

**HealingLand - DaNang Homestay booking platform**



List all of project members

* *Nguyễn Viết Thành Trung - DE170677*
* *Nguyễn Bình Tú - DE170690*
* *Nguyễn Lê Quang Huy - DE170716*
* *Trần Đồng Ánh Hồng - DE170285*
* *Hoàng Văn Hậu - DE170362*

[11,June,2024]

SWT301\_SP24

**Version:** 1.0

**Created:** 11/06/2024

**Last Updated:** //2024

**Status:** Draf

**Revision and Sign off Sheet**

**Document History** - To maintain a list of changes being made

| **Version** | **Date** | **Author** | **Description of Change** |
| --- | --- | --- | --- |
| 1 | 1/6/2024 | Trần Đồng Ánh Hồng | Draft |
| 2 | 11/6/2024 | Nguyễn Lê Quang Huy | Draft - Reviewed |
|  |  |  |  |

**Approvers List** - To track who has reviewed and signoff on the Test plan

| Name | Role | Approver / Reviewer | Approval / Review Date |
| --- | --- | --- | --- |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

**Reference Documents** - Clearly mark the document used as an input to create the test plan

| **Version** | **Date** | **Document Name** |
| --- | --- | --- |
| 1.0 |  | SRS of HealingLand - DaNang Homestay booking platform |
|  |  |  |

Table of Contents

[1](#_heading=h.gjdgxs) Introduction 4

[1.1](#_heading=h.30j0zll) Purpose 4

[1.2](#_heading=h.1fob9te) Project Overview 4

[2](#_heading=h.3znysh7) Scope 4

[2.1](#_heading=h.2et92p0) Feature to be tested 4

[2.2](#_heading=h.tyjcwt) Feature not to be tested 4

[3](#_heading=h.3dy6vkm) Testing Strategy 4

[3.1](#_heading=h.1t3h5sf) Test Objectives 4

[3.2](#_heading=h.4d34og8) Test Assumptions 4

[3.3](#_heading=h.2s8eyo1) Data Approach 5

[3.4](#_heading=h.17dp8vu) Test type 5

[3.5](#_heading=h.3rdcrjn) Unit Testing 5

[3.6](#_heading=h.26in1rg) Functional Testing 5

[3.7](#_heading=h.lnxbz9) User Acceptance Testing 6

[3.8](#_heading=h.35nkun2) Regression Testing 6

[4](#_heading=h.1ksv4uv) Execution Strategy 6

[4.1](#_heading=h.44sinio) Entry Criteria 6

[4.2](#_heading=h.2jxsxqh) Exit criteria 7

[4.3](#_heading=h.z337ya) Validation and Defect Management 7

[5](#_heading=h.3j2qqm3) Environment Requirements 8

[6](#_heading=h.1y810tw) Schedule & Estimation 8

[6.1](#_heading=h.4i7ojhp) All project task and estimation 8

[6.2](#_heading=h.2xcytpi) Schedule to complete these tasks 9

[7](#_heading=h.1ci93xb) TEST DELIVERABLES test deliverables are provided as below 9

[7.1](#_heading=h.3whwml4) Before testing phase - Test plans document. - Test cases documents - Test Design specifications. 9

[7.2](#_heading=h.2bn6wsx) During the testing - Test Tool - Simulators. - Test Data - Test Trace-ability Matrix - Error logs and execution logs. 9

[7.3](#_heading=h.qsh70q) After the testing cycles is over - Test Results/reports - Defect Report - Installation/ Test procedures guidelines - Release notes 9

# Introduction

## Purpose

The purpose of this document is to provide a comprehensive overview of the test strategy, test approach, execution strategy, and test management for the Online Homestay Booking System project. This test plan outlines the necessary steps and procedures to ensure that all components of the system are thoroughly tested, verified, and validated. It aims to ensure that the final product meets all specified requirements and functions correctly in the intended environment.

**Test Strategy:** Defines the types of testing: functional, integration, system, and user acceptance.

**Test Approach:** Details methods and techniques, including automated and manual testing, test case design, and test environments.

