Database Table Editor

Test plan

Version 1.0

04/09/2020

Document Control

Approval

The Guidance Team shall approve this document.

Document Change Control

|  |  |
| --- | --- |
| Initial Release: | 1.0 |
| Current Release: | 1.0 |
| Indicator of Last Page in Document: | \* |
| Date of Last Review: | 04/09/2020 |
| Date of Next Review: | 04/10/2020 |
| Target Date for Next Update: | 04/11/2020 |

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:

Dr. Steve Roach

Software Team Members:  
 Bianca Alvarez  
 Carlos Vargas  
 Juan Gaucin

Change Summary

The following table details changes made between versions of this document

|  |  |  |  |
| --- | --- | --- | --- |
| Version | Date | Modifier | Description |
| 1.0 | 04/09/2020 | Bianca Alvarez | First version of test plan created |
|  |  |  |  |
|  |  |  |  |

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](#_Toc37402722)

[Approval ii](#_Toc37402723)

[Document Change Control ii](#_Toc37402724)

[Distribution List ii](#_Toc37402725)

[Change Summary ii](#_Toc37402726)

[1. Introduction 4](#_Toc37402727)

[1.1. Purpose 4](#_Toc37402728)

[1.2. Scope 4](#_Toc37402729)

[1.3. System Overview 4](#_Toc37402730)

[1.4. Suspension and Exit Criteria 5](#_Toc37402731)

[1.5. Document Overview 5](#_Toc37402732)

[1.6. References 5](#_Toc37402733)

[2. Test Items and Features 6](#_Toc37402734)

[3. Testing Approach 7](#_Toc37402735)

[4. Test XX 9](#_Toc37402736)

[4.1. Test Suite 1 9](#_Toc37402737)

[4.2. Test Suite 2 12](#_Toc37402738)

[4.3. Test Suite 3 14](#_Toc37402739)

[5. User Interface Testing 17](#_Toc37402740)

[6. Test Schedule 18](#_Toc37402741)

[7. Other Sections 19](#_Toc37402742)

# Introduction

<< This section gives introductory information regarding the project, the system to be tested, and the testing approach.>>

The plan for this project is to produce a reasonable test plan for the program, database table editor.

## Purpose

<< Identify the project and stipulate the test plan purpose by indicating what the document contains (e.g., organizational responsibilities, test approach, test schedule. There are generally four different types of test plans: project test plan that describes the overall strategy for testing; the system test plan that describes the system from the customer’s point of view; integration test plan that describes integration of units and subsystems; unit test plan that describes modules or classes. This section needs to identify which of these this document is.>>

The purpose of this document is to define a project test plan which will describe the overall strategy for testing.

## Scope

<<Specify the project software releases/versions encompassed by the plan. >>

The test plan described in this document is draft by the producer and reviewers of Group 1, Team 2. The following versions have been released:

* V1 – Initial Test Plan by Producer

## System Overview

<<Describe the system to be exercised by the testing approach specified in the plan. This overview serves to identify aspects of the system operation that will be the focus of the plan’s testing approach. This should align with the systems overview of other documents in the project.>>

The system to be tested is a simple database table editor. This editor allows to view and modify the data, the editor does not allow modification of the database schema. The configuration data is stored in XML tables, and the database is organized in XML documents.

Figure 1 Types of XML documents

XML documents

Database Description

Data Tables

The database description file contains a description of the data tables, including table name and columns for each of one of the tables. As an example, consider file TEST\_DB.xml

The data tables names are included in the database description file after the tag <TABLE>. As an example, consider the file TEST\_DB\_\_BIG\_TABLE.XML, which contains the data for the database table description described above.

Our objective is to focus on file menu operations excluding compare file to other version, check for duplicate rows, and filter. Also, including the search menu option on the database window.

## Suspension and Exit Criteria

<< “suspension criteria” describes when we suspend testing, to be resumed at a later time. For example, if 40% of the test cases fail, or if any of the critical test cases fail. If there are no suspension criteria, indicate that all tests cases will be executed. “Exit criteria” indicates when testing stops. This could be based on run rate (number of test cases run divided by number of test cases specified) or pass rate (number of test cases passed divided by number of test cases run, or test cases passed divided by number of test cases specified). Nominally, we expect to run all of the specified tests. We want the pass rate to be high. We might specify that all critical tests must pass, and 90% of the non-critical must pass. In general, we want this to be high. >>

There are no suspension criteria, all tests cases will be executed.

