

Test Specification

Table of Contents

<u>TABLE OF CONTENTS</u>	1
<u>1.0 INTRODUCTION</u>	2
<u>1.1 GOALS AND OBJECTS</u>	2
<u>1.2 STATEMENT OF SCOPE</u>	2
<u>1.3 MAJOR CONSTRAINTS</u>	3
<u>2.0 TESTING PLAN</u>	3
<u>2.1 SOFTWARE (SCIS) TO BE TESTED</u>	3
<u>2.1.1 Interfaces</u>	3
<u>2.2 TESTING STRATEGY</u>	6
<u>2.2.1 Unit Testing</u>	6
<u>2.2.2 Integration Testing</u>	7
<u>2.2.3 Validation Testing</u>	7
<u>2.2.4 High-order Testing</u>	7
<u>2.3 TESTING RESOURCES AND STAFFING</u>	8
<u>2.4 TEST RECORD KEEPING</u>	8
<u>2.5 TESTING TOOLS AD ENVIRONMENT</u>	8
<u>2.6 TEST SCHEDULE</u>	9
<u>3.0 TEST PROCEDURE</u>	9
<u>3.1 SOFTWARE (SCIS) TO BE TESTED</u>	9
<u>3.2 TESTING PROCEDURES</u>	9
<u>3.2.1 Unit Testing</u>	9
<u>3.2.2 Integration Testing</u>	19
<u>3.2.3 Validation Testing</u>	20
<u>3.2.4 High-order Testing</u>	20
<u>3.3 TESTING RESOURCES AND STAFFING</u>	22
<u>3.4 TEST RECORD KEEPING AND LOG</u>	22

1.0 Introduction

This section gives a general overview of the Test Specification for the **Waste Management Inspection Tracking System (WMITS)**.

1.1 Goals and Objects

Put it in a simple way, a good product will be work perfectly, doing the right thing at the right time. To do that, the software has to go through a series of tests before its final release. Error free software is extremely difficult to achieve. After all, nothing is perfect. Especially for software developed in a short time frame. But high quality can be achieved with a detailed test specification. All (or least most) of the test case will be listed, the development team will follow it step by step, item by item, to test all the necessary objects, data flows, limits, boundaries, and constraints of the software.

Cyber Rovers would like to have a test specification to counter any difficulties that may impact the development and the future performance of the software. The team's goal is to assist the project team in developing a strategy to deal with any errors. For this, the team will take a look at the most common errors to some very uncommon errors as well.

1.2 Statement of Scope

An overall plan for integration of the software and a description of specific tests are documented in this section. Below are the different kinds of tests that the team will take to ensure the quality of the software.

- Unit Testing
 - Desktop Application
 - Database
 - Palm-size PC Application

Unit tests will be performed using black box testing methods.

- Integration Testing
 - Desktop Application
 - Database
 - Palm-size PC Application

- Validation Testing
 - We will test software as whole, so all the units of the software will be included
 - Desktop Application
 - Database
 - Palm-size PC Application

- High-order Testing
 - The software will be tested for several test methods. Units to be tested are.

- Desktop Application
- Database
- Palm-size PC Application

1.3 Major Constraints

In this section we will talk about the business, technical or resource related constraint that may keep us from performing all tests necessary.

1. The team has limitation on time to test the product at the client's facility. We have access to the facility only during the regular office hours. We also have to set our schedule around the available time of the inspector that is to help us, so time schedule will be a major constraint when we talk about testing at the site.
2. The team also has got funding for only one hand held PC. This means that we cannot test the software using the PC from some other brand or PC that is of lesser price and lower hardware.
3. The team does not know any hacker that can help us test the security problems. So we have to rely on our own knowledge and have to trust the software for the security.
4. The team also does not have large enough group to have many people use the applications at the same time to perform real stress related testing. So we will not be able to test the product for the larger user base.

Critique: Each of these constraints represents a significant product quality risk. The team should consider risk mitigation strategies.

2.0 Testing Plan

We want the product to be bug free. We also want to make sure that there are no defects in the product. So we will be spending large amount of the total software development time on the testing. Below is the description of the testing procedure and strategy. We will also be presenting the timing and scheduled of the tests to be carried out.

