Introduction To Software Engineering

Professor: Dr. Waqar Aziz

Assignment no 4

Testing

Prepared By Shehraz Sarwar

Testing Approach

Types of Testing:

System testing: System testing ensures that the entire system functions work as expected since my software contains a number of components that are connected with each other, such as room selection, availability check, payment process, verification process, etc.

Unit testing: In order to check that the software in each unit (like functions) works as intended, it focuses on testing each and individual unit of the software, which helps us to ensure each unit like functions or classes works as intended.

Component testing: In Component testing, we test larger components of the software and those components that are interconnected.

Type of Strategy: Black Box and Dynamic Testing

Reasons:

Blackbox Testing: Blackbox testing is appropriate because the focus is on testing functionality without knowing the internal structure of the software.

Dynamic Testing: As I want to observe the software behavior, so it's better to choose this strategy as it involves execution of the software. It is suitable for testing functionalities such as room availability display, user selection, and payment processing.

Test Cases

Test Case 1: Reservation Selection

Input: The user selects the 'Reservation' option.

Expected output: available rooms for reservation are displayed by the system.

Test Case 2: Room Availability Display

Input: The user chooses floor level

Expected output: The available rooms on the selected floor are displayed by the system.

Test case 3: User Selection and Preferences

Input: The user selects the room and sets preferences.

Expected output: User preferences for the chosen room are being recorded by the system.

Test Case 4: Check Availability of Rooms

Input: The user does room selection.

Expected output: The system verifies the selected room availability.

Test Case 5: Personal Details Submission

Input: The user provides his/her personal details.

Expected Output: If room is available, the system records the user's personal details, if room is unavailable, the system prompts the user to select a room again.

Test Case 6: Payment Process

Input: The user provides the payment details.

Expected output: if a user is booking, the system processes the payment and if reservations, the system asks the user to pay by the due date.

Test Case 7: Payment Verification and Room Assignment

Input: The system does payment verification.

Expected output: If the verification is successful, the system assigns the room to the user.