The exit criteria are met when all critical tests and 90% of the non-critical tests pass, this will ensure the pass rate is high.

## Document Overview

<<Describe the remainder of the document.>>

## References

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

# Test Items and Features

<< This section describes the test items (e.g., components, classes, functions or methods) and the features to be tested. It may also list features not to be tested. A class diagram is useful. A table of features is useful. >>

Features to be tested:

The File Menu has the options to open, close, and print tables. The following operations are included:

* Open – A user can open either a database description file or a data file
  + If a data file is opened, the corresponding data description is also opened
  + If a database description is opened, the window will display all the database tables
    - If selecting a table from this display, will open the data table display in a new window
    - It is possible to open more than one database at a time
    - Opening a database table file results in the database description also being opened.
    - All the windows associated with a specific database have the same color outline.
  + Opening an already-opened table should result in the table editor window for that table being raised to the top of the screen and being visible.
* Print to Fit
* Print Normal
* Print CSV
* Close
  + If the file has been modified and the user attempts to close the file, a save prompt should be displayed.
* Save(^s)
  + On file save, the user is prompted to enter a new history entry.
* Save as
* Exit
* Search Menu
  + Search opens a file search dialog that allows a user to search specified directories for files that contain a given text string.
  + Find Files
    - When using this option, files containing the text string are listed.
    - Selecting the file from the display will open the database table.

Features not to be tested:

* Compare file to other versions
* Check for duplicate rows
* Filter
* Edit Menu Operations
* Search function in the edit menu

# Testing Approach

<<Describe the approach to be used to the test the system. This description includes specifying the types of tests to be performed, e.g., tests designed to exercise system functions one by one; tests designed to exercise sequences of functions that approximate operational use of the system; tests designed to stress the system to its design and requirements limits. The description lists the specific tests to be performed, but does not give the test steps. For each of these tests, give it a name and specify its objective. Label the criticality of the test cases. >>

The criticality was classified as following:

* High – Functions required for other parts of the system to test
* Medium – Function that should be working but it is not required by another function
* Low – Display, appearance, colors,

Table 1: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE 1** | | |
| **Description of Test Suite** | Testing option Open under the File menu A user can open either a database description file or a data file. | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| T1-1 | Test that the file menu option “Open” opens a data file and database description at the same time | High |
| T1-2 | Test that the file menu option “Open” opens a database description and displays all database tables | High |
| T1-3 | After opening a database description, test if selecting a table from the window will open the data table display in a new window | High |
| T1-4 | Test when the data file is opened, the corresponding data description is also opened | High |
| T1-5 | Test when opening an already opened table should result in the table editor window for that table being raised to the top of the screen and being visible | Medium |
| T1-6 | Check that all windows associated with a specific database have the same color outline | Low |
| T1-7 | Check that is possible to open more than one database at a time | High |

Table 2: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE 2** | | |
| **Description of Test Suite** | Testing option Print under the File menu | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| T2-1 | Test the Print to Fit option | Medium |
| T2-2 | Test the Print Normal option | Medium |
| T2-3 | Test the Print CSV option | Medium |

Table 3: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE 3** | | |
| **Description of Test Suite** | Testing options Save, and Close under the File menu | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| T3-1 | Test when on file Save(^s), the user is prompted to enter a new history entry | Medium |
| T3-2 | Test option Save as | Medium |
| T3-3 | Test that if the file has been modified and the user attempts to close the file, a save prompt should be displayed | Medium |
| T3-4 | Test option Exit | Medium |

Table 4: Test Plan

|  |  |  |
| --- | --- | --- |
| **TEST SUITE 4** | | |
| **Description of Test Suite** | Testing Search Menu | |
| **Test Case Identifier** | **Objective** | **Criticality** |
| T4-1 | Test that Search opens a file search dialog that allows a user to search specified directions for files that contain a given text string | High |
| T4-2 | Test when using the option Find Files, files containing the text string are listed | High |
| T4-3 | Test that when selecting the file from the display will open the database table | High |

# Test XX

<<The purpose of this section is to:

* document test input, specific test procedures, and outcomes.
* establish test methods,
* explain the nature and extent of each test >>

