



**NATIONAL UNIVERSITY OF MODERN
LANGUAGES**

Subject:

Software Quality Eng.

Member Names:

M.Ehtasham

Ashir Ali

Saad Fasih

Shakeela Batool

Program:

BSSE 6th Afternoon

Submitted To:

Dr. Sumaira Nazir

Introduction:

☉ Test Generation:

The process of creating test cases to evaluate software functionality, reliability, and performance.

☉ Test Execution:

The act of running these test cases to identify defects or issues within the software.

Why Test Generation?

Purpose:

To detect defects, validate functionality, ensure code reliability, and maintain software quality.

- **Comprehensive Test Coverage:**

Discuss the importance of covering various scenarios to minimize bugs

- **Early Bug Identification:**

Catches bugs in early development stages.

- **Functional Validation:**

Ensures each component works as intended.

Importance

- **Quality Assurance:**

Testing ensures that the software meets specified requirements and functions correctly, enhancing its overall quality.

- **Bugs Identification:**

Helps in identifying and fixing defects or bugs early in the development process, reducing potential risks in the later stages.

- **Enhanced User Experience:**

Ensures a seamless and error-free experience for end-users, increasing customer satisfaction.

- **Cost-Efficiency:**

Identifying and fixing issues early saves time and resources

Test Generation Techniques

Automated Test Generation:

Utilizing tools and scripts to create test cases efficiently

- Model-Based Testing
- Random Testing

Manual Test Generation:

Hands-on creation of test cases by human testers.

- Exploratory Testing
- Boundary Value Analysis

Test Case Generation

- ⦿ Understanding Use Cases
- ⦿ Identifying Test Scenarios
- ⦿ Creating Test Cases
- ⦿ Executing Test Cases

Examples:

User Registration

Test Case 1: Verify User Registration Process

Test Steps:

- Enter valid credentials (email, password, personal info).

- Click on the "Register" button.

Expected Outcome: User should receive a confirmation message and be logged into their new account.

Test Case 2: Test Invalid Registration Inputs

Test Steps:

- Enter an invalid email format.
- Use a weak password (less than 8 characters).
- Leave required fields empty.

Expected Outcome: System should prompt users with appropriate error messages for each invalid input.

Booking a Travel Package

Test Case 1: Successful Booking

Test Steps:

- Login with valid credentials.
- Select a travel package.
- Fill in necessary booking details (dates, number of travelers).
- Proceed with the booking.

Expected Outcome: User should receive a booking confirmation and be redirected to a booking summary page.

Test Case 2: Invalid Booking Date

Test Steps:

- Attempt to book a package for a past date or beyond available dates.
- Confirm the booking.

Expected Outcome: User should receive an error message indicating invalid booking dates.

Reviewing a Travel Package

Test Case : Adding a Review

Test Steps:

- Log in and access a completed travel package.
- Write a review, rate the experience, and add comments.
- Submit the review.

Expected Outcome: The review should be successfully added and displayed alongside the package details.

Test Case 8: Editing a Review

Test Steps:

- Log in and access a previously submitted review.
- Attempt to edit the existing review/comments.
- Save changes and verify.
- **Expected Outcome:** The changes should be saved successfully and reflected in the review section.

Integration of Weather Updates

Test Case : Weather Update Display

Test Steps:

- Select a travel package that includes weather updates.
- Verify if the weather forecast for the destination is displayed accurately.

Expected Outcome: The app should showcase accurate weather forecasts for the specified destination for the next 7 days.

