|  |
| --- |

**FPT Cinema Project**

**Test Plan**

**Project Code: FCP**

**Document Code: FCP\_Test Plan\_v1.0**

**Hanoi, January 2024**

**RECORD OF CHANGE**

\*A - Added M - Modified D - Deleted

| **Effective Date** | **Changed Items** | **A\* M, D** | **Change Description** | **New Version** |
| --- | --- | --- | --- | --- |
| 08/03/2024 | Document Testing | A | Create new document | 1.0 |
| 10/03/2024 | Background information | A | Create background information | 1.0 |
| 11/03/2024 | Functional Items | A | Add functional items | 1.0 |
| 12/03/2024 | Security Testing | A | Add security testing | 1.0 |
| 12/03/2024 | Training Needs | A | Create training needs |  |
| 12/03/2024 | Constraints | A | Create new Constraints | 1.0 |
| 13f/03/2024 | Load Testing | A | Create new Load Testing | 1.0 |
| 12/03/2024 | Test Assumption | A | Create new test assumption | 1.0 |
| 15/03/2024 | Risk List | M | Modified risk list | 1.0 |
| 12/03/2024 | User Interface Testing | M |  | 1.0 |
| 16/03/2024 | Regression Testing | A | Add regression testing | 1.0 |

**SIGNATURE PAGE**

**ORIGINATOR:** Pham Hong Phong Jan 26, 2024

Test Leader

**REVIEWERS:** Tran Quang Tung Jan 26, 2024

Quality Assurance

Nguyen Quang Minh Jan 26, 2024

Project Manager

**APPROVAL:** Nguyen Quang Minh Jan 26, 2024

Project Manager

**TABLE OF CONTENTS**

[**1.1 Purpose 5**](#_2et92p0)

[**1.2 Definitions, Acronyms, and Abbreviations 5**](#_tyjcwt)

[**1.3 References 6**](#_3dy6vkm)

[1.4 Background information 6](#_1t3h5sf)

[1.5 Scope of testing 6](#)

[A. Target of Test 6](#)

[Functional items will not be tested due to incomplete functionality: 7](#_g0bgduhojobf)

[*B. Test Stage 7*](#_17dp8vu)

[*C. Test Assumption 8*](#_3rdcrjn)

[1.6 Constraints 8](#)

[1.7 Risk List 9](#)

[1.8 Training Needs 9](#_35nkun2)

[*2.1 Test Items 10*](#_44sinio)

[A. Functional Items 10](#)

[B. Non functional Items (Minh) 10](#)

[2.2 Acceptance Test Criteria 11](#_z337ya)

[**3.1 Test Types 11**](#_1y810tw)

[A. Function Testing (Tung) 11](#)

[B. User Interface Testing (Minh) 12](#_4i7ojhp)

[C. Load Testing (KHANG) 13](#_2xcytpi)

[D. Security Testing (hiệp) 14](#_1ci93xb)

[E. Regression Testing 15](#_gm1pyuil218j)

[F. Test Stages 16](#_3whwml4)

[**3.2 Tools 17**](#_qsh70q)

[**4.1 Human Resource 17**](#_1pxezwc)

[**4.2 Hardwares and Softwares 18**](#_49x2ik5)

1. **INTRODUCTION**

## *Purpose*

This is the comprehensive test plan of the FPT Cinema 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 |
| --- | --- |

## *Definitions, Acronyms, and Abbreviations*

| **Abbreviations** | **Description** |
| --- | --- |
| AT | Acceptance Test |
| DMS | Defect Management System (a FPT in-house tool) |
| IT | Integration Test |
| ST | System Test |
| PM | Project Manager |
| PTL | Project Technical Leader |
| QA | Quality Assurance |
| TP | Test Plan |
| TC | Test Case |
| TR | Test Report |
| UAT | User Acceptance test |
| UT | Unit Test |
| SSL | Secure sockets layer |
| COPPA | Children's Online Privacy Protection Act |

## *References*

| **Title/File name** | **Author** | **Version** | **Effective Date** |
| --- | --- | --- | --- |
| FCP\_Requirements | FPT | 1.0 | 08/03/2024 |
| RDS Document | Group 2 - SE1762 | 1.0 | 08/03/2024 |
| Test plan Template (provided by lecturer) | Group 2 - SE1762 | 1.0 | 08/03/2024 |

## *Background information*

FPT Cinema is a cinema website that offers a wide selection of movies. Customers can choose movies, choose seats and pay for purchased tickets.

