BLACKBOX TESTING:
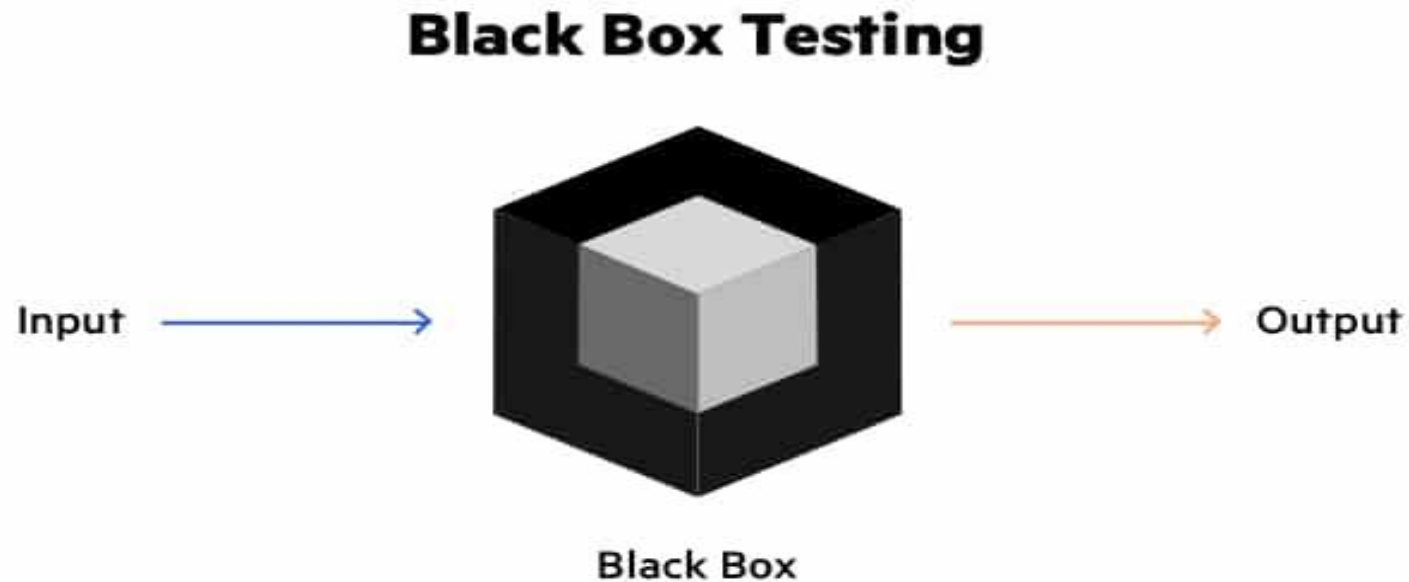
Black box testing is a technique of software testing which examines the functionality of software without peering into its internal structure or coding. The primary source of black box testing is a specification of requirements that is stated by the customer.

Black Box Testing

Input → Black Box → Output

Black Box

BLACKBOX TECHNIQUES:
BOUNDARY VALUE ANALYSIS:

Boundary value analysis is one of the widely used case design technique for black box testing. It is used to test boundary values because the input values near the boundary have higher chances of error.

Whenever we do the testing by boundary value analysis, the tester focuses on, while entering boundary value whether the software is producing correct output or not.

Fig. Boundary Value Analysis -

Example - As per the requirement, the textbox should accept the values from 4 to 20

Minimum Boundary Value

Maximum Boundary Value

. . . 2 3 **4** 5, 6 7 . . .

. . . 17 18 19 **20** 21 22 . . .

Invalid values   **Valid values**   **Invalid values**

Test Cases -

	Partitions	Input Value	Expected Result
Negative Test	Below Minimum Boundary Value	3	Fail
	Minimum Boundary Value	4	Pass
	Above Minimum Boundary Value	5	Pass
	Below Maximum Boundary Value	19	Pass
	Maximum Boundary Value	20	Pass
Negative Test	Above Maximum Boundary Value	21	Fail

EQUIVALENCE PARTITIONING:

Equivalence partitioning is a technique of software testing in which input data is divided into partitions of valid and invalid values, and it is mandatory that all partitions must exhibit the same behavior. If a condition of one partition is true, then the condition of another equal partition must also be true, and if a condition of one partition is false, then the condition of another equal partition must also be false. The principle of equivalence partitioning is, test cases should be designed to cover each partition at least once. Each value of every equal partition must exhibit the same behavior as other.