2.1 Software (SCIs) to be tested

2.1.1 Interfaces

The tests to be carried on these interface windows are described below.

Login Window

We will make use of several different names to log in to the system, so will be testing login window. We will also test OK and Cancel buttons on this window by performing test above.

DEQ – Microsoft Visual Basic [Design] Window

This is the main window that we will use to access the database using Visual Basic. We will have several different drop-down menus in this window. File, Facility, Inspection, approve, Reports, Maintenance and Help are the drop down menu that will be available in this window we will try to use all the menus and than different options available in each of the window.

File:

When file button is clicked user will be shown two choices.

Switch User:

The function of this is to switch between the users.

Exit:

The function is to exit out the user.

Facility:

When a Facility button is clicked we will be presented with Facility Screen. We will be able to add or edit facility using the screen.

Inspection:

When Inspection button is clicked user will be shown five choices.

Create/Modify:

When click on this choice we will be presented with Inspection Step 1 Screen used to create or modify inspection.

File Results:

When we select this choice we will be presented with File Result Window.

Get Approval:

When we select this choice we will be presented with Get Approval Window. We will test the button by clicking on it to go to Get Approval Window.

Print Letters:

When we select this choice we will be presented with Print Letter Window. We will be able to go to Print Letter Window. We want to make sure that the proper window is presented

View Schedule:

When we select this choice we will be presented with View Schedule Window. We will view the schedule using the button.

Approval:

This option is presented and works only for manger. We need to make sure that when someone other than inspector loges in the button is disabled. We also need to make sure

that when selected by manager software will bring up window containing queued list of letters waiting to be approved.

Reports:**Print Staff Reports:**

When we select this choice we will be presented with Report Window.

Print Blank Checklists:

When we select this choice we will be presented with Report Window.

Efficiency Report:

When we select this choice we will be presented with Report Window.

Maintenance:**Checklist:**

When we select this choice we will be presented with Checklist window. We will be able to go to Checklist Window using this window. We want to make sure that the proper window is presented

Letters:

When we select this choice we will be presented with Letters window.

Users:

When we select this choice we will be presented with Users Window.

Options:

When we select this choice we will be presented with Option Window.

Help:**Content:**

A very standard help menu will be at this section. Which will include index, search, FAQs abilities. Mostly test on the ability to find the correct data, materials.

About:

When we select this choice we will be presented with About Window (window with software Info).

Inspection Generator Window

This window is to allow inspector to generate the inspection ID before actually going out for inspection. We will have several windows to select options form and also buttons to test from.

Print Checklist Window: In this window we are presented with the print checklist window. From this window we will be able to select blank check list and print it out.

Inspection Form Window is where user needs to make inspection form or from where inspector selects appropriate form from given choices.

Print Inspection Form Window

This window will allow inspector to select and print blank print list.

Choosing Inspection Window**Main Checklist Window**Available Checklist Items Window

We will several different menus and buttons to select items from so we will be testing each of this buttons

Selected Checklist Items

Allows us to select and add remove items from the checklist.

Inspection ID Explode Window

This window will come up when Explode button is selected form Choosing Inspection Window. It is created to allow user to find the Inspection ID.

Creating Facility

This window is utilized to create a facility entry in the database if facility does not exist. User will have to simply complete from with text boxes and few drop down menus.

Hand Held PC Integration:

We will be testing to make sure that all the checklists are present in Hand Held PC. For this we will open each and every one of the checklist and will carefully look at content of the form

2.2 Testing Strategy

In the following section we will describe the testing strategy. We will user four different methods to test our product.

2.2.1 Unit Testing

In the unit test case we will be testing the separate modules of the software. We will carry out white box testing where each module or component of the software is tested individually. We will test the components by passing data through it and we will be monitoring data to find the errors.

We will be looking for entry and exit conditions of the data. We will make sure that all the components work without any troubles.

The test primarily be carried out by the programmer who designed and implemented the module. Lead tester will than carry out test on the modules to finalise the testing.

2.2.2 Integration Testing

In this method of testing we will implement the software at the clients location and will run it. So we will be testing the product on clients network. As part of testing, will be looking for any signs of the collision between our software components and those of the clients. We want to make sure there is no confusion among the application on the network when they are running simultaneously.

