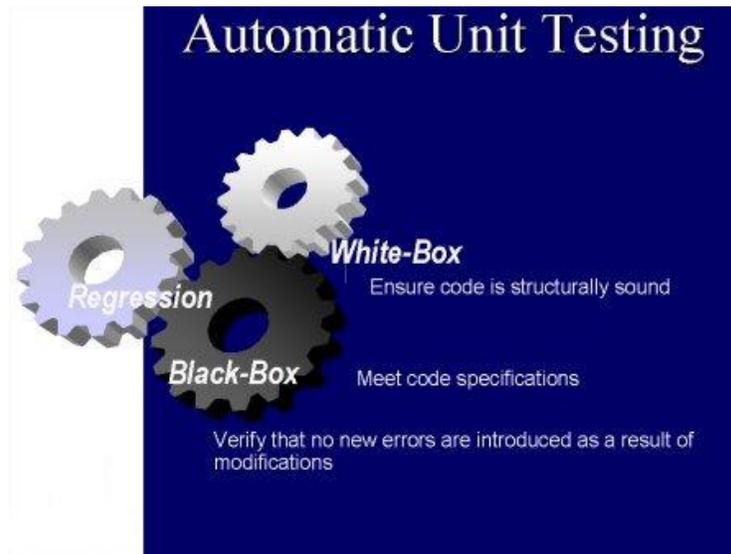




Address Management System

Black Box Test Cases



Author : Jamal Mohamed Kiyasudeen

Date : 19.12.2002

Version : 1.0

Approved By : Prof. Dr. Daniel Fischer

Signature : _____



Introduction:

Black Box Testing is testing without knowledge of the internal workings of the item being tested. For example, when black box testing is applied to software engineering, the tester would only know the "legal" inputs and what the expected outputs should be, but not how the program actually arrives at those outputs. It is because of this that black box testing can be considered testing with respect to the specifications, no other knowledge of the program is necessary. For this reason, the tester and the programmer can be independent of one another, avoiding programmer bias toward his own work.

The opposite of this would be glass box testing or white box testing, where test data are derived from direct examination of the code to be tested. For glass box testing, the test cases cannot be determined until the code has actually been written. Both of these testing techniques have advantages and disadvantages, but when combined, they help to ensure thorough testing of the product.

Advantages of Black Box Testing

- more effective on larger units of code than glass box testing
- tester needs no knowledge of implementation, including specific programming languages
- tester and programmer are independent of each other
- tests are done from a user's point of view
- will help to expose any ambiguities or inconsistencies in the specifications
- test cases can be designed as soon as the specifications are complete

Disadvantages of Black Box Testing

- only a small number of possible inputs can actually be tested, to test every possible input stream would take nearly forever

- without clear and concise specifications, test cases are hard to design
- there may be unnecessary repetition of test inputs if the tester is not informed of test cases the programmer has already tried
- may leave many program paths untested
- cannot be directed toward specific segments of code which may be very complex (and therefore more error prone)
- most testing related research has been directed toward glass box testing

Testing Strategies/Techniques

- black box testing should make use of randomly generated inputs (only a test range should be specified by the tester), to eliminate any guess work by the tester as to the methods of the function
- data outside of the specified input range should be tested to check the robustness of the program
- boundary cases should be tested (top and bottom of specified range) to make sure the highest and lowest allowable inputs produce proper output
- the number zero should be tested when numerical data is to be input
- stress testing should be performed (try to overload the program with inputs to see where it reaches its maximum capacity), especially with real time systems
- crash testing should be performed to see what it takes to bring the system down
- test monitoring tools should be used whenever possible to track which tests have already been performed and the outputs of these tests to avoid repetition and to aid in the software maintenance

In this technique, we do not use the code to determine a test suite; rather, knowing the problem that we're trying to solve, we come up with four types of test data:

1. Easy-to-compute data
2. Typical data
3. Boundary / extreme data
4. Bogus data

For example, suppose we are testing a function that uses the quadratic formula to determine the two roots of a second-degree polynomial ax^2+bx+c . For simplicity, assume that we are going to work only with real numbers, and print an error message if it turns out that the two roots are complex numbers (numbers involving the square root of a negative number).