The equivalence partitions are derived from requirements and specifications of the software. The advantage of this approach is, it helps to reduce the time of testing due to a smaller number of test cases from infinite to finite. It is applicable at all levels of the testing process.

INVALID	INVALID	VALID	VALID
1 Test case	2 Test case	3 Test case	
DIGITS >=7	DIGITS<=5	DIGITS = 6	DIGITS = 6
93847262	9845	456234	451483

DECISION TABLE:

Decision table technique is one of the widely used case design techniques for black box testing. This is a systematic approach where various input combinations and their respective system behavior are captured in a tabular form.

Decision table technique is appropriate for the functions that have a logical relationship between two and more than two inputs.

This technique is related to the correct combination of inputs and determines the result of various combinations of input. To design the test cases by decision table technique, we need to consider conditions as input and actions as output.

Conditions	R1	R2	R3
Withdrawal Amount <= Balance	T	F	F
Credit granted	-	T	F
Actions			
Withdrawal granted	T	T	F

STATE TRANSITION TECHNIQUE:

The general meaning of state transition is, different forms of the same situation, and according to the meaning, the state transition method does the same. It is used to capture the behavior of the software application when different input values are given to the same function.

STATE TRANSITION
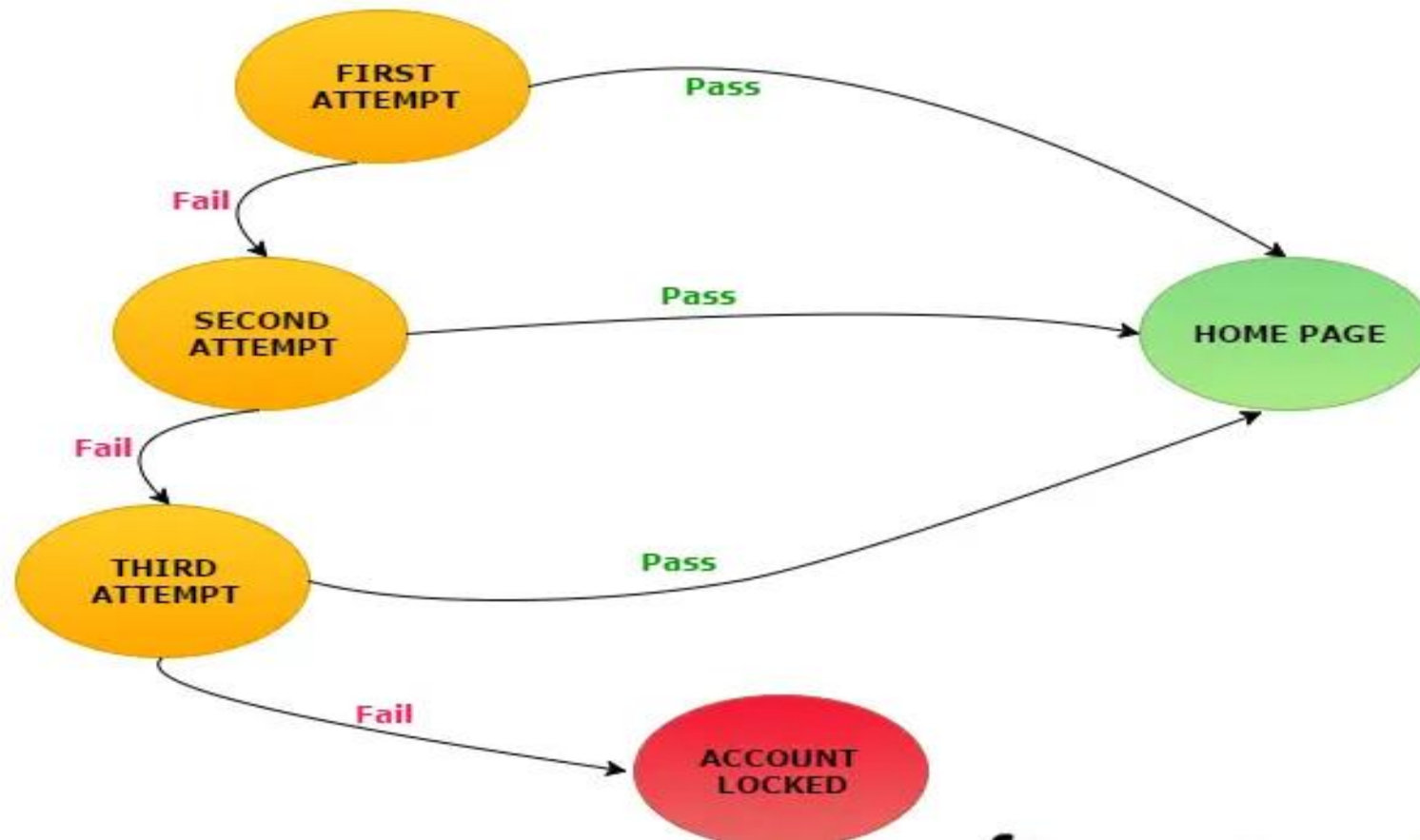
ALL PAIR TECHNIQUE:

