



FPT SOFTWARE

FTW Test Plan

Project Code: FTW

Document Code: FTW_TestPlan_v1.1

Hanoi, Oct 2022

RECORD OF CHANGE

*A - Added M - Modified D - Deleted

Effective Date	Changed Items	A* M, D	Change Description	New Version
30/10/22	Test Plan	A	Create Test Plan	1.0
04/11/22	Test Strategy	M	Modified Test Strategy	1.1

SIGNATURE PAGE

ORIGINATOR: Hoang Xuan Bach

Oct 30 2022

Test Leader

REVIEWERS: Hoang Xuan Bach

Nov 02, 2022

Developer

Hoang Xuan Bach

Nov 02, 2022

Tester

Hoang Xuan Bach

Nov 03,2022

Project Manager**APPROVAL:**

Hoang Xuan Bach

Nov 05,2022

Project Manager

TABLE OF CONTENTS

1	INTRODUCTION	5
1.1	Purpose	5
1.2	Background information	6
1.3	Scope of testing	6
1.4	Constraints	7
1.5	Risk list	8
2	REQUIREMENTS FOR TEST	11
3	TEST STRATEGY	13
3.1	Test types	13
3.1.1	<i>Functional Testing</i>	13
3.1.2	<i>User Interface Testing</i>	13
3.1.3	<i>Performance testing</i>	13
3.1.4	<i>Security Testing</i>	14
3.1.5	<i>Regression Testing</i>	15
3.2	Test stage	16
3.3	Tools	16
4	RESOURCE	17
4.1	Human Resource	18
4.2	Hardware and Software	18
5	TEST MILESTONES	20
6	DELIVERABLES	24

1. INTRODUCTION

1.1 Purpose

This is the comprehensive test plan of the FTW project. The purpose of the document describes scopes of tests and activities which need to be taken during the test process of the project. It addresses the following items:

- Scopes of Testing
- Requirements for Testing
- Test Strategy
- Test Resources
- Test Milestones
- Test Deliverables

1.2 Definitions, Acronyms, and Abbreviations

Abbreviations	Description
AT	Acceptance Test
IT	Integration Test
ST	System Test
UT	Unit Test
HR	Human Resource
GUI	Graphical User Interface
BVA	Boundary Value Analysis
EP	Equivalence Partitioning

1.3 References

Title/File name	Author	Version	Effective Date
G4.pdf (FTW Software Requirement Specification)	FTW Development Team	1.0	Dec 2021
Template_Test-plan.doc			

1.4 Background information

FTW is a software system that helps learners find tutors in different subjects. The software is targeted to help users including tutors and students connect with each other.

For students, they have the ability to send requests to the tutor they want to study.

For tutors, they can accept the student's request for subjects that students sent.

For Admin, Admin will manage accounts of tutors.

1.5 Scope of testing

Test stage includes Unit Testing, Integration Testing, System Testing and User Acceptance Testing. Test types included in those test stages are Functional Testing, Performance Testing, User Interface Testing, Security Testing, Regression Testing

Target of test

Functional items and Non-functional items will be verified and passed by FTW test team, then be validated and approved by Project Manager, including the requirements of the following primary functions:

1. Manage Authentication
2. Manage users requests
3. Manage users profiles
4. Manage feedback
5. Manage account
6. Notification

Test stage

No	Test Stages	Description
1	Unit/Component Test	Unit Test will be performed by FTW development team
2	Integration Test	IT will be performed by the FTW development team.

No	Test Stages	Description
		<p>After the UT is finished, development team will execute the UT Quality Gate/Exist Criteria based on the UT Gate checklist for each function. Integration Test will only start if the result of UT Quality Gate is Passed.</p> <p>This test stage focuses on specific areas of use cases when all requirements are completed, integration test should be performed to ensure all components incorporate well</p>
3	System Test	ST will be executed by the FTW to validate that functions and system interfaces perform properly in the production/ST environment.
4	Acceptance Test	Acceptance Test will be conducted by FTW testing team to validate business processes that base on user requirements.

Test assumption

The following assumptions are made for test process:

Group 2 is to validate and approve for final software product, test procedures and results

Verification from FTW project team for test execution, documenting and results

FTW leader to approve and commit for test resources

Requirements for test are limited to functional and non-functional requirements specified in Section 2 of this document

Test will be executed on specific hardware and software as defined in Section 4.2

1.6 Constraints

Front-end system uses ReactJS which takes up less space and tends to give better

Back-end system uses Spring Boot to develop fast servers in Java.