Equipped with state-of-the-art technology, comfortable spaces, FPT Cinema provides a top-notch and fulfilling movie-going experience for customers. This new version has the following main functions:

| * User access * Profile control * Manage account * Manage room * Manage seat | * Room * Seat * Film News * Payment |
| --- | --- |

The purpose of the website is to provide users with an application to book tickets and seats online. In the future, the website will be responsive to help users have a better experience on the phone screen

## *Scope of testing*

The scope of the test will be limited to testing the website desktop applications on desktop and mobile devices.

### Target of Test

Functional items and Non-functional items will be verified and passed by FPT development team, then be validated and approved Supervisor by via test stages, including the requirements of the following primary functions:

| * User access * Profile control * Manage account * Manage room | * Manage seat * Room * Seat * Payment |
| --- | --- |

### Functional items will not be tested due to incomplete functionality:

| * Film News |
| --- |

### 

### 

### Test Stage

| **No** | **Test Stages** | **Description** |
| --- | --- | --- |
| 1 | Unit Test | Unit Test will be performed by the FPT development team. |
| 2 | Integration Test | Integration Test will be performed by the FPT QA team.  After the Unit Test is finished, testers will execute the UT Gate based on the UT Gate checklist for each function. Integration Test will only start if the result of UT Gate is Passed.  This test stage focuses on specific areas of use cases when all requirements are completed, integration tests should be performed to ensure all components are incorporated well. |
| 3 | System Test | System Test will be executed by the FPT QA team.  Testers will perform complete, end-to-end system testing staged in the pre-production environment to validate that functions and system interfaces perform properly in the production environment. |
| 4 | Acceptance Test | **The Acceptance Test will be conducted by FPT Cinema.**  FPT’s responsibilities during Acceptance Test phase are:   * Facilitate completion on the application deployment * Support fixing bugs * Support Final User Acceptance Test |

### Test Assumption

The following assumptions are made for test process:

* **Supervisor is to validate and approve for final software product, test procedures and results**
* Code changes and bug fixes will be communicated promptly by the development team to ensure testing aligns with the latest version of the software.
* Sufficient test data will be available to simulate realistic usage scenarios.
* Test results will be accurately recorded, analysed, and reported to stakeholders in a timely manner.
* The project team will verify the test execution, documentation, and results.
* Requirements for test are limited to functional and non-functional requirements specified in Section 2 of this document

## *Constraints*

The following constraints may apply when testing is performed on system:

* Deadline for testing only can be met if development progress is on time
* Test execution can be performed when system passes Unit Test Inspection
* At least one round of testing must be performed for requirements

## *Risk List*

| # | Description | Source | Probability | Exposure | Trigger |
| --- | --- | --- | --- | --- | --- |
| 1 | Incomplete Test Coverage | Normal Dependencies | 0.6 | 3.6 | Insufficient requirement analysis |
| 2 | Ambiguous or Changing Requirements | Critical Dependencies | 0.8 | 6.4 | Frequent requirement changes |
| 3 | Time Constraints | Critical Dependencies | 0.7 | 4.2 | Delays in development activities |
| 4 | Communication and Collaboration | Normal Dependencies | 0.5 | 2.5 | Inadequate communication channels |

## *Training Needs*

| **No** | **Training items** | **Description** |
| --- | --- | --- |
| 1 | Unit Test | Learn & study from internal: FPT's other teams  Learn & study from external:   1. JUNIT: http://research.microsoft.com/en-us/downloads/d2279651-851f-4d7a-bf05-16fd7eb26559/default.aspx 2. W3school validator: <http://validator.w3.org/mobile/> |
| 2 | System Test | Learn & study from external:   1. Load test: Apache Jmeter, LoadRunner |
| 3 | Security Test | Learn & study from external:   1. Ethereal/WireShark 2. <http://www.wedi.org/snip/public/articles/testing_whitepaper082602.pdf> 3. <http://www.macadamian.com/images/uploads/whitepapers/HIPAA_TestStrategies.pdf> |

1. **REQUIREMENTS FOR TEST**

## *Test Items*

### Functional Items

| 1. Homepage - View film list    * Filter by Status 2. User access - Login 3. User access - Register 4. User access - Forget password 5. User access - Change password 6. Profile control - View profile 7. Profile control - Update profile 8. Film detail - View film detail 9. Film detail - View/choose screening times 10. Film detail - View/choose seat in the room of screening time 11. Checkout - View ticket information 12. Checkout - View payment method 13. Checkout - Confirm pay 14. Ticket History - View booked ticket | 1. Account Management - View account list 2. Account Management - Edit account    * Insert account    * Update account    * Delete account 3. Film Management - View film list 4. Film Management - Edit film list    * Insert film    * Update film 5. Room Management - View room list 6. Room Management - Edit room list    * Insert room    * Update room    * Remove room 7. Screening time management - View screening time list 8. Screening time management - Edit screening time list    * Insert screening time    * Update screening time    * Remove screening time |
| --- | --- |

