

What is Black Box Testing?

Black-box testing is a type of testing that focuses on the external behavior of the software application without considering its internal implementation details.

Black box testing involves testing a system with no prior knowledge of its internal workings. A tester provides an input, and observes the output generated by the system under test. This makes it possible identify how the system responds to expected and unexpected user actions, its response time, usability issues and reliability issues.

Characteristics of Black Box Testing

- Focuses on the **input-output** behavior of the software.
- Does not require knowledge of the internal code structure or algorithms.
- Involves checking functionality, usability, performance, security, and compliance.
- Can be used at various levels: unit, integration, system, and acceptance testing.

Black Box Testing Techniques

- Boundary Value Analysis
- Equivalence partitioning
- State Transition Testing
- Decision Table Testing
- Graph-Based Testing
- Error Guessing Technique

Test The Online Appointment System Using Black Box Testing Techniques

Test Type	Scenario	Test Case	Input	Expected Output
-----------	----------	-----------	-------	-----------------

Test Type	Scenario	Test Case	Input	Expected Output
Functionality Testing	Book an appointment successfully	Book an appointment with valid details	Valid date, time, service, and user info	Appointment successfully booked with confirmation displayed
		Book an appointment with invalid details	Past date or missing required fields	Appropriate error message shown (e.g., "Invalid date")
Boundary Value Testing	Book appointment at boundary hours	Test appointment booking at 9:00 AM and 5:00 PM	9:00 AM, 5:00 PM	Appointment successfully booked at boundary times
		Book appointment outside working hours	8:00 AM, 6:00 PM	Error message shown ("Outside working hours")
Equivalence Partitioning	Book appointment for different user types	Test with regular user, admin, and guest	User type: Regular user, Admin, Guest	Regular user and admin can book; guest gets "Unauthorized"
		Book with invalid user	Invalid login	Access denied or unauthorized error message
Usability Testing	Easy navigation and appointment booking	Test ease of booking appointment through UI	User clicks through the appointment booking interface	User easily navigates and books appointment successfully
Performance Testing	Handle multiple users booking simultaneously	Simulate multiple users booking	50 users booking at the same time	System handles load without slowdown or crash

Test Type	Scenario	Test Case	Input	Expected Output
		appointments		
Security Testing	Prevent unauthorized access to system data	Attempt to access restricted user data without login	SQL injection or unauthorized access attempt	System blocks access and prevents data leakage
		Book appointment without authorization	Guest tries to book without logging in	Error message: "Please login to book an appointment"
Error Handling Testing	Handle invalid input formats	Enter incorrect date format for booking	Date: "25-10-2024"	Error message: "Invalid date format"
		Enter invalid phone number format	Phone: "123ABCD"	Error message: "Invalid phone number"
Compatibility Testing	Check functionality across devices and browsers	Test on various devices and browsers	Devices: mobile, tablet, desktop; Browsers: Chrome, Safari	System works correctly across devices and browsers