MEKELLE UNIVERSITY

ETHIOPIAN INSTITUTE OF TECHNOLOGY - MEKELLE

SCHOOL OF COMPUTING

DEPARTMENT OF SOFTWARE ENGINEERING

Software Testing and Quality Assurance:

Assignment on : Test Design

Name	ID
MIKYAS HAFTAMU	UGR/170241/12

SUBMITTED TO: Inst. Mesele

SUBMITION DATE: Jan 18 2025

2.2 **Test Design**

Test design is the process of creating detailed test cases, test data, and test scripts based on the test conditions derived during analysis.

2.2.1 Objectives

- Create test cases to thoroughly validate the identified conditions in the payment form, ensuring all fields function as intended.
- Ensure that all test cases are detailed, providing clear instructions and expected outcomes for reproducibility.
- Define and maintain a traceability matrix to link test cases to specific requirements, ensuring complete coverage.
- Specify and prepare all necessary test data to support the execution of each test case.
- Assess the user interface and overall user experience to ensure the application is user-friendly and accessible.
- Ensure that the system correctly handles invalid inputs and displays appropriate error messages.
- Identify and prepare test cases suitable for automation to enhance efficiency and consistency in testing.

2.2.2 Test Cases

2.2.2.1 Test Case 1: First **Name Validation**

Test Case ID	**TC001**
Description	Verify that the first name field accepts valid input.
Preconditions	User is on the payment form page.
Test Steps	1. Enter a valid first name. 2. Leave the field empty. 3. Enter invalid characters (e.g., numbers).
Expected Results	1. Accepts valid input. 2. Displays an error for empty input. 3. Displays an error for invalid characters.

2.2.2.2 Test Case 2: Last Name Validation

Test Case ID	TC002
Description	Verify that the last name field accepts valid input.
Preconditions	User is on the payment form page.
Test Steps	1. Enter a valid last name. 2. Leave the field empty. 3. Enter invalid characters (e.g., numbers).
Expected Results	1. Accepts valid input. 2. Displays an error for empty input. 3. Displays an error for invalid characters.

2.2.2.3 Test Case 3: Email Validation

Test Case ID	TC003
Description	Verify that the email field accepts valid email formats.
Preconditions	User is on the payment form page.
Test Steps	1. Enter a valid email. 2. Enter an invalid email format (e.g., "test@"). 3. Leave the field empty.
Expected Results	1. Accepts valid email. 2. Displays an error for invalid email format. 3. Displays an error for empty input.

2.2.2.4 Test Case 4: Amount Field Validation

Test Case ID	TC004
Description	Verify that the amount field accepts only positive numbers.
Preconditions	User is on the payment form page.
Test Steps	1. Enter a positive number. 2. Enter zero. 3. Enter a negative number. 4.Enter a number greater than equals to 100,000.
Expected Results	1. Accepts positive number. 2. Displays an error for zero. 3. Displays an error for negative number. 4.Display an error for number out-of upperlimit.

2.2.2.5 Test Case 5: Payment Processing

Test Case ID	TC005
Description	Verify that the payment is processed correctly.
Preconditions	User has filled in all required fields.
Test Steps	1. Fill in valid details. 2. Click the "Pay" button.
Expected Results	1. Payment is processed. 2. Displays success message for successful payment. 3. Displays error message for failed payment.

2.2.2.6 Test Case 6: Session Message Display

Test Case ID	TC006
Description	Verify that success and error messages are displayed correctly.
Preconditions	User is redirected back to the payment form.
Test Steps	1. Trigger a successful payment. 2. Trigger a failed payment.
Expected Results	1. Displays success message. 2. Displays error message.

2.2.2.7 Test Case 7: Usability Testing

Test Case ID	TC007
Description	Assess the user interface and overall user experience.
Preconditions	User is on the payment form page.
Test Steps	1. Navigate through the form. 2. Observe the layout and design. 3. Check for accessibility features.
Expected Results	1. Form is easy to navigate. 2. Design is visually appealing. 3. Accessibility features are present and functional.

2.2.3 Test Data Design

Test Data for TC001: Validate Input Fields

Data Type	Input Data	Description
Valid Input	"John"	Valid first name input.
Invalid Input	""	Empty input for first name.
Invalid Input	"123"	Invalid characters (numbers).
Invalid Input	"@#$%"	Special characters.
Invalid Input	"John123"	Combination of letters and numbers.

