

Mini Project: Functional Testing

Application: Healthcare Member Portal – Claim Submission (Web/Mobile)

Objective:

Test the functionality of a member portal where users can log in, submit claims, and track claim status.

Activities:

1. Requirement Analysis

Modules: Member Login, Claim Form Submission, Upload Documents, View Claim Status.

2. Define Test Scenarios & Test Cases and Conduct a Peer Review of test cases.

Examples:

Login with valid/invalid credentials.

Submit claim with required fields.

Upload supporting documents (PDF, JPG).

View claim status (Pending, Approved, Rejected).

3. Test Execution & Result Capture

Execute test cases manually.

Record results in Excel or a test management tool.

4. Defect Identification & Documentation

Log defects with screenshots, severity, and reproduction steps.

5. Additional Activities

Prepare a Test Summary Report.

Create a Traceability Matrix.

Document, peer review outcomes.

Mini Project: Automation Testing

Application: Healthcare Member Portal – Claim Submission (Web/Mobile)

Objective:

Automate key flows of the claim submission process in the member portal.

Tools: Selenium / Appium / AccelQ

Activities:

1. Requirement Analysis

Focus on modules: Login, Submit Claim, Upload Documents, View Status.

2. Define Test Scenarios & Test Cases

Prioritize P1 flows:

Login

Submit claim with valid data

Upload document

View claim status

3. Automation of P1 Scenarios

Use best practices:

Page Object Model

Data-driven testing (e.g., multiple claim types)

Reusable functions

4. Execution & Result Capture

Run scripts and generate comprehensive report.

Make sure report Captures logs and screenshots.

5. Defect Identification & Documentation

Document failed test cases with evidence.

6. Additional Activities

Generate HTML/Excel reports.

Integrate with Git (if applicable).

Present a demo of the automation suite.

Mini Project: Database Testing

Application: Healthcare Member Portal – Claim Submission

Database: Oracle SQL

Objective:

Validate backend data operations for claim submission and tracking.

Activities:

1. Create Sample Table for Claims and Member. (Example)

```
CREATE TABLE Members (MemberID INT PRIMARY KEY, Name VARCHAR(100), DOB DATE, Email VARCHAR(100));
```

```
CREATE TABLE Claims (ClaimID INT PRIMARY KEY, MemberID INT, ClaimType VARCHAR(50), Amount DECIMAL(10,2), Status VARCHAR(20), SubmissionDate DATE, FOREIGN KEY (MemberID) REFERENCES Members(MemberID));
```

2. CRUD Operations

Create:

Insert new records into Members and Claims tables.

Ensure foreign key relationships are maintained (i.e., each claim is linked to a valid member).

Read:

Fetch claims by MemberID, Status, or SubmissionDate.

Retrieve member details along with their claims using JOIN queries.

Update:

Modify claim status (e.g., from "Pending" to "Approved").

Update member contact details if needed.

Delete:

Remove test claims and/or test members.

Ensure cascading deletes or referential integrity is handled correctly.

3. Test Scenarios

Validate insertion of member and claim records.

Ensure claim status updates reflect correctly in the database.

Check for duplicate ClaimID or MemberID entries.

Verify deletion of claims and ensure no orphaned records.

Confirm foreign key constraints between Claims and Members.

Validate JOIN queries return accurate combined data.

4. Query-Based Validation

Total claims submitted by each member

Claims with status "Pending" and member details

Claims submitted in the last 30 days

Members with no claims (if applicable)

5. Additional Activities

Document test data and queries.

Log defects for data mismatches.

Prepare a Database Test Report.