Front-end web applications should support Chrome browsers.

1.7 Risk list

#	Description	Source	Probability	Exposure	Trigger
1	Skills of Tester	Skill	0.4	5	Complexities involved in testing the applications
2	The Testing environment	Other resources	0.3	6	The Testing environment not be setup properly
3	Project budget may be over because of business situation	Other resources	0.3	3.2	
4	One or more members in the FWT testing team are unable to participate due to personal reasons.	HR	0.3	7.5	Work plan is delayed by 30%
5	Improper communication among team members	Organization	0.3	3	Work plan is delayed by 30%
6	Continually changing requirements	Technical	0.4	6	Work plan is delayed by 25%
7	Conflicting test priorities	Operation	0.2	4	Work plan is delayed by 25%

1.8 Training needs

No	Training items	Description
1	Unit Test	<p>JUnit is a Testing Framework used by developers to implement unit testing in Java, and accelerate programming speed and increase the quality of code.</p> <p>Learn & study from internal: Members in FTW Development team</p> <p>Learning from internet resources:</p> <p>https://www.tutorialspoint.com/junit/junit_test_framework.htm</p>
2	Performance test	<p>JMeter is a testing tool used for analyzing and measuring the performance of web applications.</p> <p>Learn & study from external:</p> <p>https://www.tutorialspoint.com/jmeter/index.htm</p>
3	Security Test	<p>Wireshark Testing tool: Testers use it to point out what is happening with the network and to assess traffic for vulnerabilities in real time.</p> <p>Learn & study from external:</p> <p>https://www.youtube.com/watch?v=zOYohNOnWp4</p>

2. REQUIREMENTS FOR TEST

a. Functional Items

<ol style="list-style-type: none"> 1. Manage Authentication <ol style="list-style-type: none"> 1.1. Login 1.2. Logout 1.3. Register 2. Manage Requests <ol style="list-style-type: none"> 2.2. Create requests 2.3. Edit requests 2.4. Delete requests 2.5. Invite tutor 2.6. View my request (student) 2.7. View my request (tutor) 2.8. View list tutor in my request 2.9. Search request 2.10. Offer to teach 2.11. Cancel offer 2.12. Accept/Reject student invitation 2.13. View all current request 2.14. View request detail 2.15. Filter request 3. Manage Profile <ol style="list-style-type: none"> 3.1. Change password 3.2. View tutor profile 3.3. View list tutor 3.4. Search tutor 3.5. Edit tutor 	<ol style="list-style-type: none"> 4. Manage Feedback <ol style="list-style-type: none"> 4.1. View feedback 4.2. View rate 4.3. Write feedback for tutor 4.4. Edit feedback 4.5. Delete feedback 4.6. Rate by star 5. Manage Account <ol style="list-style-type: none"> 5.1. Approve tutor's account 6. Other <ol style="list-style-type: none"> 6.1. View home page 6.2. Mail notification 6.3. Update calendar
---	--

b. Non-functional items

- The language is Vietnamese
- User interface should be clear and easy to use.
- Main functions are organized into tabs for easier access.
- Links, buttons, and checkboxes are easily clickable.
- Front-end web applications should support Chrome browsers.

- The system must be available at least 95% of the time, excluding scheduled maintenance servers. The system function is carefully observed and learned from actual needs of users (student, tutor)
- Front-end system uses ReactJS which takes up less space and tends to give better performance than other frameworks.
- Back-end system uses Spring Boot to develop fast servers in Java.
- The system shall display confirmation messages to users within an average of 3 seconds and a maximum of 5 seconds after the user submits information to the system
- The tutor will have personal information verified before being allowed to teach
- The system will have different types of users and every user has access constraints

3. TEST STRATEGY

3.1 Test types

3.1.1 Functional Testing

Test Objective:	The main objective of functional testing is to validate the functionality of the software and ensure that it is in compliance with the software requirement specification.
Technique:	Testers will design test cases using decision/extend decision table technique and combine with EP, BVA for main and important functions. Tester will use blackbox testing for testing. All navigation, interaction, screen will be tested to confirm that all flows are correct without knowing source code. Perform manual tests by hand. Test results will be recorded in the testcase. Errors found will be logged in the jira software.
Completion Criteria:	All the business rules of the application match with the requirement and there are no bugs found in the application that are related to navigation, interaction, screen.
Special Consideration:	Functional testing will not be started in case developers have not executed unit tests before passing applications to testers.