**Execution Strategy:** Outlines test scheduling, task assignments, and defect reporting and tracking.

**Test Management:** Covers planning, monitoring, control, and use of test management tools to track progress and manage test cases and defects.

## Project Overview

In today's rapidly developing context, the need for people's spiritual life is emphasized and finding diverse tourist destinations and places to stay has become an essential need for each of us. ta. Da Nang is no exception, this place is quite an attractive and potential tourist destination for tourism in our country. With a long coastline, famous landscapes, diverse cuisine and culture, this place attracts millions of domestic and foreign tourists every year. Besides traditional hotels and resorts, the homestay model is also a place that people are interested in thanks to its friendliness and closeness to real-life local experiences.

However, finding a suitable place to stay and experience is not easy due to lack of connection and systematization of information. Therefore, our Da Nang homestay booking platform was born, to solve the necessary needs and difficulties of tourism participants. This Homestay platform, in addition to providing detailed information about homestays, also provides users with functions such as searching, viewing and reviewing, and booking rooms online conveniently and quickly. The goal of our project is to provide everyone with a specialized online information technology platform for booking rooms in the Da Nang area. The platform we developed is to create a bridge between homestay owners and tourists, helping to optimize the search and booking process, enhancing the user experience. At the same time, this platform also helps homestay owners manage and promote their services in a clear, specific, professional and highly effective way.

Through research and development of this system, our project hopes to contribute to promoting the tourism industry in Da Nang. Improve booking efficiency, quality of accommodation services and meet the needs of tourists.

# Scope

## Feature to be tested

*Describes what is being tested, such as all the functions/features of a specific project/product/solution.*

*Specify interfaces, process, workflow, vendor integration.*

* ***1. User Authentication and Authorization***

***Features:***

* + *User registration (sign up)*
  + *User login (sign in)*
  + *Password recovery (forgot password, change password)*
  + *User profile(update profile, create profile)*
  + *User roles and permissions (admin, owner,customer, guest)*

***Testing Scope:***

* + ***Functional Testing:*** *Verify that each authentication function works correctly (users can sign up, sign in, recover passwords, ,update profile).*
  + ***Security Testing:*** *Ensure secure password storage, protection against common attacks (encode password , sql rejection).*
* ***2. Booking Management***

***Features:***

* + *Search availability of homestays*
  + *Make a booking (date, time, rooms , payment)*
  + *Cancel or modify bookings*
  + *View booking history*

***Testing Scope:***

* + ***Functional Testing:*** *Confirm that users can search, book, cancel, and modify services.*
  + ***Usability Testing:*** *Ensure the booking process is intuitive and user-friendly.*
  + ***Performance Testing:*** *Assess the platform's performance under high traffic conditions.*

***3. Payment Processing***

***Features:***

* + *Integration with payment gateways (MOMO)*
  + *Secure payment transactions*
  + *Payment confirmation and receipts*

***Testing Scope:***

* + ***Functional Testing:*** *Verify that payment processing works for various payment methods.*
  + ***Security Testing:*** *Ensure payment data is handled securely .*
  + ***Performance Testing:*** *Test the payment system's reliability and speed.*

***4. User Profile Management***

***Features:***

* + *View and edit personal information*
  + *Manage preferences ( notification settings)*

***Testing Scope:***

* + ***Functional Testing:*** *Ensure users can view and update their profile information.*
  + ***Usability Testing:*** *Confirm that profile management features are easy to use and accessible.*

***5. Homestays Listings and Descriptions***

***Features:***

* + *Display detailed descriptions of available homestays*
  + *Show images of homestays*
  + *Include pricing and duration information*

***Testing Scope:***

* + ***Functional Testing:*** *Check that homestays listings are accurate and complete.*
  + ***Usability Testing:*** *Ensure homestays descriptions are clear and engaging.*
  + ***SEO Testing:*** *Optimize homestays listings for search engines.*