We can come up with test data for each of the four cases, based on values of the polynomial's *discriminant* (b^2-4ac):

Easy data (discriminant is a perfect square):

A	B	C	Roots
1	2	1	-1, -1
1	3	2	-1, -2

Typical data (discriminant is positive):

A	B	C	Roots
1	4	1	-3.73205, -0.267949
2	4	1	-1.70711, -0.292893

Boundary / extreme data (discriminant is zero):

A	B	C	Roots
2	-4	2	1, 1
2	-8	8	2, 2

Bogus data (discriminant is negative, or **a** is zero):

a	B	c	Roots
1	1	1	square root of negative number
0	1	1	division by zero

As with glass-box testing, you should test your code with each set of test data. If the answers match, then your code passes the black-box test.

Why black-box testing is important?

Black-box testing ensures that methods in your class function properly and follow the specification. To perform unit-level black-box testing, you need to design a set of test cases that verify whether the class's specification is correctly implemented, then enter the test cases into a unit testing framework (or create your own framework). At least one test case should be created for each entry in the specification document; preferably, these test cases should test the various boundary conditions for each entry.

White Box Testing :

Also known as *glass box*, *structural*, *clear box* and *open box testing*. A software testing technique whereby explicit knowledge of the internal workings of the item being tested are used to select the test data. Unlike black box testing, white box testing uses specific knowledge of programming code to examine outputs. The test is accurate only if the tester knows what the program is supposed to do. He or she can then see if the program diverges from its intended goal. White box testing does not account for errors caused by omission, and all visible code must also be readable.

For a complete software examination, both white box and black box tests are required.

From the External Specification of our ADDRESS MANAGEMENT SYSTEM
the following are the Test Cases.

Name of the Window & Test Cases

Login Window

- ✓ Enter a user name with maximum alphanumeric characters. **Expected Output:-** Not a Valid user name username should be of maximum characters of (8 for example)
- ✓ Enter a correct username with incorrect password. **Expected Output:-** Incorrect Password.
- ✓ Enter a valid username and valid password. **Expected Output:-** The login window should proceed to the main window depend upon the username privilege.
- ✓ Click on the Login without entering password. **Expected Output:-** Please Enter your password.
- ✓ Click on the Login Button without entering anything in both the fields. **Expected Output:-** Please Enter your Username and Password.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-**Do you want to exit the application with “Yes” or “No”.
- ✓ Login with the same Username and Password in two systems. **Expected Output:-** User Already in use sorry access denied.
- ✓ Enter as “Guest” and try to view the Sales of a particular Customer. **Expected Output:-** Sorry, you have no access to this permission.
- ✓ Enter as “Administrator” and try to view the Sales of a particular Customer. **Expected Output:-** The window will show the sales window of the particular customer selected.

- ✓ If the Database server is down. **Expected Output:-** Database Server is down!!! Sorry datas cannot be accessed.

Main Window

- ✓ Enter some datas in the fields of main window and click on “Reset Button”. **Expected Output:-** It should clear all the datas entered and should wait for user’s entry again.
- ✓ Give the organisation name as “Maarketing” instead of “Marketing”. **Expected Output:-** “Did you mean **Marketing**, if so click on **Yes** to select Marketing.
- ✓ ADD a new Organisation(for example Organisation “A” from Country INDIA) and perform a Query operation with Value of the Country = INDIA). **Expected Output:-** Acceptable and should display all the organisation from INDIA including the latest Organisation added just.
- ✓ Type an Organisation name and it’s details then click on Reset Button and perform the Query operation for the Organisation name just entered. **Expected Result:-** Sorry! No matches found.
- ✓ Try editing and click on ADD twice. **Expected Output:-** Data Exists Overwrite(yes or no).
- ✓ Add a department and a contact person to the existing organisation and perform the Query Operation. **Expected Output:-** It should display the latest added department or the contact person depend upon the query condition.
- ✓ While the user try editing operation, if the timestamp differs from the last modification date and time. **Expected Output:-** “Timestamp differs from the last modification date and time, if you want to view the modified contents please press “Yes”.
- ✓ Delete an Organisation and try to edit the organisation by giving the name of the organisation. **Expected Output:-** No Organisation found.
- ✓ Give maximum number of characters for STREET and click on ADD. **Expected Output:-** The Street column should not exceed 15 Characters for example.
- ✓ Give maximum number of characters for CITY and click on ADD. **Expected Output:-** The CITY column should not exceed 10 for example.
- ✓ Give maximum numbers for ZIPCODE and click on ADD. **Expected Output:-** The ZIPCODE column should not exceed 5 numbers for example.
- ✓ Give negative values for for ZIPCODE and click on ADD. **Expected Output:-** The ZIPCODE column should not contain negative values.