3.1.2 User Interface Testing

Test Objective:	Navigation through the target-of-test properly reflects business functions and requirements, including screen to screen, field-to-field, and use of access methods. Objects and characteristics, such as menus, size, position, state, and focus conform to standards.
Technique:	Testers will create specific test scenarios against UI prototypes approved by customers. Testers execute tests based on test scenarios and create reports. Common defects will be collected for improved checklists. Execute each case, using valid, invalid and boundary data, to verify the expected results displayed when valid, invalid and boundary data is used.

Completion Criteria:	All GUI test cases have been executed and passed Any defects related to GUI are fixed, except those accepted by customer
Special Consideration:	N/A

3.1.3 Performance Testing

Test Objective:	Evaluating how a system performs in terms of responsiveness and stability under a particular workload
Technique:	Testers will create test scenarios, test scripts against the requirements provided by the FTW test team, using the following tool : Jmeter. Testers execute tests based on test scenarios and create reports. Common defects will be collected for improved checklists.
Completion Criteria:	The system shall display confirmation messages to users within an average of 3 seconds and a maximum of 5 seconds after the user submits information to the system
Special Consideration:	N/A

3.1.4 Security Testing

Test Objective:	To identify the threats in the system. To measure the potential vulnerabilities of the system. To help in detecting every possible security risks in the system
Technique:	Testers will create test scenarios against the requirements. Test scenarios will be created based on black box test technique and using WireShark. Testers execute tests based on test scenarios and create reports. Use WireShark tool to validate whether a transaction is encrypted or not. Common defects will be collected for improved checklists.

Completion Criteria:	All test cases have been executed to verify proper data acceptance, processing, and retrieval, and the appropriate implementation and passed
Special Consideration:	N/A

3.1.5 Regression Testing

Test Objective:	Check that the new code changes do not negatively impact the existing developed and tested functionalities of the application.
Technique:	Testers will use regression test selection and test case priority for testing or hybrid to create test cases. Execute each case, using boundary data, to verify the following: Get the expected results when boundary data is used. Data is updated correctly into the database. The appropriate errors or warning messages are displayed when invalid data is used. Each valid data input is updated correctly into the database. Each business rule is properly applied.
Completion Criteria:	All changed code by changing requirements are tested successfully without failures. All new features with their functions are test sucessfully. Specified function test cases have been executed to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the business rules, and passed The appropriate activities will be performed when valid data is used. The corresponding error/warning message mechanism is applied for each specific case. All bugs found must be fixed.
Special Considerations:	N/A

3.2 Test stages

Types of Test	Stages of Test			
	Component/ Unit	Integration	System	Acceptance
Functional Test			x	x
User Interface Test	x	x	x	
Performance Test			x	x
Regression Test			x	
Security Test			x	x

3.3 Tools

Purpose	Tool	Vendor/In-house	Version
Defect log	Jira	Atlassian	
Debugging	Chrome Debugging	Chrome	106.0.5249.110
Collect test effort	Excel	Microsoft	
Documenting	MS Word, Excel	Microsoft	2019
Unit Testing	JUnit	Netbeans	4.12
Functional Testing	Manual	Manual	
Performance Testing	Jmeter	Jmeter	Apache JMeter 5.0
Security Testing	Wireshark	Wireshark	3.6.5

4. RESOURCES

4.1 Human Resource

This table shows the staffing assumptions for the project.

Worker/Doer	Role	Specific Responsibilities/Comments
Hoang Xuan Bach	Test Leader	Manage test resources and assign test tasks Create Test Plan, Test Cases (ST), Test Scripts (ST) Review Test Data Create Test Reports Approve Test Cases (UT, ST), Test Plan, Test Results, Test Reports Responsibility for AT.
Hoang Xuan Bach	Project Manager	Approval Test plan, Test Result, Test report.
Hoang Xuan Bach	Developer	Review Test Cases (UT, ST) Execute Test(UT, ST) Create UT Cases, UT Reports
Hoang Xuan Bach	Tester	Review Test Cases (ST) Review AT Create Test Data and Execute Test (ST) Report Test Results Update Test case

4.2 Hardwares and Softwares

Hardware

Hardware	OS Version
Web Server	Apache Tomcat
CPU	Intel Core i7 11th
Ram	64GB

Storage	2TB
Client	Window OS

Software

Software	Version
Apache Tomcat Server	8.0.27
Microsoft SQL Server	2019 - Developer
Chrome	106.0.5249.110
React Developer Tool	4.25.0
Visual Studio Code	1.72

