**Experiment 9:** **Apply the knowledge of test cases for the project using white box testing**.

**Learning Objective:** Students will able to create unit test cases

**Tools:**  Junit

**Theory:**

Software testing is an investigation conducted to provide stakeholders with information about the quality of the product or service under test. Software testing also provides an objective, independent view of the software to allow the business to appreciate and understand the risks of software implementation.

**Unit Testing:**

Unit testing focuses on the building blocks of the software system, that is, objects and subsystems. The specific candidates for unit testing are chosen from the object model and the system decomposition. In principle, all the objects developed during the development process should be tested, which is often not feasible because of time and budget constraints. The minimal set of objects to be tested should be the participating objects in the use cases. Subsystems should be tested after each of the objects and classes within that subsystem have been tested individually Unit testing focuses verification effort on the smallest unit of software design—the software component or module.The unit test is white-box oriented. . In Unit testing the following are tested,

1. The module interface is tested to ensure that information properly flows into and out of the program unit under test.
2. The local data structure is examined to ensure that data stored temporarily maintains its integrity.
3. Boundary conditions are tested to ensure that the module operates properly at boundaries established to limit or restrict processing.
4. All independent paths through the control structure are exercised to ensure that all statements in a module have been executed at least once.
5. And finally, all error handling paths are tested

Write a program to calculate the square of a number in the range 1-100

#include <stdio.h>

int main()

{

    int n, res;

    printf("Enter a number: ");

    scanf("%d", &n);

    if (n >= 1 && n <= 100)

    {

res = n \* n;

printf("\n Square of %d is %d\n", n, res);

    }

    else if (n<= 0 || n > 100)

printf("Beyond the range");

    return 0;

}

|  |  |  |
| --- | --- | --- |
| Sr no | Input | Output |
| 1 | -2 | Beyond the range |
| 2 | 0 | Beyond the range |
| 3 | 1 | Square of 1 is 1 |
| 4 | 100 | Square of 100 is 10000 |
| 5 | 101 | Beyond the range |
| 6 | 4 | Square of 4 is 16 |
| 7 | 62 | Square of 62 is 3844 |

Test Cases

Test case 1 : {I1 ,O1}

Test case 2 : {I2 ,O2}

Test case 3 : {I3, O3}

Test case 4 : {I4, O4}

Test case 5 : {I5, O5}

Test case 6 : {I6, O6}

Test case 7 : {I7, O7}

**Result:**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **Test ID** | **Test Name** | **Test Data** | **Input** | **Expected Output** | **Actual Output** | **Test Status** |
| **TC\_1** | User Login | User will login using Login ID and Password | Login ID: ABC  Password: abc@123 | Successfully Logged in | Successfully Logged in | Pass |
| **TC\_2** | Home Page | Home page should be displayed | - | Home page display | Home page display | Pass |
| **TC\_3** | Balance Enquiry | Balance: Check the balance  Display: Display the Cash | Click on Balance Enquiry tab | Balance page opens | Balance page opens | Pass |
| **TC\_4** | Withdrawal | Users should be able to withdraw money | Click on withdraw tab  Display1: Limit of per day transaction (<25000)  Display2: Amount should be between 500 to 25000 | Withdrawal successfull | Withdrawal successfull | Pass |
| **TC\_5** | Fund Transfer | Users should be able to transfer fund | Enter account details to whom the transfer is to be made | Transfer succesfull | Transfer succesfull | Pass |
| **TC\_6** | Home Page | Return to home page | Click on Home button/tab | Home page display | Home page display | Pass |
| **TC\_7** | Account Log | Access the monetary transaction in account log | Account log button/tab | Display account log | Display account log | Pass |
| **TC\_8** | Customer Details | Go to customer detail page | Click on Customer button/tab | Displays customer details | Displays customer details | Pass |
| **TC\_9** | Update  Customer  Details | Update customer and their details in the log | -Enter new details of the the user | Log created | Log not created | Fail |
| **TC\_10** | User Logout | Logout from the system | Click Logout button | Successfully logged out | Successfully logged out | Pass |

**Learning Outcomes:** Students should have the ability to

**LO1:** Students will be able to understand Software Testing Concepts and the various Software standards.

**LO2**: to test a software with the help ofJunit

**LO3**: create test cases

**LO4**: To understand different tools for testing

**Outcomes:** Upon completion of the course students will be able to write test cases for the project.

**Conclusion:**

Thus we designed a few test Cases for our chosen project.

**Viva Questions:**

1. **What is difference between git and Github?**
2. **What is version control? Why is it required?**
3. **What are other tools for version control?**
4. **What are different types of version control?**

For Faculty Use

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Correction Parameters** | **Formative Assessment [40%]** | **Timely completion of Practical [ 40%]** | **Attendance / Learning Attitude [20%]** |  |
| **Marks Obtained** |  |  |  |  |