
Group 13

**Share Bug
Master Test Plan**

Version 1.3

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

Revision History

| Date | Version | Description | Author |
|------------|---------|--|-----------------------|
| 12/06/2024 | 1.0 | <ul style="list-style-type: none"> Evaluation Mission and Test Motivation Initial Test-Idea Catalogs and Other Reference Sources Entry and Exit Criteria Responsibilities, Staffing, and Training Needs Initial Test-Idea Catalogs and Other Reference Sources User Interface Profiling, Stress Testing, Failover and Recovery Testing | Nguyễn Lê Anh Chi |
| 13/06/2024 | 1.1 | <ul style="list-style-type: none"> Target Test Items Outline of Planned Tests Management Process and Procedures Function Testing, Volume Testing, Security and Access Control Testing | Nguyễn Xuân Quỳnh Chi |
| 14/06/2024 | 1.2 | <ul style="list-style-type: none"> Introduction Deliverables Iteration milestones Load Testing, Configuration Testing, Deployment Testing | Bùi Ngọc Kiều Nhi |
| 15/06/2024 | 1.3 | <ul style="list-style-type: none"> Testing Workflow Environmental Needs Risks, Dependencies, Assumptions, and Constraints Data and Database Integrity Testing, Business Cycle Testing, Performance Profiling | Lê Ngô Song Cát |
| 20/08/2024 | 2.0 | <ul style="list-style-type: none"> Adjustments to meet teacher's feedback | Nguyễn Lê Anh Chi |

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

Table of Contents

| | | |
|-------|-------------------------------|---|
| 1. | Overview | 4 |
| 1.1 | Purpose | 4 |
| 1.2 | Test Objectives | 4 |
| 1.3 | Roles and Responsibility | 4 |
| 1.4 | Process and Workflows | 5 |
| 2. | Testing Levels and Techniques | 6 |
| 3. | Test Results | 7 |
| 3.1 | Release 1 | 7 |
| 3.2 | Release 2 | 8 |
| 3.3 | Final | 8 |
| 4. | Outline of Planned Tests | 8 |
| 4.1 | Outline of Test Inclusions | 8 |
| 4.1.1 | Functional Testing | 8 |
| 4.1.2 | Performance Testing | 8 |
| 4.1.3 | Security Testing | 8 |
| 4.1.4 | Usability Testing | 9 |
| 4.1.5 | Compatibility Testing | 9 |
| 4.1.6 | Regression Testing | 9 |
| 5. | Resources and Schedules | 9 |
| 5.1 | Resources | 9 |
| 5.2 | Schedules | 9 |

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

Test Plan

1. Overview

1.1 Purpose

The purpose of the Iteration Test Plan is to gather all of the information necessary to plan and control the test effort for a given iteration. It describes the approach to testing the software, and is the top-level plan generated and used by managers to direct the test effort.

This *Test Plan* for the Share Bug supports the following objectives:

- Identifies the items that should be targeted by the tests.
- Identifies the motivation for and ideas behind the test areas to be covered.
- Outlines the testing approach that will be used.
- Identifies the required resources and provides an estimate of the test efforts.
- Lists the deliverable elements of the test project.

1.2 Test Objectives

The purpose of the Iteration Test Plan is to gather all of the information necessary to plan and control the test effort for a given iteration. It describes the approach to testing the software, and is the top-level plan generated and used by managers to direct the test effort.

1.3 Roles and Responsibility

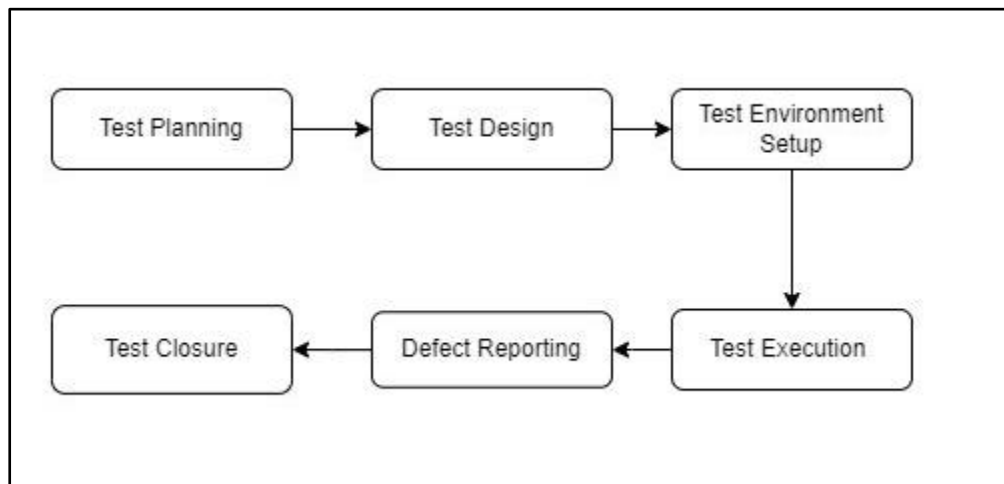
The testing sessions will be conducted 3 times, for Release 1, Release 2 and Final.

