**Name:** PATEL UMME AIMAN ABDUL WAHID..

**Roll No:** 22DCO05 **Batch:** 1 **Subject:** SE

**Experiment No. 10**

**Aim:** To write test cases for black box testing.

**Theory:**

Black Box Testing is a software testing method in which the functionalities of software applications are tested without having knowledge of internal code structure, implementation details and internal paths. Black Box Testing mainly focuses on input and output of software applications and it is entirely based on software requirements and specifications. It is also known as Behavioral Testing. Test cases are created considering the specification of the requirements. These test cases are generally created from working descriptions of the software including requirements, design parameters, and other specifications. For the testing, the test designer selects both positive test scenario by taking valid input values and adverse test scenario by taking invalid input values to determine the correct output. Test cases are mainly designed for functional testing but can also be used for non-functional testing. Test cases are designed by the testing team, there is not any involvement of the development team of software.

The different types of Black Box Testing are:

**Equivalence Partitioning:**

Divides the input domain into classes or partitions and selects a representative value from each class for testing.

**Boundary Value Analysis (BVA):**

Focuses on testing the boundaries or extreme values of input ranges, as these often have a higher likelihood of causing errors.

**Decision Table Testing:**

Defines different conditions and actions in a table format, allowing testers to derive test cases for different combinations of conditions and actions.

**Test Cases for AIKTC Training & Placement Website:**

| **Test**  **Code** | **Test Case** | **Test Steps** | **Expected Result** | **Actual Result** | **Pass/Fail** |
| --- | --- | --- | --- | --- | --- |
| Test 1 | Test System  Performance  Under Various  Loads | Simulate a low load scenario  (e.g., 10 concurrent users). | Low load:  Response times within  an acceptable range. | Low load:  Response times within  an acceptable range. | Pass |
| Simulate a moderate  load scenario  (e.g., 50 concurrent users). | High load:  Response times increase but remain manageable. | High load:  Response times  increased but remained manageable. |
| Test 2 | Test the system's response to  unexpected inputs and errors | Submit a form with  incomplete/ missing  mandatory fields. | Error message displayed for  incomplete/mi ssing data. | Error message displayed for  incomplete/mi ssing data. | Pass |
| Submit a request with  invalid URL or endpoint. | Error message displayed for invalid URL. | Error message displayed for invalid URL. |
| Test 3 | Verify Correct  Implementation of Authentication and  Authorization  Mechanisms | Attempt to access a  protected resource without  proper  authenticatio  n. | Access is denied without proper  authentication. | Access denied without proper authentication. | Pass |
| Use valid  credentials for an  authorized user. | Authentication and access are both successful. | Authentication and access both  successful as expected. |

**Conclusion:**

In this experiment, we undertook a comprehensive black-box testing approach to evaluate the functionality, usability, and robustness of the fundraising Website for education .The objective was to ensure the platform meets the expected requirements and offers a seamless experience to users.