<< for each test case, complete the following: >>

## Test Suite 1

**Objective:** Test that the file menu option “Open” opens a database description and displays all database tables

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T1-2 | | | | **Current Status:** Passed | | |
| **Test title:** Open database description | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Open the application dbEdit | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window is displayed with no tables | **COMMENTS** |
| 2. | Click the button “File” located at the top left corner | Opens the file menu | | | The file menu is displayed |  |
| 3. | Click the button “Open” from the File menu | Option to open a file | | | The file search path window opens |  |
| 4. | Select an XML file, for this test case, select the file TEST\_DB.xml | Test to prove that when opening a database description, all table names in this db are displayed | | | The window will display all the database tables |  |
| **Concluding Remarks:** When opening a database description file, all tables names will appear on the window display | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** After opening a database description, test if selecting a table from the window will open the data table display in a new window

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T1-3 | | | | **Current Status:** Passed | | |
| **Test title:** Open table from the window display | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >>  Refer to steps in T2 to open a database description file | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Ensure the application is opened and you have opened the file TEST\_DB.xml | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  The window will display all the database tables | **COMMENTS** |
| 2. | Select the table BIG\_TABLE from the window display | Check that a table will open in a new window | | | Table BIG\_TABLE opened in a new window |  |
| 3. | Select the table CONSTRAINT\_NAME from the window display | Check that a table will open in a new window | | | Table CONSTRAINT\_NAME opened in a new window |  |
| 4. | Select the table TYPE\_TABLE from the window display | Ensure all tables from the database, will open in a new window | | | Table TYPE\_TABLE opened in a new window |  |
| **Concluding Remarks:** All tables in the window display from the database description, opened in a new window | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** Test when the data file is opened, the corresponding data description is also opened

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T1-4 | | | | **Current Status:** Passed | | |
| **Test title:** Open data file | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Open the application dbEdit | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window is displayed with no tables | **COMMENTS** |
| 2. | Click the button “File” located at the top left corner | Opens the file menu | | | The file menu is displayed |  |
| 3. | Click the button “Open” from the File menu | Option to open a file | | | The file search path window opens |  |
| 4. | Select an XML file, for this test case, select one of the data files, like TEST\_DB\_TYPE\_TABLE.xml | Ensure that the corresponding data description is also opened when opening a data file | | | The application opens the database description TEST\_DB and the data file TYPE\_TABLE |  |
| **Concluding Remarks:** When opening a data file, the corresponding database description will open | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** Test when opening an already opened table should result in the table editor window for that table being raised to the top of the screen and being visible

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T1-5 | | | | **Current Status:** Passed | | |
| **Test title:** Opening an already opened table | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | In the list of tables in the main window, click BIG\_TABLE | Open fist table | | | BIG\_TABLE will be displayed in a new window |  |
| 3. | In the list of tables in the main window, click CONSTRAINT\_TABLE | Open second table | | | CONSTRAINT\_TABLE will be displayed in a new window |  |
| 4. | In the list of tables in the main window, click TYPE\_TABLE | Open third table | | | TYPE\_TABLE will be displayed in a new window |  |
| 5. | In the list of tables in the main window, click again BIG\_TABLE | Open an already opened table | | | BIG\_TABLE window will appear in front of other windows |  |
| 6. | Click the button “File” located at the top left corner, then select Open and select the file TEST\_DB\_CONSTRAINT\_TABLE.xml | Open an already opened table using the File Menu | | | CONSTRAINT\_TABLE window will appear in front of other windows | To stop selecting a table name from the list, right click outside this list |
| **Concluding Remarks:** When opening an already opened table, this table window will appear to the top of the screen and being visible | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

## Test Suite 2

**Objective:** Test the Print to Fit option

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T2-1 | | | | **Current Status:** Passed | | |
| **Test title:** Print to Fit | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | Click the button “File” located at the top left corner, then select Print to Fit | Test Print to Fit option | | | The window Print will appear |  |
| 3. | On the option Name, select Microsoft to Print to PDF, then click on Print, and select the path destination to save the PDF file | Test Print to Fit option using PDF print | | | The file should be saved on the path indicated |  |
| 4. | Look for the file that was printed in the step before, open the file to see the printed file | Open file printed to PDF | | | The database description table is on the printed PDF file (the size of the font was increased to fit on the page) |  |
| **Concluding Remarks:** The option Print to Fit works as expected | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** Test the Print Normal option