We will install the software at the clentes site and will run it. We will have several different other applications open as well. This applications will be the once that have to interact with our software on normal bases. We will make sure that all the data is saved correctly and there is no loss of data or data base anomalies in the product.

We will start from the login window and will go through all the all the software functions and will test the. We will be carefully looking for any sort of collision between several different applications

2.2.3 Validation Testing

In this method of the test we will be working with the customer to find out if the software developed in valid for the clients. We want to make sure that the client is getting what he asked for. We will look at the software requirement document in the case of conflict or misunderstanding with client regarding software components.

We will perform the black box testing where the software is completed and we test all the software components together. We will have several input data or test data that we will derive results for. We will insert this data in the software and will get results form the software. We will compare the results from the software with the results that we derived. This way will check for the validation of the software.

In case there are problems with the software we will create a deficiency list and will record all the problems in there. We will test all the components and subcomponents of the software to perform validation test.

We have and will try our best so that we don't have to create deficiency list. This is necessary because if the errors are found at this stage of the software development we cannot fix them by the time we reach the software deliverance date. In this case we have to negotiate with the customer to give us extension on the project.

2.2.4 High-order Testing

In this test method we will combine several different other types of the testing. We will test for several different conditions by following several different test methods.

- **Recovery testing**

Here we are concerned with ability of the software to retrieve lost data. We want to make sure that the software is fault tolerance and does not loose data in case of system shutdown or if the system ceases.

- **Security Testing**
In this method of the test we want to make sure that the security checks are working and no one is able to temper with the data. This is crucial since our software is design to track the activity that is not legal.
- **Stress Testing**
In this test method we want to monitor tress caused to system and the software due to simultaneous use. We want to make sure that the system does not break down under the extreme use conditions.
- **Performance Testing**
Performance bounds are set during the design part of the software development. These bounds will help us in determining the effectiveness of the software. It will also help us to minimize stress level that is caused to user because of our software.

2.3 Testing Resources and Staffing

We will use several different resources to carry out the test on our software. Since The time is constraint for us we will try to use help from everyone that we can. Following is non-detailed description of the test resources

- WMD Staff
- WMD Resources
- Palm Size PC
- Bug Resource Report
- UMD Computer Resources

2.4 Test Record Keeping

Test record keeping and Test work Products are described in section 3.4 of Test Specifications Document. For Information regarding these topics, please refer to section 3.4 of the Test Specification Document

2.5 Testing Tools ad Environment

Testing tools will involve the computer resources form the University of Michigan-Dearborn Computer Labs and from the computer resources at Waste Management

Division of the Department for Environmental Quality. The Cyber Rovers Team will also use resources available to software development team outside of these two facilities.

2.6 Test Schedule

Following is the tentative schedule for the testing of the WMITS.

Project Test Plan

–2/9/2000 – 2/15/2000

This part is straightly theory stage. No any real actions will be performing.

System Testing

–3/6/2000 – 3/10/2000

Generate Testing Report

–3/7/2000 – 4/7/2000

3.0 Test Procedure

In this section we will describe the test procedures in detail.

3.1 Software (SCIs) to be tested

For detailed list of the software component items please refer to section 2.1 from the Test Specification document.

3.2 Testing Procedures

In this section we will try to describe over all software specification. We will describe the methods for all the different tests to be performed and will also declare the expected outputs.

2.2.1 Unit Testing

In this method of testing we will test the smallest unit of software called modules. We will be testing all the important paths to find any errors within the boundary of module. So here we will apply sort of white box search. We will be testing parts of the software rather than the entire software. The modules are as follows.

Login Window

We will make use several different names to log in to the system. We will use correct and incorrect User Names and Passwords to access the software and thus

access database. We will not be allowed to log in using incorrect passwords and error message will be shown. When correct password is presented we will be able to log in be able to next window (DEQ- Microsoft Visual Basic window). We will also test OK and Cancel buttons on this window by performing test above.

DEQ – Microsoft Visual Basic [Design] Window

This is the main window that we will use to access the database using Visual Basic. We will have several different drop down menu in this window. File, Facility, Inspection, approve, Reports, Maintenance and Help are the drop down menu that will be available in this window we will try to use all the menus and than different options available in each of the window.

File:

When file button is clicked user will be shown two choices.