### Non functional Items

| 1. The system should be able to scale to accommodate an increasing number of users, especially during peak times like popular movie releases. 2. In the normal condition (could be 100 concurrent users or less), each page should load in 4 seconds or less. 3. In the stress condition (could be more than 100 concurrent users), each page should load in 12 seconds or less. 4. The user interface should be intuitive and user-friendly, allowing users to book seats with minimal effort. 5. The codebase should be well-documented and structured to facilitate easy updates and maintenance. |
| --- |

## *Acceptance Test Criteria*

| **No** | **Test Stages** | **Qualified ratios** |
| --- | --- | --- |
| 1 | Unit Test | To pass this stage, all unit test cases must be tested and passed 100%. All defects should be fixed and re-tested. Average of 11 bugs/KLOC. |
| 2 | Integration Test | To pass this stage, all test cases must be tested and passed 100%. All defects should be fixed and re-tested. Average of 4 bugs/KLOC. |
| 3 | System Test | To pass this stage, all test cases must be tested and passed 100%. All defects should be fixed and re-tested. Average of 0.5 bugs/KLOC. |
| 4 | Acceptance Test | Acceptance Test will be conducted and approved by El Camino Hospital. |

|  |  |  |
| --- | --- | --- |
| 1 | Functional requirement | 100% of all requirement defined on part 2.1 A |
| 2 | Non-functional requirement | 100% of all requirement defined on part 2.1 B |

1. **TEST STRATEGY**

## *Test Types*

### Function Testing

| **Test Objective:** | Verify the application and its internal processes by interacting with the FPT Cinema Project (FCP) through its Graphical User Interface (GUI) and analyzing the outputs or results |
| --- | --- |
| **Technique:** | - Testers will create test scenarios against the requirements provided by the customer. Test scenarios will be created based on black box test technique.  - Testers execute tests based on test scenarios and create reports. Common defects will be collected for an improved checklist.  - Execute each case, using valid and invalid data, to verify the following:  + Get the expected results when valid and invalid data is used  + Valid input data is updated correctly into database  + The appropriate errors or warning messages are displayed when invalid data is used  - Execute each case, using boundary data, to verify the following:  + Get the expected results when boundary data is used  + Data is updated correctly to 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 functional 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:** | Functional testing will NOT be started in case of developers have not executed unit test before passing application to testers |

### 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:** | - Creating Test Scenarios: Testers will anddevelop specific scenarios to test against the UI prototypes that have been approved by the customer. These scenarios are designed to cover all aspects of the UI to ensure it meets the agreed requirements.  - Executing Tests: Once the scenarios are created, testers will carry out the tests as per those scenarios. This involves using different types of data:   * Valid data that the system is expected to handle correctly. * Invalid data to test how the system handles unexpected or erroneous inputs.   Boundary data which sits at the upper or lower limits of what is considered valid, to check the system's hling of edge cases.  - Reporting: Testers will document their findings, highlighting any common defects which can then be used to refine future testing checklists. This iterative process helps in enhancing the quality of UI testing over time. |
| **Completion Criteria:** | - All graphical user interface (GUI) test cases must be executed and passed. This ensures that every aspect of the UI has been thoroughly checked and meets the requirements.  - Any defects found during the testing must be fixed, except for those that the customer has agreed to accept.  - All defects identified by testers are to be recorded and addressed in the Defect Management System (DMS). This system tracks and manages defects from identification through to resolution. |
| **Special Considerations:** | Device and Browser Compatibility |

### Load Testing

| **Test Objective:** | * Under normal conditions: With up to 100 users accessing and using the website simultaneously (selecting movies, choosing seats, processing payments, etc.), each page should load in no more than 4 seconds. * Under stress conditions: When user traffic exceeds 100 simultaneous users, each page should load in no more than 12 seconds, ensuring user experience remains uninterrupted during peak times. |
| --- | --- |
| **Technique:** | * Testers will develop test scenarios based on the requirements of the FPT Cinema project, using black-box testing techniques. These scenarios will be supported by specialized testing tools, such as IBM Rational Robot & Manager or equivalent, to simulate user behavior on the website. * Testing will be conducted based on the developed scenarios, collecting data on page load times and system responses under different situations |
| **Completion Criteria:** | * Ensure that in both normal and stress conditions, the requirements for page load times are met. * All performance requirements must be fulfilled, including the ability to smoothly serve users during high traffic volumes. |
| **Special Considerations:** | * The testing tool must be capable of simulating at least 100 virtual users to assess the website's performance under various scenarios. * This version of the "Test Strategy" has been enhanced to reflect the specific objectives of the FPT Cinema project, focusing on ensuring the website can effectively handle user requests, both under normal conditions and when faced with high traffic stress. |