| Human Resources | | |
|-----------------|--|--|
| Role | Minimum Resources Recommended (number of full-time roles allocated) | Specific Responsibilities or Comments |
| Test Manager | 1 (Anh Chi) | Provides management oversight. Responsibilities include: <ul style="list-style-type: none"> • planning and logistics • agree mission • identify motivators • acquire appropriate resources • present management reporting • advocate the interests of test • evaluate effectiveness of test effort |
| Test Analyst | 1 (Quynh Chi) | Identifies and defines the specific tests to be conducted. Responsibilities include: <ul style="list-style-type: none"> • identify test ideas |

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

| | | |
|---------------|---------------------------|--|
| | | <ul style="list-style-type: none"> • define test details • determine test results • document change requests • evaluate product quality |
| Test Designer | 2 (Song Cat, Kieu Nhi) | <p>Defines the technical approach to the implementation of the test effort.</p> <p>Responsibilities include:</p> <ul style="list-style-type: none"> • define test approach • define test automation architecture • verify test techniques • define testability elements • structure test implementation |
| Tester | 4 (All) | <p>Implements and executes the tests.</p> <p>Responsibilities include:</p> <ul style="list-style-type: none"> • implement tests and test suites • execute test suites • log results • analyze and recover from test failures <p>document incidents</p> |

1.4 Process and Workflows



1. **Test Planning:** Define objectives, targets, requirements, and necessary tools/resources.
2. **Test Design:** Design test cases, test scenarios based on test conditions and generate the data needed to execute the test cases.
3. **Test Environment Setup:** Setup necessary environments, including development, test, and staging environments.

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

4. **Test Execution:** Develop scripts for automated testing where applicable, execute all test cases, and use bug tracking tools to log the defects.
5. **Defect Reporting:** Document all the testing activities, results, defects after testing.
6. **Test Closure:** analyze the testing process, ensure to meet all requirements, provide an evaluation of the software quality and testing effectiveness.

2. Testing Levels and Techniques

The testing sessions will be conducted 3 times, for Release 1, Release 2 and Final.

- **Release 1:** Mostly Unit test, works are divided according to functions that was said to have been completed at the time this session began.
 - Exit Criteria:
 - **Test Execution:** All planned test cases for the cycle have been executed.
 - **Defect Reporting:** All identified defects have been logged and prioritized.
 - **Defect Closure:** All critical and high-priority defects have been addressed or deferred with acceptable workarounds.

| Test Spec | Responsibility | Test Type |
|--|-----------------------|---|
| Search – In these tabs: Projects list, Requirements, TestCases, TestRuns, TestRun and Results, TestPlans, Issues Reports, Administration. | Nguyễn Lê Anh Chi | Functional/GUI/Security/Compability/Performance |
| Sort – In these tabs: Project list, Requirements, Test cases, Test run and Results, Test plans, Issues | Lê Ngô Song Cát | Functional/GUI/Security/Compability/Performance |
| Details – Display details of these objects: Release, TestCase, TestRun, Issue | Nguyễn Xuân Quỳnh Chi | Functional/GUI/Security/Compability/Performance |
| Details – Display information of these pages: Dashboard, Projects list, một Project, Requirements, Releases, Modules, TestCases, TestRuns, TestRun and Results, TestPlans, Issues, Reports, Administration | Bùi Ngọc Kiều Nhi | Functional/GUI/Security/Compability/Performance |