Switch User

We will test if the Switch User button works. We need to make sure that the login screen is presented when this choice is selected. We will test this by choosing the Switch User and than by logging in as another user in Login Window.

Exit

We want to make sure that user is logged out or is able to exit out when Exit button is selected. We will test it by first logging in as a user and than By utilizing Exit button to exit out.

Facility:

When a Facility button is clicked we will be presented with Facility Screen. We can add or edit facility using the screen. We will test to make sure that the facility screen is presented when the button is clicked. We will do this by clicking on the button and making sure that the Facility Screen is presented.

Inspection:

When Inspection button is clicked user will be shown five choices.

Create/Modify

When click on this choice we will be presented with Inspection Step 1 Screen used to create or modify inspection. We want to make sure that this screen is presented when selection is made. We will test it by making the selection in drop down menu and by making sure that the correct window is presented.

File Results

When we select this choice we will be presented with File Result Window. We will test the button by clicking on it to go to File Result Window. We Want to make sure that the proper window is presented.

Get Approval

When we select this choice we will be presented with Get Approval Window. We will test the button by clicking on it to go to Get Approval Window. We want to make sure that the proper window is presented.

Print Letters

When we select this choice we will be presented with Print Letter Window. We will test the button by clicking on it to go to Print Letter Window. We want to make sure that the proper window is presented

View Schedule

When we select this choice we will be presented with View Schedule Window. We will test the button by clicking on it to go to View Schedule Window. We want to make sure that the proper window is presented

Approval:

This option is presented and works only for manger. We need to make sure that when someone other than inspector loges in the button is disabled. We will do this by logging in as both as inspector and as manger. We will make sure that manger is able to use the button while inspectors will not be able to use it. We also need to make sure that when selected by manager software will bring up window containing queued list of letters waiting to be approved.

Reports:**Print Staff Reports**

When we select this choice we will be presented with Report Window. We will test the button by clicking on it to go to Report Window. We want to make sure that the proper window is presented

Print Blank Checklists

When we select this choice we will be presented with Report Window. We will test the button by clicking on it to go to Report Window. We want to make sure that the proper window is presented

Efficiency Report

When we select this choice we will be presented with Report Window. We will test the button by clicking on it to go to Report Window. We want to make sure that the proper window is presented

Maintenance:**Checklist**

When we select this choice we will be presented with Checklist Window. We will test the button by clicking on it to go to Checklist Window. We want to make sure that the proper window is presented.

Letters:

When we select this choice we will be presented with Letters Window. We will test the button by clicking on it to go to Letters Window. We want to make sure that the proper window is presented

Users

When we select this choice we will be presented with Users Window. We will test the button by clicking on it to go to Users Window. We want to make sure that the proper window is presented

Options

When we select this choice we will be presented with Option Window. We will test the button by clicking on it to go to Option Window. We want to make sure that the proper window is presented

Help:

Content

When we select this choice we will be presented with Content Window (help contents). We will test the button by clicking on it to go to the Window. We want to make sure that the proper window is presented.

About

When we select this choice we will be presented with About Window (window with software Info). We will test the button by clicking on it to go to Option Window. We want to make sure that the proper window is presented.

Inspection Generator Window

This window is to allow inspector to generate the inspection ID before actually going out for inspection. We will have several windows to select options form and also buttons to test from.

Inspection ID box

This box automatically generates Inspection ID. We will make sure that the ID is automatically generated and that the ID dose not conflict with previous ID. We will test it by comparing the ID number with other automatically generated ID.

Facility ID box

This box is where user will select facility by clicking on facility from the facility list. We will make sure that all the facilities are available here and that correct facility information is associated with the facility.

Detail Window

We want to make sure that common information regarding that facility is shown in the Details window. We have to make sure that the information shown here is correct. To so we will use Facility ID known to us and this will bring up the information that we already know. Now we can check available information to us with information provided by software.

Scheduling

We want to make sure that all the boxes for the Scheduling information works properly.

We will insert certain Date, Time and Comments regarding inspection. After details are filled we will use schedule button, which will create the schedule of the inspection in database with Date and Time specified.

Follow UP Inspection box

test this by typing existing ID and than will try to view its content. This should work without any trouble.

