# **Test Plan (Benefit Store Application)**

[**Test Plan (Benefit Store Application)** 1](#_Toc193720554)

[**1. Introduction** 2](#_Toc193720556)

[**2. Objectives** 2](#_Toc193720557)

[**3. Scope** 2](#_Toc193720558)

[**Inclusions** 2](#_Toc193720559)

[**Exclusions** 3](#_Toc193720560)

[**Test Environments** 3](#_Toc193720561)

[**Defect Reporting Procedure** 3](#_Toc193720562)

[**Test Strategy** 4](#_Toc193720563)

[**Test Schedule** 5](#_Toc193720564)

[**Test Deliverables** 6](#_Toc193720565)

[**Entry & Exit Criteria** 6](#_Toc193720566)

[**Requirement Analysis** 6](#_Toc193720567)

[**Test Execution** 7](#_Toc193720568)

[**Test Closure** 7](#_Toc193720569)

[**4. Tools** 7](#_Toc193720570)

[**5. Risks and Mitigations** 7](#_Toc193720571)

[**6. Approvals** 8](#_Toc193720572)

# 

# **1. Introduction**

The Test Plan for **Benefit Store application** outlines the testing strategy, approach, execution plan for the application. This document provides an overview of the testing process to be followed during the development, ensuring that the application meets all functional, security, usability, and performance requirements, providing a seamless and reliable experience for users.

# **2. Objectives**

The objective of this test plan is to ensure that the application meets all the functional requirements, provides a user-friendly experience, is secure, and performs well under various load conditions. To achieve this, the testing process will focus on the following key objectives:

* Ensure data accuracy and validation of user inputs.
* Validate Plan selection and Cart functionality.
* Ensure the correct calculations of costs and fees.
* Ensure the validation of personal and billing information.
* Identify potential defects before production development.
* Ensure the system meets security and performance benchmarks.

# **3. Scope**

## **Inclusions**

* Landing Page
* Plan Listing Page
* Plan Details Page
* Cart Page
* Checkout Process:
  + Personal Information
  + Contact Information
  + Billing Information
  + Agreement and Signature
  + Review and Confirmation

## **Exclusions**

* User login/ registration.
* Integration with third-party payment gateways.
* Backend database operations.

From my understanding I believe these features represent the core of the application and should be tested.

## **Test Environments**

**Operating Systems:** Windows, Android.

**Browsers:** Chrome, Edge, Firefox.

**Devices:** Desktop, Mobile Tablet.

**Network Connectivity:** Wi-Fi, Cellular, and Wired connections.

## **Defect Reporting Procedure**

During the test execution –

* Any deviation from the expected behavior by the application will be noted. If it can’t be noted as a defect, it’d be reported as an observation/issue or posed as a question.
* Any usability issues will also be reported.
* After discovery of a defect, it will be retested to verify reproducibility of the defect.  
  Screenshots with steps to reproduce are documented.
* Every day, at the end of the test execution, defects encountered will be sent along with the observations. (No real client but this is what I’d do if I had a real client)
* Bug Report Title will adhere to the following syntax:
  + Bug ID
  + Module Name
  + Priority/Severity
  + Steps to produce
  + Expected vs Actual Result
  + Attachments

**Note:**

* Defects will be documented in JIRA or ZOHO Sprints.
* Test Scenarios and Test Cases will be documented in an excel document.

## **Test Strategy**

As part of functional testing, I am going to follow the below approach for this project:

**Step 1 – Creation of Test Scenarios and Test Cases for all features in scope.**

* We will apply several Test Design techniques while creating the test case
* Equivalence Class Partition
* Boundary Value Analysis
* Decision Table Testing
* State Transition Testing
* Use Case Testing
* Additional Approaches:
* Exploratory Testing
* Error Guessing
* Test cases will be prioritized based on their importance and impact.

**Step 2 –Test Execution Process**

Firstly, Perform Smoke Testing to check the different and important functionalities of the application working at a high level.

* We will reject the build, if the Smoke Testing fails and will wait for the stable build before performing in depth testing of the application functionalities.
* Once We receive a stable build, which passes Smoke Testing, we will perform in-depth testing using the Test Cases created.
* We will then report the bugs in the “Trello/JIRA/ZOHO” bug tracking tool and send you the defect found on that day in a status-update end of the day email (I have no client in this project but that’s what I would do).
* If applicable, we will repeat the Test Cycles until we get a quality product.

**Step 3 – Follow the below test practices:**

* **Context Driven Testing** – We will perform the testing based on the context of the Project.
* **Shift Left Testing** – We will perform testing from the development stages of the Project.
* **Exploratory Testing** – Apart from executing planned test cases, we will conduct unscripted testing to find unexpected issues.
* **End to End Flow Testing** – We will test full user journeys across multiple features to simulate real-world scenarios.

## **Test Schedule**

Following is the test schedule planned for the project –

|  |  |
| --- | --- |
| **Task** | **Dates** |
| Requirement Analysis | **2025/03/21** |
| Creating Test Plan | **2025/03/24** |
| Test Case Creation | **2025/03/25** |
| Test Case Execution | **2025/03/26** |
| Summary Report Submission Date | **2025/03/27** |

## **Test Deliverables**

|  |  |  |
| --- | --- | --- |
| **Deliverable** | **Description** | **Target Completion Date** |
| Test Plan | Details on the scope of the Project, test schedule, resource requirements, test deliverables and schedule | March 24 |
| Test Scenarios | Functionalities for which we can write test cases. | March 25 |
| Test Cases | Test Cases created for the scope defined | March 25 – March 26 |
| Bug Reports | Detailed bug reports of the bugs identified along with screenshots and steps to reproduce | March 26 |

The following are to be delivered to the client:

## **Entry & Exit Criteria**

The below are the entry and exit criteria for every phase of Software Testing Life Cycle:

## **Requirement Analysis**

**Entry Criteria:**

* Requirement documents are finalized.
* Test environments are set up.
* Test cases are prepared and reviewed.

**Exit Criteria:**

* List of requirements are explored and understood by the Testing team
* Doubts are cleared

## **Test Execution**

**Entry Criteria:**

* Test Scenarios and Test Cases documents are signed off by the client
* Application is ready for testing

**Exit Criteria:**

* Test Case Reports and Bug Reports are ready

## **Test Closure**

**Entry Criteria:**

* Test Case Reports and Bug Reports are ready

**Exit Criteria:**

* Test Summary Report

# **4. Tools**

The following are the list of Tools we will be using in this Project:

* JIRA / Zoho Sprints
* Mind map Tool
* Snipping Screenshot Tool
* Word and Excel documents

# **5. Risks and Mitigations**

The following are the list of risks possible and the ways to mitigate them:

|  |  |
| --- | --- |
| Risk | Mitigation |
| Non – Availability of resources | Backup resource planning. |
| Build URL is not working | Resources will work on other tasks |
| Limited testing time | Prioritize testing based on risk. |

# **6. Approvals**

Team will send different types of documents for Approval like below:

|  |  |  |
| --- | --- | --- |
| **Document** | **Approval Required From** | **Signature** |
| Test Plan | QA Lead, Project Manager |  |
| Test Scenarios | QA Lead, Developers |  |
| Test Cases | QA Lead, Project Manager |  |
| Reports | QA Lead, Business Stakeholders |  |

Testing will only continue to the next steps once these approvals are done.

### 