**TEST-SYSTEM CRMSTAR**

**Star Academy – Kaopiz Software**



**Automation Test Plan**

**Created by: Ngan Do Thuy**

**Version: 1.0**

**March 28th , 2025**

**Table of Contents**

[**1.**](#_heading=h.n0lopsc4j6tp) **INTRODUCTION 3**

[**2.**](#_heading=h.sziwy5w45il2) **TEST OBJECTIVES 3**

[**3.**](#_heading=h.p0e5gwbd3qou) **TEST STRATEGY 3**

[***3.1.***](#_heading=h.bhh1foqkmsb2) ***Scope of Testing* 3**

[*3.1.1.*](#_heading=h.nmk6cmqfrkra) *In-scope testing 3*

[*3.1.2.*](#_heading=h.1ua8o6ldmzm7) *Out of scope testing 3*

[***3.2.***](#_heading=h.netan4w2s3fq) ***Test Levels and Test Types* 4**

[***3.2.1.***](#_heading=h.oeohxqhetc5x) ***Test levels* 4**

[***3.2.2.***](#_heading=h.9txiejv2iztr) ***Test types* 4**

[***3.3.***](#_heading=h.yhze6xtcugql) ***Risks* 5**

[**4.**](#_heading=h.ubo8ebf2d2zb) **TEST CRITERIA 5**

[***4.1.***](#_heading=h.j4lo7z3rcso1) ***Entry Criteria* 5**

[***4.2.***](#_heading=h.4leh84igutxy) ***Suspension Criteria* 6**

[***4.3.***](#_heading=h.jjz044j19piu) ***Exit Criteria* 6**

[**5.**](#_heading=h.e0a09gmeyx0k) **RESOURCE PLANNING 6**

[***5.1.***](#_heading=h.hffxh9bxf34) ***System resource* 6**

[***5.2.***](#_heading=h.rmo739h2zo6t) ***Test Resource* 7**

[**6.**](#_heading=h.10mm81qw5trg) **SCHEDULE AND ESTIMATION 7**

[**7.**](#_heading=h.fxn7s02gp61t) **TEST DELIVERABLE 7**

[**8.**](#_heading=h.tc9soykd0jtl) **AUTOMATION TESTING TOOL AND STRATEGY 9**

[***8.1.***](#_heading=h.jdu4pv61gmf) ***Testing process* 9**

[***8.2.***](#_heading=h.iepacfkjaond) ***Automation strategy* 9**

[***8.3.***](#_heading=h.l5ws4232pfad) ***Automation testing framework* 9**

[**9.**](#_heading=h.v298ai2he6tw) **LIFECYCLE OF BUG AND RISK 10**

1. **INTRODUCTION**

The purpose of the project is to build a system that *[project description]*.

The Test Plan determine the scopes, objectives and risk of testing. Defining the overall test approach, the overall framework, the enviroment that will support test activities, resolution for risk and contingency, test criteria, timeline and resource planning.

Build automation test scripts that are used for the regression test phase.

1. **TEST OBJECTIVES**

* Ensure that system functionality works as expected without any critical bugs, high bugs.
* Ensure that the response time when using the system meets expectations.
* Ensure that the integration between modules and systems works as expected.

1. **TEST STRATEGY**
   1. *Scope of Testing* 
      1. *In scope testing*

In-scope items will be system testing for usability of the following features.

The following is the list of features to be tested in this project:

|  |  |
| --- | --- |
| No. | UserStory/ Function |
| 1 | QLKLV – Thêm Mới |
| 2 | QLKLV – Xóa |
| 3 | QLKLV – Sửa |
| 4 | QLKLV – Xem |
| 5 | QLKLV – Tìm Kiếm |
| 6 | QLKLV – Xuất Excel |
| 7 | QLKLV – Phân Trang |

Degree: 1 – Complex, 2 – Medium, 3 – Low

* + 1. *Out of scope testing*

There are some out-of-scope items:

* Database Testing
* Mobile Testing
* Performance Testing
* Security Testing
* API Testing
* Stress Testing
* Data Migration Testing
* Accessibility Testing
* Unit Testing
  1. *Test Levels and Test Types*
     1. *Test levels*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test level** | **Method** | **People in charge** | **Note** |
| 1 | Integration test | Manual/Automation | Tester | * Testing the integration of systems and packages; testing interfaces to external organizations * Test to find defects in the interfaces and the interactions between integrated components or systems |
| 2 | System test | Manual/Automation | Tester | Verify that the whole system meets with requirements |
| 3 | User acceptance test | Manual/Automation | Client | The client tests the system to make sure that the system meets the requirements. To ensure the usability, maintainability, and reliability. |