All-pairs testing technique is also known as pairwise testing. It is used to test all the possible discrete combinations of values. This combinational method is used for testing the application that uses checkbox input, radio button input (radio button is used when you have to select only one option for example when you select gender male or female, you can select only one option), list box, text box, etc.

Product	Order category	Location	Registration number	Order type	Order time
BMW	Buy	Delhi	Valid		
BMW	Sell	Mumbai	Invalid		
Audi	Buy	Mumbai	Valid		
Audi	Sell	Delhi	Invalid		
Mercedes	Buy	Delhi	Valid		
Mercedes	Sell	Mumbai	Invalid		

CAUSE-EFFECT TECHNIQUES:

Cause-effect graph comes under the black box testing technique which underlines the relationship between a given result and all the factors affecting the result. It is used to write dynamic test cases.

The dynamic test cases are used when code works dynamically based on user input. For example, while using email account, on entering valid email, the system accepts it but, when you enter invalid email, it throws an error message. In this technique, the input conditions are assigned with causes and the result of these input conditions with effects.

Cause-Effect graph technique is based on a collection of requirements and used to determine minimum possible test cases which can cover a maximum test area of the software.

The main advantage of cause-effect graph testing is, it reduces the time of test execution and cost.

This technique aims to reduce the number of test cases but still covers all necessary test cases with maximum coverage to achieve the desired application quality.

Cause-Effect graph technique converts the requirements specification into a logical relationship between the input and output conditions by using logical operators like AND, OR and NOT.
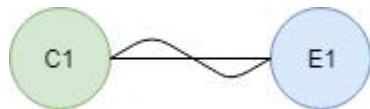
AND - E1 is an effect and C1 and C2 are the causes. If both C1 and C2 are true, then effect E1 will be true.
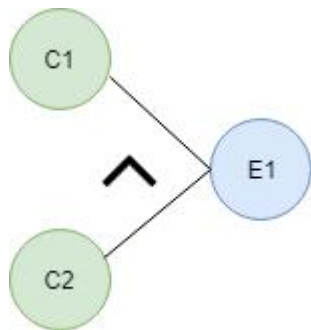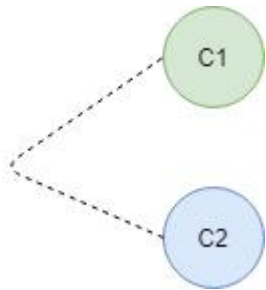
OR - If any cause from C1 and C2 is true, then effect E1 will be true.
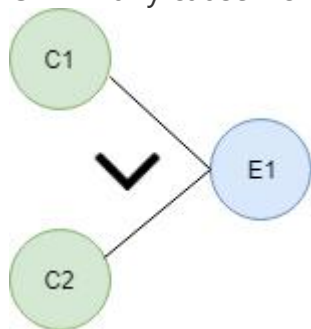
NOT - If cause C1 is false, then effect E1 will be true.

