thus, suspension of testing is required if the first three test plans have less than a 90% pass rate. Otherwise, all test plans must pass for completion of edit menu operation functionality.

## Document Overview

The remainder of this document outlines the edit menu features to be tested as well as other features of the software which will not be tested. The Testing Approach is an overview of each test plan. Each test plan focuses on one edit menu operation. Each test plan has more than one test to test the edit menu operation. Following the test plans is a comment on the user interface testing, then a test schedule and finally the appendix.

## References

Spec.docx

DbEditTestPlanAssign.pdf

TestPlanAssign040120.pptx

# Test Items and Features

The following is a table which outlines which features are to be tested and which features are not to be tested.

|  |  |
| --- | --- |
| *Features to be tested* | *Features not to be tested* |
| Copy | Check for Duplicate Rows |
| Delete | Close |
| Edit Menu Operations | Column Selection |
| Insert | Compare |
| Paste | File Menu |
| Undo | File Save |
|  | Filter |
|  | Help |
|  | Open |
|  | Options |
|  | Options Display |
|  | Print |
|  | Search Menu |

# Testing Approach

The following is a table that describes each test case. Each test case is designed to test each of the edit menu operations. Each test case contains more than one test to test the edit menu operation.

Table 1: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE EditMenu** | | |
| **Description of Test Suite** | **This test suite is designed to test the functionality of all Edit Menu operations.** | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| Insert | **To test the functionality of the Insert operation** | **5** |
| Delete | **To test the functionality of the Delete operation** | **5** |
| Copy/Paste | **To test the functionality of the Copy and Paste operations** | **5** |
| Undo | **To test the functionality of the Undo operation** | **5** |

# Test EditMenu

Each test case is separated by sections with each section containing more than test. Each of the following tests use the TEST\_DB.xml file in conjunction with the system. Each test has a series of steps which are to be followed. A description of the purpose of each step and the expected output is also included. A test fails if the expected output does not match the actual output. A test passes if the expected output matches the actual output. A test is pending if the test has not been performed or if the test suite has been suspended due to a below 90% pass rate. The test team should identify themselves at the bottom of each test, additional comments should be made in case of any unforeseen problems or constraints. Finally, the date should be recorded for configuration management.

## Test Insert

**Objective:** To test the functionality of the Insert operation in the Edit Menu.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: I001 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Insert Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Insert operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 4 | Select “Insert” | This completes the Insert Row test. | | | An empty row which was not there previously should appear at the end of the table. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

**Notes:** The test is to be conducted using the TEST\_DB.xml file. There is only one test in this section because the Insert operation should execute only one function. The Insert function is strictly a row operation and thus, only one expected output exists when executing the Insert operation.

## Test Delete

**Objective:** To test the functionality of the Delete operation in the Edit menu.

**Notes:** The following tests are to be conducted using the TEST\_DB.xml file. There are six tests in this section. The Delete function is strictly a row operation. Thus, any Delete attempts on columns should be unsuccessful and the user should be notified of the constraint.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D001 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete First Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the first row in the table. | This selects the row which is to be deleted. | | | The selected row is highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete First Row test. | | | The first row should be removed from the table. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D002 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete Last Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the last row in the table. | This selects the row which is to be deleted. | | | The selected row is highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete Last Row test. | | | The last row should be removed from the table. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D003 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete Middle Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the third row in the table. | This selects the row which is to be deleted. | | | The selected row is highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete Middle Row test. | | | The third row should be removed from the table. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D004 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete Multiple Rows | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the first three rows in the table. | This selects the rows which are to be deleted. | | | The selected rows are highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete Multiple Rows test. | | | The first three rows should be removed from the table. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D005 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete Header Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the header row. | This selects the row which will attempt to be deleted. | | | The selected row is highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete Header Row test. | | | The header row should remain unchanged and the user should be notified that the Header row cannot be deleted. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: D006 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Delete Column | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the INDEX column. | This selects the column which will attempt to be deleted. | | | The selected column is highlighted and the system is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | This completes the Delete Column test. | | | The INDEX column should remain unchanged and the user should be notified that the delete function is strictly a row operation. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

## Test Copy/Paste

**Objective:** To test the functionality of the Copy and Paste operations.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: CP001 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Copy First Row Paste on Last Row | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXPECTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 appendix for contents.* |  |
| 3 | Select the first row. | This selects the row which is to be copied. | | | The selected row is highlighted and is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Copy operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Copy” | This should copy the first row. | | | The system returns to the previous display in step 2. |  |
| 6 | Select the last row. | This selects the row which is to be pasted. | | | The selected row is highlighted and is awaiting the next operation. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Paste operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Paste” | This completes the Copy First Row Paste on Last Row test. | | | The contents of the first row should be displayed on the last row. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