Inspection Checklist Box

When we select or type in Inspection ID it will show the checklist that we had

also check if the spell checks function works in this window by typing the incorrect spelling this should give us error or will draw line under the wrong text.

Show in Letter Checkbox

We will test "Show in letter checkbox" to see if it adds the content in the letter generated. We will select the box and upon completion of the form will see if the content are correctly inserted in letter. This should work without any problems.

Compliance Status Box

We will have to test Compliance Status box to see if the selected compliance is inserted in to the letter. We will do this by choosing or highlighting an option from the Compliance status drop box menu and this will include the selected

Add Button

We will use Add button to see if the selected information from the form is added

~~Select Checklist Items~~ This should work without any troubles.

Remove Item

We will test if we are able to remove an item from inspected item box by using remove highlighted item button. We will make sure that only the selected item is removed in case the box contains more than one item in it. If the program is written correctly this should work.

Preview Button

We will test preview button to see if the letter is geeraovF3 1 the window to the right of the Generate checklist window. This window will contain the letter with listing of all the regulations selected. The letter should be geerated without any troubles.

Update Button

We will use update button to se if the changes made to the checklist are updated

~~Create Letter Button~~
We will check Create letter button to see if the final version of the letter is geeoovFed. We will test this by clicking on the button, which should give us final letter.

We will check the cancel button to see if the user can exit out of the window without any problems when the button is selected. This should work without any problem.

Help

We will check the help button to see if the help menu is made available to the user when help button is selected. The menu should popup without any problems when help button is selected.

Save Button

At the end will test the save button to see if all the data is correctly recorded and the generated letter is saved. This should work without any troubles.

Export Button

We will use Export button, which will pipe the letter to the word window. We will click the button while having letter in front of us and this will open the Microsoft word and letter will be piped in it. This should work without any troubles.

Inspection ID Explode Window

This window will come up when Explode button is selected form Choosing Inspection Window. It is created to allow user to find the Inspection ID.

Start At Box

We want to make sure that Start At box works, that is when first letter or character is typed of the Inspection ID we will get to the sorted list of all the Facility ID's starting with that character or letter.

Ok Button

We will test OK button that should take us back to Choosing Inspection Window with the Inspection ID number in the box. We will do this by selecting the ID and than clicking on the OK button.

Cancel

We will test Cancel button to make sure that when clicked it will cancel all the work done or will delete all the options selected. And will take us to main VB Window.

View Button

We will test view button which will able us to view the Inspection ID record. We will test this button by selecting ID and than clicking the View button that will present us with data. We will also test it the button by not selecting any Inspection ID and clicking View button. This will not bring us to any other window since no selection is made.

Creating Facility

This window is utilized to create a facility entry in the database if facility does not exist. User will have to simply complete from with text boxes and few drop down menus.



We will have the function that will check if the boxes are filled when we hit the save button after the completion of the form. This will not apply to the boxes that are in Mailing Information section or Owner Information section. If any other box from remaining section other than Fax Number and EPA ID is left blank error message will be shown and user will be told to fill the box to proceed ahead. We will test this by leaving some of the boxes empty that needs to be filled. We will leave EPA ID and Fax Number boxes empty. This should not give us any error messages.

Facility ID Box

Facility ID will box will contain the ID assigned to new facility. We will make sure that the box accepts all the characters and number in ID. We will test to make sure that the record is updated or added in the database. Also we will test to make sure that this box is not allowed to leave blank.

Explode Button

We will test Explode button (button with bomb), this should bring up the window that will allow us to search for the Facility ID.

Search Button

We will also test search button (One with folder), which should bring us to historical data window. We will test it by clicking on it and checking if it takes us to historical data window.

EPA ID Box

EPA ID will box will contain the EPA ID assigned to new facility. We will make sure that the box accepts all the characters and number in ID. We will test to make sure that the record is updated or added in the database. We will also make sure that the record is saved even when this box is left blank. We will do so by keep the box empty while clicking save.

Facility Type Box

Facility Type box will contain facility type. We will make sure that user is able to put in data in the box and that the data is correctly inserted in the database. We test by filling in box with some numbers and clicking save after all the data is filled. We will also leave the box empty and when we click save it will display error message saying the box is empty.