Mutually Exclusive - When only one cause is true.

ERROR GUESSING TECHNIQUES:

The test case design technique or methods or approaches that need to be followed by every test engineer while writing the test cases to achieve the maximum test coverage. If we follow the test case design technique, then it became process-oriented rather than person-oriented.

The test case design technique ensures that all the possible values that are both positive and negative are required for the testing purposes. In software testing, we have three different test case design techniques which are as follows:

- o Error Guessing
- o Equivalence Partitioning
- o Boundary Value Analysis[BVA]
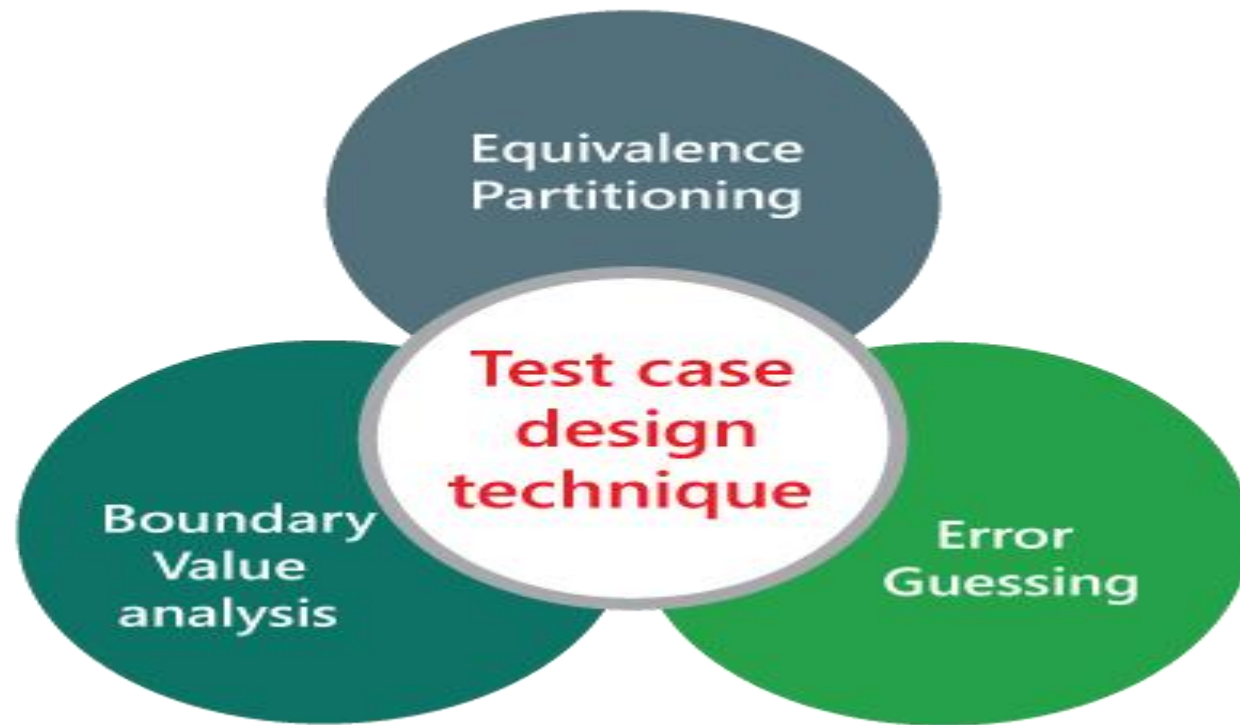
USER CASE TECHNIQUES:

The use case is functional testing of the black box testing used to identify the test cases from the beginning to the end of the system as per the usage of the system. By using this technique, the test team creates a test scenario that can exercise the entire software based on the functionality of each function from start to end.