**Notes:** The following tests are to be conducted using the TEST\_DB.xml file. The tests take an integration of units approach to test the Copy function with the Paste function because these functions are necessarily used in conjunction together and cannot be tested separately.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: CP002 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Copy Last Three Rows Paste on First Three Rows | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the last three rows. | This selects the rows which are to be copied. | | | The selected rows are highlighted and are awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Copy operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Copy” | This should copy the last three rows. | | | The system returns to the previous display in step 2. |  |
| 6 | Select the first three rows | This selects the rows which are to be pasted. | | | The selected rows are highlighted and are awaiting the next operation. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Paste operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Paste” | This completes the Copy Last Three Rows Paste on First Three Rows test. | | | The contents of the last three rows should be displayed on the first three rows. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: CP003 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Copy Paste Column | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the INDEX column. | This selects the column which is to be copied. | | | The selected column is highlighted and is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Copy operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Copy” | This should copy the INDEX column. | | | The system should notify the user that columns cannot be copied |  |
| 6 | Select the DESCRIPTION column | This selects the column which is to be pasted. | | | The selected column is highlighted and is awaiting the next operation. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Paste operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Paste” | This completes the Copy Paste Column | | | The table should remain unchanged and the system should notify the user that columns cannot be edited. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

## Test Undo

**Objective:** To test the functionality of the Undo operation.

**Notes:** The following tests are to be conducted using the TEST\_DB.xml file. The tests take an integration of units approach to test the Undo function against all of the other functions because the undo function is meant to undo the edits done by the other functions. Please suspend testing here if less than a 90% pass rate has been achieved by sections 3.1, 3.2 and 3.3 tests. Please continue otherwise.

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: U001 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Undo Insert | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Insert operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Insert” | The Undo operation will be tested against the Insert operation | | | An empty row which was not there previously should appear at the end of the table. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Undo operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Undo” | This completes the Undo Insert test. | | | The empty row which appeared in step 5 should be removed and the table should return to its original state displayed in step 2. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: U002 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Undo Delete | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the last row. | This selects the row which is to be deleted. | | | The selected column is highlighted and is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Delete operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Delete” | The Undo operation will be tested against the Delete operation | | | The selected row should be removed. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Undo operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Undo” | This completes the Undo Delete test | | | The row which was deleted in step 5 should return and the table should return to its original display shown in step 2. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Test No.: U003 | | | | Current Status: << Passed / Failed / Pending >> | | |
| Test title: Undo Copy Paste | | | | | | |
| Testing approach: To perform this test, run TEST\_DB.xml file in conjunction with the system. The following operations may be done manually or may be automated. | | | | | | |
| STEP  1 | OPERATOR ACTION  Run TEST\_DB.xml with the system | PURPOSE  This initiates the testing environment | | | EXEPCTED RESULTS  The system displays a database edit window with the names of each Table. | COMMENTS |
| 2 | In the database edit window, select CONSTRAINT\_TABLE | This takes the tester to the Table Edit window where the Edit Menu operations are accessible. | | | The system displays a table edit window with the CONSTRAINT\_TABLE displayed. *Please see A1 in appendix for contents.* |  |
| 3 | Select the first row. | This selects the row which is to be copied. | | | The selected row is highlighted and is awaiting the next operation. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Copy operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Copy” | This should copy the first row. | | | The system should notify the user that columns cannot be copied |  |
| 6 | Select the last row. | This selects the row which is to be pasted. | | | The selected row is highlighted and is awaiting the next operation. |  |
| 7 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Paste operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 8 | Select “Paste” | This completes the copy paste function which is to be tested against the Undo function. | | | The table should display the contents of the first row copied onto the last row. |  |
| 4 | In the table edit window, select the Edit button on the menu options displayed above the table. | This opens the Edit Menu where the Undo operation is accessible. | | | The system displays a drop down menu containing: “Insert”, “Delete”, “Copy”, “Paste”, and “Undo”. |  |
| 5 | Select “Undo” | This completes Undo Copy Paste test. | | | The last row should return to its original contents. |  |
| Concluding Remarks: | | | | | | |
| Testing Team:  << List members of testing team and lead >> | | | Date Completed: | | | |

# User Interface Testing

The following tests the user interface of the system as it concerns the Edit Menu. The tests are represented as questions and each question needs a response of “Yes” to pass the test.

|  |  |
| --- | --- |
| Tests | Pass/Fail |
| *Spelling* |  |
| In the file menu, is the edit button displayed as “Edit”? |  |
| In the edit menu, is the insert button displayed as “Insert”? |  |
| In the edit menu, is the delete button displayed as “Delete”? |  |
| In the edit menu, is the copy button displayed as “Copy”? |  |
| In the edit menu, is the paste button displayed as “Paste”? |  |
| In the edit menu, is the undo button displayed as “Undo”? |  |
|  |  |
| *Usability* |  |
| Do each of the operations execute using one click from the edit menu? |  |
| Is the edit menu displayed at the top of the table view window? |  |
|  |  |
| *Error Handling* |  |
| Whenever the user tries to commit an edit menu operation on a column, is the user notified that the edit menu operation cannot be conducted on columns? |  |
| Whenever the user tries to commit an edit menu operation on the header row, is the user notified that the edit menu operation cannot be conducted on the header row? |  |

# Test Schedule

The following is a table which schedules the task, the task completion date, the persons to conduct the task and a description of the task for purposes of scheduling organization and completion. The test schedule will be determined at a later date.

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

# Other Sections

TBD

# Appendix

A1. The expected output described in every test prior to performing the edit menu operation under test or after performing the undo operation in the undo tests.

A1.