Test Data for TC002: Last Name Validation

Data Type	Input Data	Description
Valid Input	"Doe"	Valid last name input.
Invalid Input	""	Empty input for last name.
Invalid Input	"456"	Invalid characters (numbers).
Invalid Input	"@#$%"	Special characters.
Invalid Input	"Doe456"	Combination of letters and numbers.

Test Data for TC003: Email Validation

Data Type	Input Data	Description
Valid Input	"test@example.com"	Valid email.
Valid Input	"user.name@domain.co"	Valid email with subdomain.
Invalid Input	"test@"	Invalid email format (missing domain).
Invalid Input	"user@.com"	Invalid email format (missing username).
Invalid Input	"user@domain"	Invalid email format (missing TLD).
Invalid Input	"user@domain..com"	Invalid email format (consecutive dots).
Invalid Input	""	Empty input for email.

Test Data for TC004: Amount Field Validation

Data Type	Input Data	Description
Valid Input	"10"	Valid positive number.
Valid Input	"100.50"	Valid positive decimal number.
Invalid Input	"0"	Zero input.
Invalid Input	"-10"	Negative number input.
Invalid Input	"-100.50"	Negative decimal number input.
Invalid Input	"100000"	Out-of limit input.
Invalid Input	"abc"	Non-numeric input.

Test Data for TC005: Payment Processing

Data Type	Input Data	Description
Valid Data	"John", "Doe", "test@example.com", "100"	Valid details for payment processing.
Invalid Data	"", "Doe", "test@example.com", "100"	Missing first name.
Invalid Data	"John", "", "test@example.com", "100"	Missing last name.
Invalid Data	"John", "Doe", "test@", "100"	Invalid email format.
Invalid Data	"John", "Doe", "test@example.com", "0"	Zero amount input.

Test Data for TC006: Session Message Display

Data Type	Input Data	Description
Valid Data	"John", "Doe", "test@example.com", "100"	Trigger successful payment.
Invalid Data	"InvalidData"	Trigger failed payment.

Test Data for TC007: Usability Testing

Data Type	Input Data	Description
Usability Test	N/A	User navigates through the form.

3. Test Environment Setup

Hardware Requirements

- **Server**:
  - CPU: Quad-core processor or higher
  - RAM: Minimum 16 GB
  - Storage: SSD with at least 100 GB free space
- **Client Machines**:
  - CPU: Dual-core processor or higher
  - RAM: Minimum 8 GB
  - Storage: Minimum 50 GB free space

Software Requirements

- **Operating System**:
  - Windows 10 or later / macOS
- **Web Browser**:
  - Latest versions of Chrome, Firefox, Safari, and Edge
- **Database**:
  - MySQL >= 5.7
- **Testing Tools**:
  - Test Management Tool (e.g., JIRA, TestRail)
  - Automation Tools (e.g., Laravel Dusk, PHPUnit, Mockery)

Network Requirements

- **Internet Access**:
  - Stable internet connection for accessing online resources and tools
- **Local Network**:
  - Ensure the testing environment is configured similarly to the production setup, including firewall settings and network configurations.

Production Environment Replication

- Ensure the test environment mimics the production environment in terms of:
  - Software versions
  - Configurations
  - Database schema
  - Network settings

4. Test Automation Design

Identify Test Cases Suitable For Automation

- **Test Cases for Automation**:
  - TC001: Validate Input Fields
  - TC002: Last Name Validation
  - TC003: Email Validation
  - TC004: Amount Field Validation
  - TC005: Payment Processing
  - TC006: Session Message Display

Selected Tools

- **Automation Tools**:
  - Laravel Dusk: For web application testing.
  - PHPUnit: For structuring and executing test cases.

Writing TestScripts

- Ensure all test scripts are:
  - Reusable: Common functions and methods should be abstracted.
  - Maintainable: Clear documentation and comments within the code.
  - Version-controlled: Use a Git repository for tracking changes.

5. Traceability Matrix (RTM)

Requirement ID	Test Case ID	Description
REQ001	TC001	Validate first name input.
REQ002	TC002	Validate last name input.
REQ003	TC003	Validate email format.
REQ004	TC004	Validate amount input.
REQ005	TC005	Process payment correctly.
REQ006	TC006	Display session messages correctly.
REQ007	TC007	Assess usability of the user interface.

6. Entry Criteria

- **Test Conditions**: All test conditions must be approved prior to execution.
- **Test Design Templates**: Ensure all test design templates and standards are finalized and ready for use.
- **Environment Setup**: The test environment must be fully configured and verified to be operational.
- **Test Data**: All necessary test data must be prepared and accessible for each test case.