### Security Testing

| **Test Objective:** | Verify that the application is COPPA & SSL compliance |
| --- | --- |
| **Technique:** | - Testers will create test scenarios against the requirements which are based on COPPA & SSL compliance. Test scenarios will be created based on the black box test technique. Refer to:  1.<https://www.ftc.gov/legal-library/browse/rules/childrens-online-privacy-protection-rule-coppa>  2.[https://support.google.com/richmedia/answer/6015286?hl=en#:~:text=The%20secure%20sockets%20layer%20(SSL,site%20must%20also%20use%20SSL.](https://support.google.com/richmedia/answer/6015286?hl=en#:~:text=The%20secure%20sockets%20layer%20)  for more details.  - Testers execute tests based on test scenarios and create reports. Use the WireShark tool to validate whether transactions are encrypted or not. Common defects will be collected for improved checklists.  - Execute each case, using valid and invalid data, to verify the following: The expected results occur when valid COPPA & SSL compliance. |
| **Completion Criteria:** | All test cases have been executed to verify proper data acceptance, processing, and retrieval, and the appropriate implementation of the COPPA & SSL compliance rules, and passed |
| **Special Considerations:** | Compliance with COPPA regulations regarding children's online privacy.  Implementation of SSL to ensure secure transmission of data. |

### Regression Testing

| **Test Objective:** | To ensure that new software updates, bug fixes, or enhancements do not adversely affect existing functionality. This testing aims to identify any unintended side effects caused by modifications to the application. |
| --- | --- |
| **Technique:** | * **Selective Retest Strategy:** Testers selectively choose a subset of test cases that are directly affected by the recent changes, as well as those that cover the most critical functionalities of the application. * **Risk-Based Selection:** Focus on areas with high-impact changes or where the application has shown vulnerability in the past. This includes analyzing the impact of code changes to identify which functionalities are most likely to be affected. * **Automated Regression Suite:** Employ automated testing tools to efficiently run regression tests, especially for repetitive or high-volume tests. This aids in quickly identifying regressions introduced by recent code commits. * **Test Case Prioritization:** Prioritize test cases based on the criticality of application features, recent changes, and historical defect trends. This ensures that the most significant and impactful areas of the application are tested first. |
| **Completion Criteria:** | * All selected test cases have been executed, covering the functionalities affected by recent changes as well as critical areas of the application. * The application processes data correctly and according to business rules, both in new and existing functionalities. * Valid data inputs across the tested functionalities result in appropriate actions and outputs without errors. * Error and warning messages are correctly displayed for invalid operations or data inputs, in accordance with specified requirements. * No significant regressions are introduced, and any new defects found during regression testing are documented and addressed. * The stability and integrity of other functionalities not directly impacted by recent changes are confirmed. |
| **Special Considerations:** | * **Comprehensive Coverage:** While focusing on changed or affected areas, ensure that the regression test suite also includes tests for core functionalities, critical paths, and common use cases to maintain overall application integrity. * **Test Environment:** The testing environment should closely mirror the production environment to ensure accurate and relevant results. This includes configurations, data, and any external integrations. * **Feedback Loop:** Establish a rapid feedback loop between testers and developers to quickly address any defects found. This allows for more efficient remediation and retesting of issues. * **Continuous Improvement:** Regularly review and update the regression test suite to incorporate new test cases for recent features and to remove obsolete or redundant tests. This helps in keeping the test suite effective and relevant. |

### *Test Stages*

| **Types of Test** | **Stages of Test** | | | |
| --- | --- | --- | --- | --- |
| **Unit** | **Integration** | **System** | **Acceptance** |
| Function Test | X | X | X | X |
| User Interface test | X | X | X | X |
| Load test |  |  | X | X |
| Security test | X |  | X | X |
| Regression test | X | X | X |  |