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T2-2 | | | | **Current Status:** Passed | | |
| **Test title:** Print Normal | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | Click the button “File” located at the top left corner, then select Print Normal | Test Print Normal option | | | The window Print will appear |  |
| 3. | On the option Name, select Microsoft to Print to PDF, then click on Print, and select the path destination to save the PDF file | Test Print Normal option using PDF print | | | The file should be saved on the path indicated |  |
| 4. | Look for the file that was printed in the step before, open the file to see the printed file | Open file printed to PDF | | | The database description table is on the printed PDF file |  |
| **Concluding Remarks:** The option Print Normal works as expected | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** Test the Print CSV option

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T2-3 | | | | **Current Status:** Passed | | |
| **Test title:** Print CSV | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | Open the data file CONSTRAINT\_TABLE | Beginning of test | | | The table CONSTRAINT\_TABLE will open |  |
| 3. | Click the button “File” located at the top left corner, then select Print CSV | Test Print CSV | | | A message will appear on top of the window with the message of the numbers of records found + the destination path |  |
| 4. | Look for the CSV file on the folder where the application is located, open the file | Open CSV file | | | When opening the csv file, the records of the CONSTRAINT\_TABLE are visible |  |
| **Concluding Remarks:** The option Print CSV works as expected | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

## Test Suite 3

**Objective:** Test when on file Save(^s), the user is prompted to enter a new history entry

**Notes:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T3-1 | | | | **Current Status:** Passed | | |
| **Test title:** Save(^s) | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | Open the data file CONSTRAINT\_TABLE | Beginning of test | | | The table CONSTRAINT\_TABLE will open |  |
| 3. | Change a value on the table, then on your keyboard press CTRL + S | Test Save ^s | | | The window XML history input opens |  |
| 4. | Fill the blanks on the window, entering the author and description, then press Okay | Save a new version of the table | | | A message will appear on top of the window with the destination path of the new table saved |  |
| 5. | Go to the main window, where all the table names are displayed Click on File > Open > Select the file TEST\_DB\_CONSTRAINT\_TABLE  Repeat this step to open the file TEST\_DB\_CONSTRAINT\_TABLE~ | Open the two versions of the table CONSTRAINT\_TABLE | | | The two tables appear on different windows, the table with the (~) at the end of the name, is the table history | Compare both tables |
| **Concluding Remarks:** The option Save(^s) works as expected | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

**Objective:** Test option Save as

**Notes:**  The new table will not be saved on the XML file of the database description, for this table to appear on the database, the user will need to enter manually the name of the table to the XML file

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test No.:** T3-2 | | | | **Current Status:** Passed | | |
| **Test title:** Save as | | | | | | |
| **Testing approach:** <<Included in this section is a description of test harnesses, testing frameworks, environmental requirements, test tools and test automation that will be employed to achieve testing. Include naming conventions for tests and test scripts if appropriate. Provide requirements traceability and test priority.  >> | | | | | | |
| **STEP**  1. | **OPERATOR ACTION**  Follow instructions on T1-2 to open a database description file | **PURPOSE**  Beginning of test | | | **EXEPCTED RESULTS**  Main window will display list of tables in database | **COMMENTS** |
| 2. | Open the data file CONSTRAINT\_TABLE | Beginning of test | | | The table CONSTRAINT\_TABLE will open |  |
| 3. | Click on File > Save as | Test Save as | | | The window XML history input opens |  |
| 4. | Fill the blanks on the window, entering the author and description, then press Okay | Save a new version of the table | | | A message will appear on top of the window with the destination path of the new table saved | The name for the new table should follow the convention name TEST\_DB\_\_CO  NSTRAINT\_  TABLE2 |
| 5. | Go to the main window, where all the table names are displayed Click on File > Open > Select the file TEST\_DB\_CONSTRAINT\_TABLE  Repeat this step to open the file TEST\_DB\_CONSTRAINT\_TABLE2 | Open the two versions of the table CONSTRAINT\_TABLE | | | The two tables appear on different windows | Compare both tables |
| **Concluding Remarks:** The option Save as works as expected | | | | | | |
| **Testing Team:**  Bianca Alvarez -Lead Carlos Vargas Juan Gaucin | | | **Date Completed:** 04/09/2020 | | | |

# User Interface Testing

<<This section focuses on the interaction between the user and the system. For testing the user interface, consider the following traits:

* Consistent terminology, shortcut keys, menu selections, and presentation
* Correct language, spelling, and grammar.
* Flexibility in navigation between windows and interface elements.
* Error handling that will inform user of critical operations.
* Follows standards and guidelines such as placement of scroll bars, windows, and menu items.

This section could be integrated into Section 4.

>>

# 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** |
| Test T1-1 to T |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |

# 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

>>