- ✓ Give a string for ZIPCODE and click on ADD. **Expected Output:-** The ZIPCODE column should not contain a string value.
- ✓ Give maximum number of characters for COUNTRY and click on ADD. **Expected Output:-** The COUNTRY column should not exceed 10 characters for example.
- ✓ If we change the contact person from the pull down menu. **Expected Result:-** All the particulars like street, city, zipcode, country, phone, fax, email and home page should be changed.
- ✓ Give maximum numbers and characters for PHONE and click on ADD. **Expected Output:-** The PHONE field should not exceed 11 numbers without characters for example.
- ✓ Give negative numbers for PHONE and click on ADD. **Expected Output:-** The PHONE field should not be Negative.
- ✓ Give maximum numbers and characters for FAX and click on ADD. **Expected Output:-** The FAX field should not exceed 11 numbers without characters for example.
- ✓ Give negative numbers for FAX and click on ADD. **Expected Output:-** The FAX field should not be Negative.
- ✓ Give maximum numbers and characters for EMAIL and click on ADD. **Expected Output:-** The EMAIL field should not exceed 25 alphanumeric characters for example.
- ✓ Give maximum numbers and characters for HOME PAGE and click on ADD. **Expected Output:-** The HOME PAGE field should not exceed 30 alphanumeric characters for example.
- ✓ Enter 10 organisation with same data then perform a name search. **Expected Result:-** All names should be displayed.
- ✓ Click an organisation from the Result list the datas will be shown in the middle of the window then perform DELETE operation. **Expected Output:-** All the Departments and the contact persons will be deleted, do you want to continue ("Yes" or "No").
- ✓ Select an organisation and its comment window then try to edit the name of the Organisation. **Expected Output:-** Before editing do you want to close the comment window (Say "Yes" or "No")
- ✓ Select an organisation and its sales window then try to edit the name of the Organisation. **Expected Output:-** Before editing do you want to close the sales window (Say "Yes" or "No")

- ✓ Try to delete an organisation from the result list. **Expected Output:-** The organisation details displayed should gone and it should display the next organisation details.
- ✓ When the cursor is moved in the result window. **Expected Output:-** The corresponding details should be updated in the window.
- ✓ If we press Edit without editing anything. **Expected Output:-** Warning Message Nothing updated.
- ✓ If we perform query with wildcards. **Expected Output:-** The Result should be visible depending upon the wild card condition.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-**Do you want to exit the application with “Yes” or “No” and then it should display the login window.
- ✓ If we click on EXIT button. **Expected Output:-** The virtual machine in the task manager should not be visible for that window.

Administration Window

Creating “Guest” or “Normal” User.

- ✓ Create two username and password for “Guest” and “Normal” user and login as Guest and try to view the Organisations Monthly sales. **Expected Output:-** Sorry, you have to access permission.
- ✓ And login as Normal user and try to view the Organisations Monthly sales. **Expected Output:-** Acceptable and it should proceed to the Sales window.
- ✓ Enter some datas and click on Reset Button. **Expected Window:-** It should clear the contents entered.
- ✓ Give same user name and click on ADD. **Expected Output:-** Username Already exists try different Name.
- ✓ Give only two or three alphanumeric characters for password. **Expected Output:-** The password should be 6 to 8 characters.
- ✓ After Adding the data before resetting it click on add again. **Expected Output:-** Data Exists Overwrite(yes or no).
- ✓ The Firstname and Lastname of the same Department should not matches . **Expected Output:-** Data Already exists within the department try with little modifications.

- ✓ Same with Edit operation also if the data matches with the already existing datas. **Expected Output:-** Data Already exists try with little modifications.
- ✓ While filing the field “Department” the alphabets should matches exactly with already existing departments. Exacly Matches. **Expected Output:-** Acceptable, Otherwise **Expected Output:-** “Do you want to create a New Department”(Confirmation Message with “Yes” or “No”).
- ✓ If we move the cursor in the Result list. Expeceted Output:- The coresponding data should be shown
- ✓ If we try to click edit without editing the contents. **Expected Output:-** Warning message should be shown with “Nothing edited”.
- ✓ If we delete a data by pointing the cursor in the Result List. **Expected Output:-** The data should be deleted and it should show the next data’s in the respective fields.
- ✓ Query with small letters or one letter missing. **Expected Output:-** No Matches Found.
- ✓ Try “Query” with exact alphabets in the respective columns. **Expected Output:-** Acceptable.
- ✓ Query with wild cards. **Expected Output:-** It should display in the result list depending upon the wild cards condition.
- ✓ If we click CLOSE in the window. **Expected Output:-** It should close the window and the virtual machine running of that particular window and then should display the login window.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-** It should close the window by discarding the unsaved data and it should return to login window.

