

SNS COLLEGE OF ENGINEERING

Kurumbapalayam(Po), Coimbatore – 641 107 Accredited by NAAC-UGC with 'A' Grade Approved by AICTE, Recognized by UGC & Affiliated to Anna University, Chennai

Department of Artificial Intelligence and Data Science

23ITT203 Object Oriented Software Engineering

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED SOFTWARE ENGINEERING/SNSCE

4/9/2025





Black Box Testing



SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED SOFTWARE ENGINEERING/SNSCE





What is Black box Testing?

Definition:

- Black Box Testing is a software testing technique in which the internal structure, design, or code of the program is not known to the tester.
- Testing is done based on inputs and expected outputs.

Why is it Called "Black Box"?

The system is like a black box for the tester.

The tester knows only:

Inputs provided to the software

Expected outputs from the software

The internal logic or structure is hidden (like inside a black box).

4/9/2025

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED SOFTWARE ENGINEERING/SNSCE





Objective of Black Box Testing

- To verify the functionality of the software.
- To ensure that software works as per the user requirements.
- To detect errors in: \bullet
 - Incorrect functionalities •
 - Missing functions
 - Interface errors \bullet
 - Performance errors





Characteristics of Black Box Testing

- No knowledge of internal code \bullet
- Focus on input/output behavior \bullet
- Conducted by Testers \bullet
- Suitable for Functional Testing \bullet
- User-centric approach lacksquare





Process of Black Box Testing

- Step 1: Understand the requirements & specifications \bullet
- Step 2: Identify test inputs (valid & invalid) \bullet
- Step 3: Determine expected outputs
- Step 4: Execute test cases •
- Step 5: Compare actual output with expected output
- Step 6: Report defects (if any)





Techniques Used in Black Box Testing

Technique	Description
Equivalence Partitioning	Divide inputs into groups & test one valu
Boundary Value Analysis	Test values at boundary edges like min, m
Decision Table Testing	Used for systems with complex business l
State Transition Testing	Used when system behavior changes base
Error Guessing	Based on experience of tester to guess po

4/9/2025

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED **SOFTWARE ENGINEERING/SNSCE**





e from each group

ax, just inside/outside limits

ogic/rules

ed on previous states

ossible errors



1. Equivalence Partitioning: Divides the input data into equivalent partitions, with each partition being regarded the same by the program. Testing one representative from each partition is usually enough to cover all potential scenarios.

Example: For a form that accepts age input between 18 and 65, equivalence partitions might include: Valid partition: 18-65 (e.g., age 25)

Invalid partition: Below 18 (e.g., age 15)

Invalid partition: Above 65 (e.g., age 70)





2. Boundary Value Analysis: Tests the bounds of input ranges, as errors frequently arise on the edge of input limits.

Example: For an input field that accepts values from 1 to 100, boundary values would include: Lower boundary: 1

Just below lower boundary: 0

Just above upper boundary: 101

Upper boundary: 100





3. Decision Table Testing: A decision table is used to represent and test different combinations of inputs and predicted outcomes. This method is effective for testing systems that involve several conditions and actions.

Example: For a loan application system with conditions like credit score (high/low) and income (above/below threshold), a decision table might include:

Credit Score	Income	Lo
High	Above Threshold	Y
High	Below Threshold	Y
Low	Above Threshold	Ν
Low	Below Threshold	Ν

4/9/2025

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED **SOFTWARE ENGINEERING/SNSCE**

an Approved (Y/N)



4. State Transition Testing: Tests the system's behaviour in various states and transitions between them. It ensures that the system functions properly when transitioning from one state to another. **Example:** For a user login system, states might include:

Logged Out

- Logged In
- Suspended

Transitions would be:

From Logged Out to Logged In (successful login) From Logged In to Suspended (suspend account) From Suspended to Logged Out (logout from suspended state)





5. Error Guessing:

Error Guessing is a black box testing technique where testers predict possible mistakes based on experience. It helps to check how software behaves for wrong or unexpected inputs. Example (2 Lines)

Example: In an ATM, if a user enters the wrong PIN like "123A" or leaves it blank, the system should show an error message like "Invalid PIN".







5. Error Guessing:

Error Guessing is a black box testing technique where testers predict possible mistakes based on experience. It helps to check how software behaves for wrong or unexpected inputs. Example (2 Lines)

Example: In an ATM, if a user enters the wrong PIN like "123A" or leaves it blank, the system should show an error message like "Invalid PIN".







Example of Black Box Testing

Example: Login Page Testing Scenario: A login page accepts: Username (Required) Password (Required)

Test Cases:

Test Case No.	Input	Ex
1	Valid Username & Valid Password	Sı
2	Invalid Username & Valid Password	Eı
3	Valid Username & Invalid Password	Eı
4	Blank Username & Password	P
5	Special Characters in Username	Eı
4/9/2025	SOWMIYA R/AP/AI&DS/23ITT203 OBJ SOFTWARE ENGINEERING/SI	ECT NSC



xpected Output

- uccessful Login
- rror Message
- rror Message
- rompt to Enter Values
- rror Message
- ORIENTED
- E



Tools Used for Black Box Testing

- Selenium
- QTP (UFT) ullet
- TestComplete \bullet
- Ranorex lacksquare
- Appium (Mobile Testing) lacksquare





Advantages of Black Box Testing

- Tester does not need programming knowledge.
- Helps in detecting missing functionalities. \bullet
- Effective for large systems.
- Tests from user's perspective.
- Suitable for acceptance testing. •





Disadvantages of Black Box Testing

- Limited coverage (cannot test internal paths).
- Difficult to design test cases without knowing code. \bullet
- Chances of missing logical errors inside code. \bullet
- Repetitive testing of same functionalities.





Difference Between Black Box Testing & White Box Testing

Black Box Testing

No knowledge of internal code

Focus on functionality

Done by Testers

Functional Testing

4/9/2025

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED **SOFTWARE ENGINEERING/SNSCE**



White Box Testing

Full knowledge of code

Focus on logic & structure

Done by Developers

Structural Testing



Real-life Example for Easy Understanding

Example: ATM Machine

Input

Correct PIN

Incorrect PIN

No Card Inserted

Expected Output Access to Account

Error Message

Prompt to Insert Card

Tester doesn't know the internal program of ATM.

4/9/2025

SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED **SOFTWARE ENGINEERING/SNSCE**









SOWMIYA R/AP/AI&DS/23ITT203 OBJECT ORIENTED **SOFTWARE ENGINEERING/SNSCE**