Name Box

Name box will contain the name of facility. We will allow name to be as long as it want to be. We will test several names to test this box and will make sure that data is updates in the database. We also want to make sure that this box is not allowed to be left blank. This can be done by leaving the box blank while clicking the save button which will give us error message.

Address Box



Address box will contain the address of the facility. We will allow address to be as long as it want to be. We will use several different addresses to test this box and will make sure that data is updates in the database. We also want to make sure that the box is not allowed to be left blank. Test can be done by leaving the box blank while clicking the save button which will give us error message.

Contact Box

Contact box will contain the name of contact person at the facility. We will allow name to be as long as it want to be. We will use several different names to test this box and will make sure that data is updates in the database. We also want to make sure that the box is not allowed to be left blank. Test can be done by leaving the box blank while clicking the save button which will give us error message.

City Box

City box will contain the name of the city that facility is at. We will allow city name to be as long as it want to be. We will use several different names to test this box and will make sure that data is updates in the database. We also want to make sure that the box is not allowed to be left blank. Test can be done by leaving the box blank while clicking the save button which will give us error message.

County Box

County box will contain the name of the county that facility is at. We will allow the name to be as long as it want to be. We will use several different names to test this box and will make sure that data is updates in the database. We also want to make sure that the box is not allowed to be left blank. Test can be done by leaving the box blank while clicking the save button which will give us error message.

Zip Code Box

Zip Code box will allow only five letter and no characters. If letters typed are less than five letters error will be shown. Error will also be shown if the character is typed in. We will test this by typing in number that contains five letters. We will also insert incorrect data that contains characters and less than five letters. This will give us an error message.

Fax Number Box

Fax number box will only contain numbers and no characters. If the character is typed in it will show error message. We will test the box by typing in correct and incorrect data. Incorrect data will give us error. Correct data should be stored in database.

Phone Number Box

Phone Number box will contain the phone number of the facility. We will allow the number to be 7 or 10 digits long. We will use several different numbers to test this box and will make sure that data is updates in the database. We also want to make sure that the box is not allowed to be left blank. Test can be done by leaving the box blank while clicking the save button which will give us error message.

We will use numbers that are more or less than seven digits but not more than or equal to 10 this will show us error message and will no allow us to proceed forward.

Comments Box

IN Comments box user can insert unlimited lines of comments. We will test to see if comment is correctly inserted into the form and also in database. We will test it by filling it with data and then by clicking save. This should work without any problems.

Save button will save the data in database. We will test the button to see if the data is actually saved and that database is updated.

Help Button

We will check the help button to see if the help menu is made available to the user when help button is selected. The menu should popup without any problems when help button is selected.

Save

At the end will test the save button to see if all the data is correctly recorded and the generated letter is saved. This should work without any troubles.

3.2.2 Integration Testing

In this method of testing we will implement the software at the client's location and will run it. So we will be testing the product on clients network. We will be following Top-Down model to test. We will start the test with login window and then main Visual Basic Window. After the Visual Basic Window we will be testing each and every sub component or functions of the software

We will be using Stubs to perform test. Stubs are dummy function that we will use to test the separate modules. Each of the module to be tested will have its own distinct stub, the stub will be created depending on the functions of the each software component. Stub will help us to test the product without actually having all the functions of the software.

As part of testing, will be looking for any signs of the collision between our software components and those of the clients. We want to make sure there is no confusion among the application on the network when they are running simultaneously.

Each of the software (SCIs) items will be the test case for the integration testing. So each form as whole will be a test case. We will be testing each and every form for all the errors that logically can occur in it.

We will install the software at the clients' site and will run it. We will have several different other applications open as well. This applications will be the once that have to interact with our software on normal bases. We will make sure that all the data is saved correctly and there is no loss of data or data base anomalies in the product.

At the end of the test all the results should be positive. All of the software components should work properly. In case we come across the errors we will record them in the error document and these errors will be fixed at the latter time.

3.2.3 Validation Testing

This test is performed to validate the software. The test is known as black box testing where the entire software will be created and will test all the component of the software together. We will be working with clients to show them the software and reach agreements about the completion or failure of the software.

We will have a person or a team of the inspectors forms the DEQ with us when we perform test. We perform a black box test where we will have software as whole with us and we will along with team from DEQ test the software. We will enter predetermined data with expected results. We will compare predicted results with those that software gives us and will determine if the software is valid or not.