***6. Customer Support***

***Features:***

* + *Feedback Homestays*
  + *Contact forms*

***Testing Scope:***

* + ***Functional Testing:*** *Verify that customers can access support through various channels.*
  + ***Usability Testing:*** *Ensure support features are easily accessible and helpful.*
  + ***Performance Testing:*** *Test the responsiveness and reliability of customer support features.*

***7. Notifications and Alerts***

***Features:***

* + *Email and SMS notifications for bookings, cancellations, and reminders*

***Testing Scope:***

* + ***Functional Testing:*** *Confirm that notifications are sent and received correctly.*
  + ***Usability Testing:*** *Ensure notifications are clear and informative.*
  + ***Performance Testing:*** *Check the timeliness and reliability of notifications.*

***8. Admin Panel***

***Features:***

* *Manage bookings and user accounts*
* *Generate reports and analytics*
* *Configure services and pricing*
* *Accept or reject homestays*

***Testing Scope:***

* ***Functional Testing:*** *Verify admin functionalities work as intended.*
* ***Security Testing:*** *Ensure only authorized personnel can access admin features.*
* ***Usability Testing:*** *Confirm that the admin panel is user-friendly for staff.*

## Feature not to be tested

*Identify all features and combinations of features which will not be tested and the reasons.*

### *Features Not to Be Tested*

***Features:*** *Detailed functionalities of third-party services ( payment gateways beyond basic transaction verification, email verification of google ).*

***Reasons:***

***Assumption of Reliability:*** *These services are assumed to be reliable as they are maintained by external, specialized providers.*

***Limited Control:*** *The development team has limited control over these third-party services, making in-depth testing less feasible.*

1. ***Infrequently Used Admin Panel Features***

***Features:*** *Rarely used administrative functions such as exporting data to obscure formats.*

***Reasons:***

***Low Usage Frequency:*** *These features are used infrequently and may not impact the overall user experience significantly.*

***Resource Allocation:*** *Prioritizing more critical admin features that are essential for day-to-day operations.*

1. ***Non-Critical Edge Cases***

***Features:*** *Extremely rare edge cases in booking and payment processes.*

***Reasons:***

***Low Impact:*** *These cases are unlikely to affect a significant number of users.*

***Resource Optimization:*** *Focus resources on more common scenarios that impact a larger user base.*

1. ***Cosmetic and Minor UI Issues***

***Features:*** *Minor visual inconsistencies that do not affect functionality (e.g., pixel-level alignment issues).*

***Reasons:***

***User Tolerance:*** *Users are generally tolerant of minor cosmetic issues.*

***Prioritization:*** *Focus on more impactful usability and functional issues.*

1. ***Complex Multi-Step Booking Modifications in Rare Scenarios***

* ***Combination:*** *Modifying a booking after multiple changes (e.g., changing service, date, and then modifying payment method).*
* ***Reasons:***
  + ***Complexity and Rarity:*** *Such complex modifications are rare and add significant testing complexity.*
  + ***Focus on Core Flows:*** *Concentrate on ensuring the basic and most common booking modification flows work seamlessly.*

# Testing Strategy

## Test Objectives

* Ensure all functionalities of the Online Homestay Booking System work as expected.
* Verify the system's integration with payment gateways and third-party services.
* Validate user interface and user experience meet design specifications.
* Confirm data security and privacy measures are implemented correctly.
* Identify and resolve any defects or issues before the system goes live.

#### 

#### 

#### 

#### ***Tasks and Responsibilities***

* ***QA Team****: Design and execute test cases, report defects, and verify fixes.*
* ***Developers****: Fix reported defects and support the QA team during testing.*
* ***Project Manager****: Oversee the testing process, ensure timelines are met, and coordinate between teams.*
* ***Business Analyst****: Validate test cases against business requirements and provide clarifications.*
* ***Operations Team****: Test system performance, scalability, and reliability.*

## Test Assumptions

* The development team will provide a stable build for testing.
* All necessary testing environments and tools will be available and configured correctly.
* Test data will be prepared and provided before the testing begins.
* Stakeholders will be available for feedback and review throughout the testing process.
* Any changes in requirements will be communicated promptly to the QA team.