- **Release 2:** Retest the failed TCs in previous release and conduct new tests for other objects.
 - Exit Criteria:
 - **Test Execution:** All planned test cases for the cycle have been executed.
 - **Defect Reporting:** All identified defects have been logged and prioritized.
 - **Defect Closure:** All critical and high-priority defects have been addressed or deferred with acceptable workarounds.

| Test Spec | Responsibility | Test Type |
|---------------------|-------------------|---|
| CRUD-Test Plan | Nguyễn Lê Anh Chi | Functional/GUI/ Compability/Performance |
| CRUD-Administration | Nguyễn Lê Anh Chi | Functional/GUI/Security/Compability/Performance |

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

| | | |
|------------------|-----------------------|---|
| CRUD-Test Run | Lê Ngô Song Cát | Functional/GUI/Security/Compability/Performance |
| CRUD-Requirement | Nguyễn Xuân Quỳnh Chi | Functional/GUI/Security/Compability/Performance |
| CRUD-Issue | Nguyễn Xuân Quỳnh Chi | Functional/GUI/Security/Compability/Performance |
| Add -Release | Nguyễn Xuân Quỳnh Chi | Functional/GUI/Security/Compability/Performance |
| Add -Report | Bùi Ngọc Kiều Nhi | Functional/GUI/Security/Compability/Performance |
| Add-Module | Bùi Ngọc Kiều Nhi | Functional/GUI/Security/Compability/Performance |

- **Final:** Retest the failed TCs from previous releases and follow Pareto thesis, mostly System testing.
 - Exit Criteria:
 - **Test Execution:** All planned test cases for the cycle have been executed.
 - **Defect Reporting:** All identified defects have been logged and prioritized.
 - **Defect Closure:** All critical and high-priority defects have been addressed or deferred with acceptable workarounds.

| Test Spec | Responsibility | Test Type |
|--|-----------------------|---|
| Sign Up – Register a new account as user | Nguyễn Lê Anh Chi | Functional/API/Performance |
| CRUD-Test Plan | Nguyễn Lê Anh Chi | Functional/GUI/Performance/API |
| CRUD-Administration | Nguyễn Lê Anh Chi | Functional/ Security |
| | Lê Ngô Song Cát | Functional/GUI/Performance/API |
| CRUD-Issues | Nguyễn Xuân Quỳnh Chi | Functional/GUI/Security/Compability/Performance |
| CRUD-Releases | Nguyễn Xuân Quỳnh Chi | Functional/GUI/ Compability |
| CRUD-Test Cases | Nguyễn Xuân Quỳnh Chi | Functional/GUI/ Compability |
| Add - Requirement | Bùi Ngọc Kiều Nhi | Functional/GUI/Performance/API |
| Add -Report | Bùi Ngọc Kiều Nhi | Functional/GUI/Performance/API |
| Add-Module | Bùi Ngọc Kiều Nhi | Functional/GUI/Performance/API |

3. Test Results

3.1 Release 1

- **Basic Information:**

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

- Date of Test: 16/06/2024
- Test Level: Unit Tests
- Test Types: GUI, Functional, Security, Compability, Performance.
- **Result:**
 - Total TCs: 31
 - Passed: 21
 - Failed: 5
 - Blocked: 5

3.2 Release 2

- **Basic Information:**
 - Date of Test: 08/07/2024
 - Test Level: Unit, Integration testing.
 - Test Types: GUI, Functional, Security, Compability, Performance.
- **Result:**
 - Total TCs: 93
 - Passed: 39
 - Failed: 48
 - Blocked: 3
 - Skipped: 3

3.3 Final

- **Basic Information:**
 - Date of Test: 08/2024
 - Test Level: Integration, System Testing.
 - Test Types: GUI, Functional, Security, Compability, Performance, API.
- **Result:**
 - Total TCs: 31
 - Passed: 21
 - Failed: 5

4. Outline of Planned Tests

4.1 Outline of Test Inclusions

4.1.1 Functional Testing

- **Unit Tests:** Verify individual components for expected behavior. hese tests ensure that each component of the software operates correctly in isolation.
- **Integration Tests:** Ensure combined components interact correctly and validate data flows between them. These tests verify that different modules or services used by your application work well together.
- **System Tests:** Validate the entire system's functionality in an end-to-end environment. This ensures that the system is functioning correctly.