Every button, tab or menus will be tested. We will also test Palm Size PC for it ability to correctly store the checklist and also to download or upload it to and from desktop. We will Test the correctness of the database and will make sure there are no errors regarding database updates.

We should not have in troubles with the software since we have already tested all the parts of the software before. Every thing here should work fine and we should be able to validate the software. We will be validated only if each and every thing works in the software. Any software in the error will force us to wait for the validation until we have fixed all the errors.

3.2.4 High-order Testing

High-order tests are combination of several different test methods. We will be performing four different tests on our software. These tests will be performed at software developers facilities and at clients site using the hardware that are available to clients.

- **Recovery Testing**

In this test method we are concerned with the software's ability to retrieve lost data or to make sure that software does not loose any data during the updating of the database. We will be mainly looking at transaction processing when we talk about the recovering testing.

In transaction protection we will be testing the software to make sure that when it saves any thing it will save all of it or none of it. This is necessary to avoid existence of corrupt data in database.

- **Security Testing**

In this testing section we are concerned about the security of the software. We will be testing the software to see if unauthorized users are able to access sensitive parts of the software. We have divided the security section in three stages.

1. Password Login

We will try to log in using invalid user name or valid user name and invalid password. We will also test the software to see if it allows access without any identification what so ever. We will also test the software so that password in not saved in any way within computer for others to view.

2. Modular Access

Our software identifies the user and allows him or her to access only certain modules. We will test to see if the software restricts unauthorized users from accessing certain modules of the software. In particular we want ti0 make sure that administrator cannot access modules for the manger. Also we want to make sure that other users cannot access modules that are only for the managers or administrator.

3. Priority Access

The software is setup so that inspectors can view each others work but can not update or delete it. We will test to make sure that inspector or managers in not able to delete the data entered by another user/inspector. We will do this by trying to delete the data entered using some other login name.

We should not have any problems with any sections of the security testing. Security measures are serious issue and we have made all attempts to keep it from any bug.

- **Stress Testing**

In this test method we are concerned with the software's ability to allow concurrent transaction. Too much of the work at the same time may cause system shutdown or frees. We what to test and to make sure that this does not happen. As test procedure we will try to create as much traffic for the software as we can by opening several applications concurrently. We will also monitor network traffic while performing this test to see if the clients network is able to sustain high traffic demands. We should not have any trouble in achieving our goal in this section of the test since our product is not at the large-scale software products and does not generate to much traffic either.

- **Performance Testing**

We will be testing the software to see if it meets the performance criteria set during design system specification. We will be performing these tests at the client's facility since we want to know if the hardware available to client will meet the performance criteria. Below are just some of the performance bounds we will be looking for.

Response time of search function

–Best Case Scenario -> Immediate

–Worst Case Scenario -> 3 seconds

Response time of browse function

- Next List of Records in 0.1 seconds
- Import export to WMITS Mobile
- Done within 0.5 second

3.3 Testing Resources and Staffing

We need to have large number of resources available to us in order for us to test the entire software properly. We will use help form several different people to be able to

Resources

-WMD Staff

We will take help of the DEQ staff of the Waste Management Devison to help us test the product. We are to allow DEQ staff member or members to test the product as part of validation testing. We will have the DEQ staff record any errors found in the software and will correct them before the delivery of the software.

-Palm-Size PC

We will use the available palm size PC to carry out test that will tell us if or if not PC is able to transfer data from and to the desktop computer. We will also test it to see if the data is stored in correct format and that data is not lost. We should not have any errors regarding palm size PC since we have carefully coded the PC to the software requirement.