## Data Approach

**Functional Testing**

* **Test Data Preparation**: Create test data that covers all functional scenarios, including valid, invalid, boundary, and edge cases.
* **Data Sources**: Use a combination of synthetic data generated specifically for tests and anonymized production data.
* **Data Maintenance**: Ensure data consistency across different test environments. Regularly refresh test data to reflect the latest changes in the application.

#### **User Acceptance Testing (UAT)**

* **Realistic Data**: Use data that closely mimics real-world scenarios to provide a realistic testing environment.
* **User Roles**: Include data for various user roles (e.g., guests, hosts, administrators) to test different user journeys.
* **Data Security**: Maintain data privacy and security by masking sensitive information. Ensure compliance with data protection regulations.
* **Environment Setup**: Ensure the UAT environment mirrors the production environment to identify potential issues early.

## Test type

| **Test Type** | **Description** | **Responsible Parties** |
| --- | --- | --- |
| Functional Testing | Test individual functions of the system to ensure they work as expected. | QA Team |
| Integration Testing | Test interactions between different modules and third-party services to ensure seamless operation. | QA Team, Developers |
| System Testing | Conduct end-to-end testing of the complete system in a controlled environment. | QA Team, Project Manager |
| User Acceptance Testing (UAT) | Validate the system with real-world scenarios to ensure it meets user requirements. | Business Analyst, End Users |
| Performance Testing | Assess the system's performance, including load, stress, and scalability testing. | Operations Team |
| Security Testing | Ensure the system's data protection and privacy measures are effective and secure. | Security Team |

## Unit Testing

Participants:

| **Tester’s Name** | **Department/ Area** | **Role** |
| --- | --- | --- |
| Nguyễn Viết Thành Trung | QA Department | Test Manager, Test Analyst |
| Nguyễn Bình Tú | Software Development | Test Lead, Functional Tester |
| Nguyễn Lê Quang Huy | QA Department | Test Analyst |
| Trần Đồng Ánh Hồng | User Experience (UX) | UX Tester |
| Hoàng Văn Hậu | Database Administration | Database Tester, Test Analyst |

## 

## 

## 

## 

## 

## 

## 

## 

## Functional Testing

Participants:

| **Tester’s Name** | **Department/ Area** | **Role** |
| --- | --- | --- |
| Nguyễn Viết Thành Trung | QA Department | Test Manager, Test Analyst |
| Nguyễn Bình Tú | Software Development | Test Lead, Functional Tester |
| Nguyễn Lê Quang Huy | QA Department | Test Analyst |
| Trần Đồng Ánh Hồng | User Experience (UX) | UX Tester |
| Hoàng Văn Hậu | Database Administration | Database Tester, Test Analyst |

## User Acceptance Testing

Participants:

| **Tester’s Name** | **Department/ Area** | **Role** |
| --- | --- | --- |
| Nguyễn Viết Thành Trung | QA Department | Test Manager, Test Analyst |
| Nguyễn Bình Tú | Software Development | Test Lead, Functional Tester |
| Nguyễn Lê Quang Huy | QA Department | Test Analyst |
| Trần Đồng Ánh Hồng | User Experience (UX) | UX Tester |
| Hoàng Văn Hậu | Database Administration | Database Tester, Test Analyst |

## 

## 

## 

## 

## Regression Testing

Participants:

| **Tester’s Name** | **Department/ Area** | **Role** |
| --- | --- | --- |
| Nguyễn Viết Thành Trung | QA Department | Test Manager, Test Analyst |
| Nguyễn Bình Tú | Software Development | Test Lead, Functional Tester |
| Nguyễn Lê Quang Huy | QA Department | Test Analyst |
| Trần Đồng Ánh Hồng | User Experience (UX) | UX Tester |
| Hoàng Văn Hậu | Database Administration | Database Tester, Test Analyst |