4.1.2 Performance Testing

- **Load Testing:** Assess system performance under expected load conditions. This helps determine how the system behaves under normal usage.
- **Stress Testing:** Determine the system's robustness by testing beyond normal operational capacity. This identifies the system's breaking point and how it behaves under extreme conditions.

4.1.3 Security Testing

- **Vulnerability Scanning:** Identify security vulnerabilities in the system. This involves automated tools to scan for known vulnerabilities.
- **Authentication and Authorization Testing:** Ensure proper implementation of user access controls. This

| | |
|-------------------------------|------------------|
| Share Bug | Version: 1.3 |
| <Iteration/ Master> Test Plan | Date: 27/08/2024 |
| MTP-01 | |

verifies that only authorized users can access certain features or data.

4.1.4 Usability Testing

- **User Interface (UI) Testing:** Verify that the UI meets design specifications and is user-friendly. This involves testing the interface to ensure it is intuitive and easy to use.
- **User Experience (UX) Testing:** Gather user feedback on the overall experience with the system. This helps identify areas of improvement from a user's perspective.

4.1.5 Compatibility Testing

- **Browser Compatibility:** Test the product on different web browsers to ensure consistent behavior and appearance.
- **Device Compatibility:** Ensure the product works across various devices (mobile, tablet, desktop). This ensures that the application is accessible and functional on different types of hardware.

4.1.6 Regression Testing

- **Automated Regression Tests:** Use automated tools to run previously successful tests to ensure new changes haven't introduced errors. This helps in quickly identifying issues caused by recent changes.
- **Manual Regression Tests:** Perform manual checks on critical functionality or those that are too complex for automated tests.

5. Resources and Schedules

5.1 Resources

| Software Element Name | Type and Other Notes |
|---------------------------------------|----------------------|
| Windows 8 and above | Operating System |
| MS Edge, GG Chrome and other browsers | Internet Browser |

5.2 Schedules

| Milestone | Planned Date | Start | Actual Date | Start | Planned Date | End | Actual Date | End |
|-----------------------------------|--------------|-------|-------------|-------|--------------|-----|-------------|-----|
| Iteration Plan agreed | 20/06/2024 | | 15/06/2024 | | 21/06/2024 | | 21/06/2024 | |
| Iteration starts | 21/06/2024 | | 22/06/2024 | | 21/06/2024 | | 21/06/2024 | |
| Requirements baselined | 22/06/2024 | | 20/06/2024 | | 28/06/2024 | | 28/06/2024 | |
| Architecture baselined | 29/06/2024 | | 20/06/2024 | | 05/07/2024 | | 05/07/2024 | |
| User Interface baselined | 23/06/2024 | | 20/06/2024 | | 31/06/2024 | | 31/06/2024 | |
| Release 1 delivered to test | 01/07/2024 | | 05/07/2024 | | 02/07/2024 | | 06/07/2024 | |
| Release 1 accepted into test | 02/07/2024 | | 07/07/2024 | | 03/07/2024 | | 07/07/2024 | |
| Release 1 test cycle finishes | 03/07/2024 | | 10/07/2024 | | 15/07/2024 | | 15/07/2024 | |
| Release 2 delivered to test | 26/07/2024 | | 26/07/2024 | | 27/07/2024 | | 29/07/2024 | |
| Release 2 accepted into test | 28/07/2024 | | 28/07/2024 | | 29/07/2024 | | 30/07/2024 | |
| Release 2 test cycle finishes | 30/07/2024 | | 29/07/2024 | | 10/08/2024 | | 10/08/2024 | |
| Final Release delivered to test | 10/08/2024 | | 08/08/2024 | | 10/08/2024 | | 08/08/2024 | |
| Final Release accepted into test | 11/08/2024 | | 11/08/2024 | | 12/08/2024 | | 11/08/2024 | |
| Final Release test cycle finishes | 13/08/2024 | | 11/08/2024 | | 25/08/2024 | | 26/08/2024 | |
| Iteration Assessment review | 25/08/2024 | | 25/08/2024 | | 26/08/2024 | | 27/08/2024 | |

| | | | | |
|-------------------------------|------------|------------------|------------|------------|
| Share Bug | | Version: 1.3 | | |
| <Iteration/ Master> Test Plan | | Date: 27/08/2024 | | |
| MTP-01 | | | | |
| | | | | |
| Iteration ends | 26/08/2024 | 26/08/2024 | 27/08/2024 | 27/08/2024 |