-Bug Resource Reports

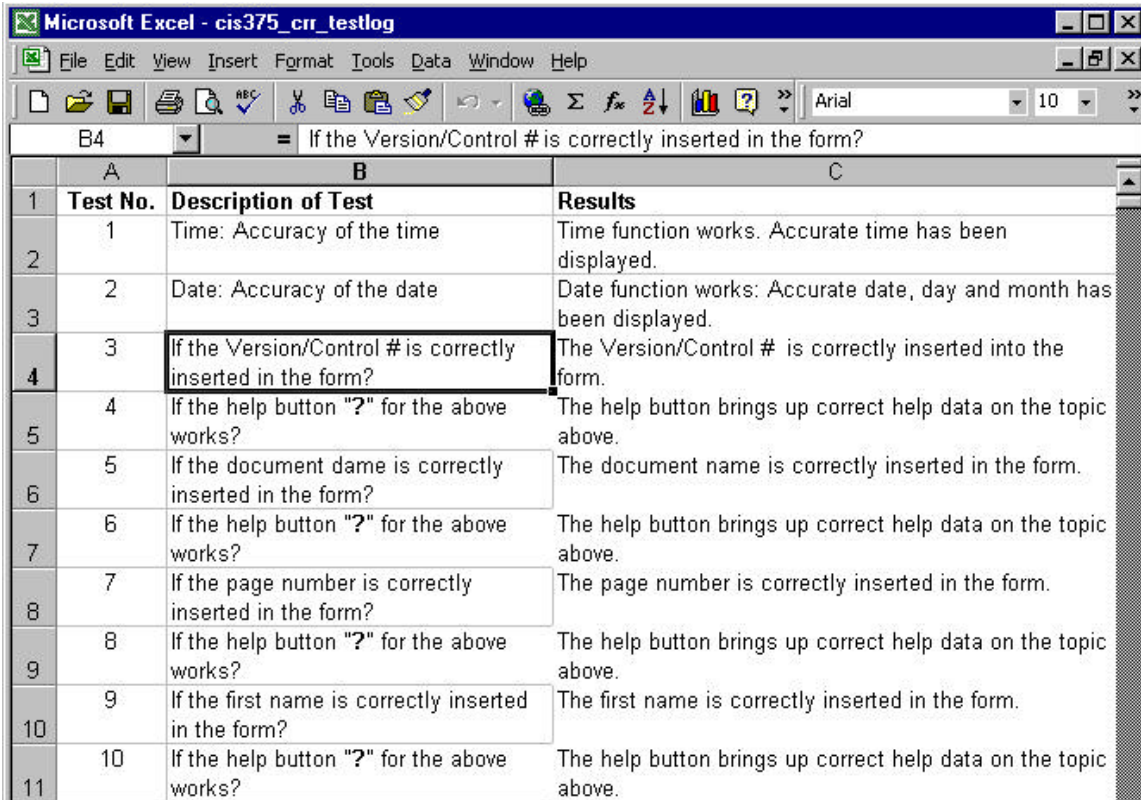
We will use Bug Resource Report where we will identify the bugs found during the testing and will try to identify the reasons for their occurrence. This will help teams that may work on the product latter to identify the soft spots for the bugs and will help them to come up with way to design products so that bugs are avoided.

Staffing

We have decided to use simple method for staffing people for the testing. Each program will test the components or functions created by him separately and will hand them over to lead tester. Lead tester will test each component and will make a note of the result in test result table. Once the product is completely developed we all the member of the software project team will test the software with combined effort. DEQ staff will also assist in testing the software.

3.4 Test Record Keeping and Log

We will use table created in excel to log all the test, describe them and to record the results of the tests. Below is the example of such table. The table will also be out test work product.



Microsoft Excel - cis375_crr_testlog

File Edit View Insert Format Tools Data Window Help

B4 = If the Version/Control # is correctly inserted in the form?

	A	B	C
1	Test No.	Description of Test	Results
2	1	Time: Accuracy of the time	Time function works. Accurate time has been displayed.
3	2	Date: Accuracy of the date	Date function works: Accurate date, day and month has been displayed.
4	3	If the Version/Control # is correctly inserted in the form?	The Version/Control # is correctly inserted into the form.
5	4	If the help button "?" for the above works?	The help button brings up correct help data on the topic above.
6	5	If the document name is correctly inserted in the form?	The document name is correctly inserted in the form.
7	6	If the help button "?" for the above works?	The help button brings up correct help data on the topic above.
8	7	If the page number is correctly inserted in the form?	The page number is correctly inserted in the form.
9	8	If the help button "?" for the above works?	The help button brings up correct help data on the topic above.
10	9	If the first name is correctly inserted in the form?	The first name is correctly inserted in the form.
11	10	If the help button "?" for the above works?	The help button brings up correct help data on the topic above.