# Execution Strategy

## 4.1 Entry Criteria

* The entry criteria refer to the desirable conditions in order to start test execution; only the migration of the code and fixes need to be assessed at the end of each cycle.
* Entry criteria to start the execution phase of the test: the activities listed in the Test Planning section of the schedule are 100% completed.
* Entry criteria to start each cycle: the activities listed in the Test Execution section of the schedule are 100% completed at each cycle.

| **Entry Criteria** | **Test Team** | **Technical Team** | **Notes** |
| --- | --- | --- | --- |
| *Test environment(s) is available* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *Test data is available* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |
| *Code has been merged successfully* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *Development has completed unit testing* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *Test scripts are completed, reviewed and approved by the Project Team* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |

## 4.2 Exit criteria

* The exit criteria are the desirable conditions that need to be met in order proceed with the implementation.
* Entry and exit criteria are flexible benchmarks. If they are not met, the test team will assess the risk, identify mitigation actions and provide a recommendation. All this is input to the project manager for a final “go-no go” decision.

| **Exit Criteria** | **Test Team** | **Technical Team** | **Notes** |
| --- | --- | --- | --- |
| *100% Test Scripts executed* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *90% pass rate of Test Scripts* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *No open Critical and High severity defects* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *All remaining defects are either canceled or documented as Change Requests for a future release* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *All expected and actual results are captured and documented with the test script* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *All test metrics collected based on reports from daily and Weekly Status reports* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *All defects logged in Defect Tracker/Spreadsheet* |  | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |
| *Test environment cleanup completed and a new back up of the environment* | C:\Users\arxp\AppData\Local\Microsoft\Windows\Temporary Internet Files\Content.IE5\7F9Z3IW4\MC900441310[1].png |  |  |

## 4.3 Validation and Defect Management

#### **Validation of Test Cases/Test Scenarios**

1. **Test Case Validation Process**

* **Execution**: Testers will execute all test scripts according to the planned test cycles. Each test script will be run against the latest stable build in the designated test environment.
* **Result Documentation**: Testers will document the results of each test case, indicating whether the test case passed or failed. Detailed notes should be included for any deviations or unexpected behavior.
* **Peer Review**: Test results and the corresponding documentation will be peer-reviewed by another team member to ensure accuracy and completeness.
* **Approval**: Once reviewed, test results will be submitted to the test lead or project manager for approval.

1. **Validation Criteria**

* **Completeness**: All steps in the test case have been executed and results are documented.
* **Accuracy**: The actual results are accurately recorded and compared against expected results.
* **Traceability**: Each test case should be traceable to the corresponding requirement or user story.
* **Consistency**: Test cases should be executed consistently across different cycles, and any discrepancies should be analyzed and documented.

#### **Defect Management**

1. **Defect Tracking Tools**

* **Defect Tracker/Spreadsheet**: All defects will be tracked using a centralized defect tracker or a spreadsheet, ensuring visibility and traceability of issues.

1. **Defect Management Process**

* **Defect Logging**:
  + - **Responsibility**: It is the tester’s responsibility to log any defects discovered during test execution.
    - **Details Required**: Each defect should include a clear title, description, steps to reproduce, expected results, actual results, severity, priority, screenshots or logs (if applicable), and the environment in which the defect was found.
* **Defect Triage**:
  + - **Responsibility**: The test lead or defect manager will review logged defects, assign severity and priority, and determine the responsible developer.
    - **Meeting**: Regular defect triage meetings will be held to discuss and prioritize defects.
* **Defect Resolution**:
  + - **Responsibility**: The assigned developer will investigate, fix the defect, and update the status in the defect tracker.
* **Defect Verification**:
  + - **Responsibility**: Once a defect is marked as resolved, the original tester will retest the issue to verify the fix.
    - **Closure**: If the fix is successful and the defect no longer occurs, the tester will close the defect. If the defect persists, it will be reopened and re-assigned for further investigation.
* **Defect Reporting**:
  + - **Responsibility**: Test leads or project managers will generate defect reports to monitor the overall defect status and trends throughout the testing cycles.