## *Tools*

| **Purpose** | **Tool** | **Vendor/In-house** | **Version** |
| --- | --- | --- | --- |
| Defect log | DMS | FPT tool |  |
| Collect test effort | Timesheet | FPT tool |  |
| Documenting | MS Word, Excel | Microsoft | 2010 |
| Unit Testing | Junit | <open source> | 4.13.2 |
| Function Testing | Manual | Manual |  |
| Load Testing | IBM Rational Robot & Manager | Microsoft | 2010 |
| Security Testing | WireShark | <open source> | 1.6.1 |
| Database | Mysql Workbench | Oracle | 2008 |

1. **RESOURCE**

## *Human Resource*

This table shows the staffing assumptions for the project.

| **Worker/Doer** | **Role** | **Specific Responsibilities/Comments** |
| --- | --- | --- |
| Pham Hong Phong | Test Leader | Manage test resources and assign test tasks  Create Test Plan, Test Cases (IT, ST), Test Scripts (IT, ST)  Review Test Data  Create Test Reports |
| Cao Duc Hiep | Tester | Review Test Cases (IT, ST)  Create Test Data and Execute Test (IT, ST)  Report Test Results  Create UT Cases, UT Reports |
| Tran Quang Tung | Quality Assurance | Final Inspection Test Cases, Test Plan, Test Reports |
| Tran Dinh Duc | Project Manager | Approve Test Cases (UT, IT, ST), Test Plan, Test Results, Test Reports |
| Tran Quang Tung | Developer, Quality Assurance | Final Inspection Test Cases, Test Plan, Test Reports  Create UT Cases, UT Reports |
| Nguyen Sy Khang | Developer | Approve Test Cases (UT, IT, ST)  Create UT Cases, UT Reports |

## *Hardwares and Softwares*

The FPT Cinema Project HTML5 web-based application and iPhone, Android Hybrid applications will require testing on the following desktop devices:

| **Hardware** | **OS Version** |
| --- | --- |
| Windows PC | Dell, HP, Lenovo, and Acer |
| Smart TVs | LG |

| **Software** | **Version** |
| --- | --- |
| Microsoft Windows Server | 2020 |
| Java Development Kit & Integrated Development Environment | JDK 17 |
| Netbean | 17 |
| MySQL | 8.0.26 |

1. **TEST MILESTONES**

| **Milestone Task** | **Effort (pd)** | **Start Date** | **End Date** |
| --- | --- | --- | --- |
| Create Test Plan | 2 | 20/01/2024 | 29/01/2024 |
| Review & Update Test Plan | 1 | 30/01/2024 | 01/02/2024 |
| Create test case for unit, integration test version 1.0 | 1 | 05/02/2024 | 06/02/2024 |
| Execute unit test | 3 | 08/02/2024 | 11/02/2024 |
| Execute integration test | 3 | 12/02/2024 | 13/02/2024 |
| Write unit test and integration report | 0.5 | 25/02/2024 | 25/02/2024 |
| Create test case for unit, integration test version 1.1 | 1 | 27/02/2024 | 28/02/2024 |
| Execute unit test | 3 | 28/02/2024 | 01/03/2024 |
| Execute integration test | 1 | 02/03/2024 | 03/03/2024 |
| Write unit test and integration report version 1.1 | 1 | 05/03/2024 | 06/03/2024 |
| Review SRS and Test documents | 1.5 | 07/03/2024 | 08/03/2024 |
| Write SRS and Test documents report | 0.5 | 09/03/2024 | 10/03/2024 |
| Execute system test | 2.5 | 15/03/2024 | 16/03/2024 |
| Write system test report | 0.5 | 17/03/2024 | 18/03/2024 |

1. **DELIVERABLES**

| **No** | **Deliverables** | **Delivered Date** | **Delivered by** | **Delivered to** |
| --- | --- | --- | --- | --- |
| 1 | Test Plan version 1.0 | 01/03/2024 | Development team | ThanhDT |
| 2 | Test Case version 1.0 | 03/03/2024 | Development team | ThanhDT |
| 3 | Test Reports version 1.0 | 06/03/2024 | Development team | ThanhDT |
| 4 | Test Plan version 1.1 | 08/03/2024 | Development team | ThanhDT |
| 5 | Test Case version 1.1 | 10/03/2024 | Development team | ThanhDT |
| 6 | Test Reports version 1.1 | 12/10/2024 | Development team | ThanhDT |
| 7 | Test Plan version 1.2 | 13/09/2024 | Development team | ThanhDT |
| 8 | Test Case version 1.2 | 15/09/2024 | Development team | ThanhDT |
| 9 | Test Reports version 1.2 | 17/09/2024 | Development team | ThanhDT |