5. TEST MILESTONES

Milestone Task	Effort	Start Date	End Date
Create Test Plan	6	30/10/2022	1/11/2022
Review Test Plan	2	02/11/2022	03/11/2022
Update Test Plan	2	04/11/2022	05/11/2022
Deliver Test Plan	0	05/11/2022	05/11/2022
Package version 1.0		05/11/2022	05/12/2022
Create Test Case (UT, ST)	6	05/11/2022	08/11/2022
Review Test Case	2	09/11/2022	10/11/2022
Update Test Case	2	11/11/2022	12/11/2022
Execute UT	2	13/11/2022	14/11/2022
Waiting for UT fix bug	2	15/11/2022	16/11/2022
Re-UT again	1	16/11/2022	16/11/2022
Execute IT	2	17/11/2022	18/11/2022
Waiting for IT fix bug	2	19/11/2022	20/11/2022
Execute ST	2	21/11/2022	22/11/2022
Waiting for ST fix bug	2	23/11/2022	24/11/2022
Re-ST again	1	25/11/2022	25/11/2022
Execute RT	2	26/11/2022	27/11/2022
Waiting for ST fix bug	2	28/11/2022	29/11/2022
Execute AT	2	30/11/2022	01/12/2022
Review AT	2	02/12/2022	03/12/2022
Create Test Report for Software Package version 1.0	1	04/12/2022	04/12/2022
Test Deliver for Software Package version 1.0	0	05/12/2022	05/12/2022
Package version 1.1		06/12/2022	06/01/2023
Create Test Case (UT, ST)	6	06/12/2022	09/12/2022
Review Test Case	2	10/12/2022	11/12/2022
Update Test Case	2	12/12/2022	13/12/2022
Execute UT	2	14/12/2022	15/12/2022
Waiting for UT fix bug	2	16/12/2022	17/12/2022

Execute IT	2	18/12/2022	19/12/2022
Waiting for IT fix bug	2	20/12/2022	21/12/2022
Re-UT again	1	22/12/2022	22/12/2022
Execute ST	2	23/12/2022	24/12/2022
Waiting for ST fix bug	2	25/12/2022	26/12/2022
Re-ST again	1	27/12/2022	27/12/2022
Execute RT	2	28/12/2022	29/12/2022
Waiting for ST fix bug	2	30/12/2022	31/12/2022
Execute AT	2	01/01/2023	02/01/2023
Review AT	2	03/01/2023	04/01/2023
Create Test Report for Software Package version 1.1	1	05/01/2023	05/01/2023
Test Deliver for Software Package version 1.1	0	06/01/2023	06/01/2023
Package version 1.2		07/01/2023	09/02/2023
Create Test Case (UT, ST)	6	07/01/2023	10/01/2023
Review Test Case	2	11/01/2023	12/01/2023
Update Test Case	2	13/01/2023	14/01/2023
Execute UT	2	15/01/2023	16/01/2023
Waiting for UT fix bug	2	17/01/2023	18/01/2023
Re-UT again	1	19/01/2023	19/01/2023
Execute IT	2	20/01/2023	21/01/2023
Waiting for IT fix bug	2	22/01/2023	23/01/2023
Execute ST	2	24/01/2023	25/01/2023
Waiting for ST fix bug	2	26/01/2023	27/01/2023
Re-ST again	1	28/01/2023	28/01/2023
Execute RT	2	29/01/2023	30/01/2023
Waiting for ST fix bug	2	01/02/2023	02/02/2023
Execute AT	2	03/02/2023	04/02/2023
Review AT	2	05/02/2023	06/02/2023
Create Test Report for Software Package version 1.2	1	07/02/2023	08/02/2023
Test Deliver for Software Package version 1.2	0	09/02/2023	09/02/2023

Execute System Test for Software Package final version	2	10/02/2023	11/02/2023
Create Test Report for Software package final version	1	12/02/2023	13/02/2023
Test Deliver for Software Package final version	0	12/02/2023	13/02/2023

6. DELIVERABLES

No	Deliverables	Delivered Date	Delivered by	Delivered to
1	FTW Test Plan	05/11/2022	FTW Development Team	Lecturer
2	Test Report Deliver for Software Package version 1.0	05/12/2022	FTW Development Team	Lecturer
3	Test Reports Deliver for Software Package version 1.1	06/01/2023	FTW Development Team	Lecturer
4	Test Reports Deliver for Software Package version 1.2	09/02/2023	FTW Development Team	Lecturer
5	Test Reports Deliver for Software Package final version	13/02/2023	FTW Development Team	Lecturer