1. **Defect Life Cycle Stages**

* **New**: A defect is newly logged and yet to be reviewed.
* **Assigned**: A defect is assigned to a developer for resolution.
* **In Progress**: The developer is currently working on the defect.
* **Fixed**: The developer has fixed the defect and it is ready for retesting.
* **Retesting**: The defect is being retested by the tester.
* **Reopened**: The defect is reopened if the issue persists after the fix.
* **Closed**: The defect is verified as fixed and closed.
* **Deferred**: The defect is deferred to be fixed in future releases due to low priority or other reasons.

Defects found during the Testing should be categorized as below:

| **Severity** | **Impact** |
| --- | --- |
| *1 (Critical)* | * This bug is critical enough to crash the system, cause file corruption, or cause potential data loss * It causes an abnormal return to the operating system (crash or a system failure message appears). * It causes the application to hang and requires rebooting the system. |
| *2 (High)* | * It causes a lack of vital program functionality with workaround. |
| *3 (Medium)* | * This Bug will degrade the quality of the System. However there is an intelligent workaround for achieving the desired functionality - for example through another screen. * This bug prevents other areas of the product from being tested. However other areas can be independently tested. |
| *4 (Low)* | * There is an insufficient or unclear error message, which has minimum impact on product use. |

# Environment Requirements

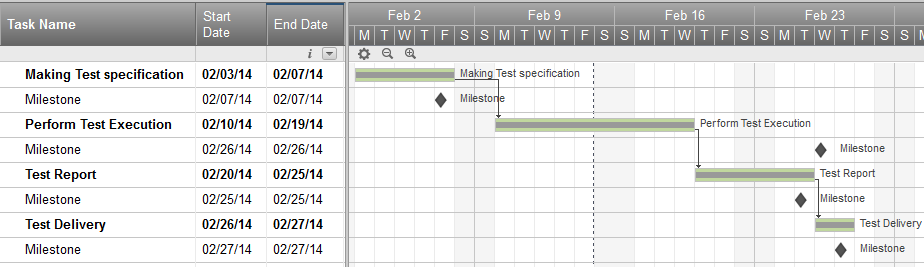
* **Development Environment**: For initial testing by developers.
* **Staging Environment**: Mirrors the production environment for pre-production testing.
* **Testing Environment**: Main environment for functional, integration, and performance testing.
* **User Acceptance Testing (UAT) Environment**: For final validation by end-users.
* **Performance Testing Environment**: Configured for load testing to evaluate performance.

# Schedule & Estimation

## All project task and estimation

| **Task** | **Members** | **Estimate effort** |
| --- | --- | --- |
| **Create the test specification** | Test Designer | 170 man-hour |
| **Perform Test Execution** | Tester, Test Administrator | 80 man-hour |
| **Test Report** | Tester | 10 man-hour |
| **Test Delivery** | 20 man-hour |  |
| **Total** | **280 man-hour** |  |

## Schedule to complete these tasks



# TEST DELIVERABLES test deliverables are provided as below

#### **Before Testing Phase**

## Test Plan Document: Outlines the scope, approach, resources, and schedule of the testing activities.

## Test Cases Documents: Detailed test cases that describe the inputs, execution conditions, and expected results.

## Test Design Specifications: Descriptions of the test design and the requirements it covers.

#### **During the Testing**

## Test Tool: Software or tools used for automated a’nd manual testing.

## Simulators: Tools that mimic the operation of the homestay booking platform to test different scenarios.

## Test Data: Data used for executing the test cases.

## Test Traceability Matrix: A document that maps and traces user requirements with the test cases.

## Error Logs and Execution Logs: Records of errors and logs of test executions for analysis and debugging.

#### **After the Testing Cycles is Over**

## Test Results/Reports: Summarize the outcomes of the testing activities.

## Defect Report: Details the defects found during testing, their status, and their resolution.

## Installation/Test Procedures Guidelines: Instructions for installing the system and performing the tests.

## Release Notes: Document the new features, improvements, and known issues in the release.

## 