Comment Window

Click on Comments from the Main Window will show the Comment Window.

- ✓ Try to modify the **organisation name** and **date of the last modified** field, which has been selected from the Main Window. **Expected Output:-** Sorry, Access Denied. You can modify only the Comments below and click on Apply.

- ✓ Try to type more than 150 characters in the comment column and click on Apply. **Expected Output:-** Sorry the maximum limit is 100 characters(for example).
- ✓ Entering comments less than or equal to 100 characters. **Expected Output:-** Acceptable and it should update the current date in the database for the Last modified field.
- ✓ Without clicking apply view the comments again. **Expected Output:-** No changes in comments entered.
- ✓ Click apply after edition in the comments column and view the comments again. **Expected Output:-** Changes in comments.
 - ✓ If we click CLOSE in the window. **Expected Output:-** It should close the window and the virtual machine running of that particular window and then should display the login window.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-** It should close the window by discarding the unsaved data and it should return to parent window (i.e., Main window).

Sales Window

Assume that the month is September now.

- ✓ Click on Oct or Fourth Quarter and click Apply. **Expected Output:-** We are in September “TRY LATER”.
- ✓ Click on Jan or First Quarter **Expected Result:- Acceptable.**
 - ✓ If we click CLOSE in the window. **Expected Output:-** It should close the window and the virtual machine running of that particular window and then should display the login window.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-** It should close the window by discarding the unsaved data and it should return to parent window (i.e., Main window).

Update Window

After Relod button is clicked.

(DATE Test Case)

- ✓ Missing in proper location --- **Expected Output:** When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"
- ✓ Missing leading zero in day component (e.g.,06/MAR/1998 --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"
- ✓ Month name component is not a valid 3-character abbreviation --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"
- ✓ Day component is less than 1--- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and in red it should say "Date is Required"
- ✓ Day component is greater than the number of days in current month (Test for leap year could be made here) --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"
- ✓ Year component is less than 0 --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"

- ✓ Year component is greater than 2999 --- Expected Output: When the user is going to add the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Date is Required"
- ✓ Valid date of the form DD/MM/YYYY(German Standard) --- Expected Output: Acceptable

Time Test Case:

- ✓ Missing in proper location --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Missing leading zero in hour component (e.g.,1:10 vs. 01:10 --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Missing leading zero in minute component (e.g.,01:5 vs. 01:05 --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Enter more than 2 digits for hours or minutes component --- Expected Output: The text field will not allow it and therefore only the 2 digits per component will be submitted.
- ✓ Minute component is less than 0--- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Hour component is less than 1--- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"

- ✓ Minute component is greater than 60 --- Expected Output: When the user is going to add the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Hour component is greater than 24 --- Expected Output: When the user is going to save the data into the database the window will not be passed since it recognised an Error & return to the page and it should say "Time is Required"
- ✓ Minute component is between 0 and 60 --- Expected Output: Acceptable
- ✓ Hour component is between 1 and 24 --- Expected Output: Acceptable
- ✓ Valid time of the form HH:MM on a 24hour clock --- Expected Output: Acceptable

Reload Button:

- ✓ Click on Reload button without clicking Save button observe the changes. Expected:- No changes.
- ✓ Click on Reload button and click on Save button, observe the changes now. Expected:- Changes made are saved to the system.
- ✓ Enter the version number below the current version. **Expected Output:-** The version number should be greater than the current version.
- ✓ Enter a negative number for the version. **Expected Output:-** The version number should not be a negative number.
- ✓ Enter the version number as a string. **Expected Output:-** The version number should not be a string.
- ✓ Click on Cancel button before clicking Save operation. **Expected Output:-** Abort the window with no changes.

- ✓ Click on Cancel button after clicking Save operation. **Expected Output:-**
Abort the window with changes.
- ✓ Click on the “X” on the top left upper corner of the window:- **Expected Output:-** It should close the window and the virtual machine running in the system for that window.

These are the test cases for the Black Box Testing. Next we proceeding with Test procedures. Test Procedures will be released soon.