* + 1. *Test types*

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **No.** | **Test type** | **Method** | **People in charge** | **Notes** |
| 1 | Review requirement (static testing) | Manual | Tester | To verify the correction and level of detail of the User Story |
| 2 | Review code | Automated/Manual | Dev team | Self-review, Peer Review |
| 3 | Functional testing (dynamic testing) | Automated/Manual | Tester/Dev team | Verify that the system meets functional requirements |
| 4 | Non-functional testing | Manual | Tester | Performance test, API test, Security test |
| 5 | Re-testing | Manual | Tester/Dev team | Testing after fixing the bug to confirm the bug is removed |

* 1. *Risks*

|  |  |  |
| --- | --- | --- |
| **No.** | **Risk** | **Mitigation** |
| 1 | Requirement contains a lot of technical aspects related to encryption and module communication, not to mention a huge number of distributed actors, which makes testers difficult to design test cases | * Testers need to review the user story/requirement before writing test cases (usually the first 2 days of each Sprint) * Improve communication in the team * Peer review testcase |
| 2 | The project schedule is tight and short, it may be hard to complete testing on time | * Set priority for test activities: focus testing on the features that contain many critical bugs. * Write a test checklist instead of detailed test cases, step by step, if time is limited. |
| 3 | The system is complex, so deployment may have potential issues. | * In sprint planning, the team needs to clarify the deployment schedule. |
| 4 | The client has delayed the review of test cases. | * The QC team continues executing test cases first and plans to update them later based on the client's feedback. |
| 5 | Reviewing test case feedback is not within the current scope. | * BA needs to reconfirm with the client and then create the enhancement |
| 6 | Using dummy data for testing may lead to issues when working with real data. | * It would be better if the client could provide example data that closely matches real data. |
| 7 | The client is unable to provide the third-party information on time. | * Request the client to provide third-party information before implementing the function. |
| 8 | The build delivery is delayed by the development team. As a result, the QC team may:   * Miss or slip the test plan due to the late build delivery * Be unable to complete tasks outlined in the test plan * Work overtime or reduce the testing scope to meet the deadline. | * Continuously update the build plan to ensure progress stays on track. If any issues arise, inform the relevant teams and reschedule the testing plan accordingly. |
| 9 | The testing environment is not ready for testing. | * Inform the relevant teams(DevOps, IT, or development team) to configure the testing environment before deploying. |
| 10 | The server shuts down during testing when a huge number of users access it. | * Plan and allocate resources for environmental recovery. * Conduct load testing and stress testing to ensure system capacity. |

1. **TEST CRITERIA**
   1. *Entry Criteria*

The entry criteria refer to the conditions to start testing activities (test design, test execution):

1. The specification of a feature is approved and available on Jira before starting test design
2. Development completed (status must be Ready for test)
3. Features under test are already on the QC environment
4. Test-cases, test data are peer-reviewed before test execution
   1. *Suspension Criteria*

The suspension criteria refer to the conditions which test activities will be suspended if suspension criteria are met during testing:

1. Any main flows are not covered
2. The main feature does not meet the acceptance criteria
   1. *Exit Criteria*

The exit criteria are the targeted results of the test that need to be met to complete a test phase:

1. 100% of Test cases executed
2. No Critical, high, or medium bugs remaining
3. 5 % leakage (total bugs found by client / total bugs) : lọt lỗi sang khách hàng: khách hàng tìm được 5 bugs / 100 bugs team dự án tìm
4. **RESOURCE PLANNING**
   1. *System resource*

|  |  |  |
| --- | --- | --- |
| No. | Resources | Description |
| 1 | Server | * Servers: *[list out if have]* |
| 2 | Test tool | * Snagit: Provide evidence for bugs * Jira, Zephyr: Task, bug tracking, Quality monitoring, Test case management, Test execution |
| 3 | Computer | * The PCs with the hardware, software required |
| 4 | **Mobile device** | * *[list out iOS or Android version if have]* |
| 5 | **Web browser** | * *[list out Browser version if have]* |

* 1. *Test Resource*

|  |  |  |  |
| --- | --- | --- | --- |
| **Name** | **Role** | **Start date** | **End date** |
| Tran Thi Hoan | The tester work 50% | 25-Nov, 2021 | 10-Jan, 2022 |
| Ho Tran Nhat Anh | The tester work 100% | 25-Nov, 2021 | 10-Jan, 2022 |
| Pham Thi Dinh | The tester work 100% | 25-Nov, 2021 | 10-Jan, 2022 |

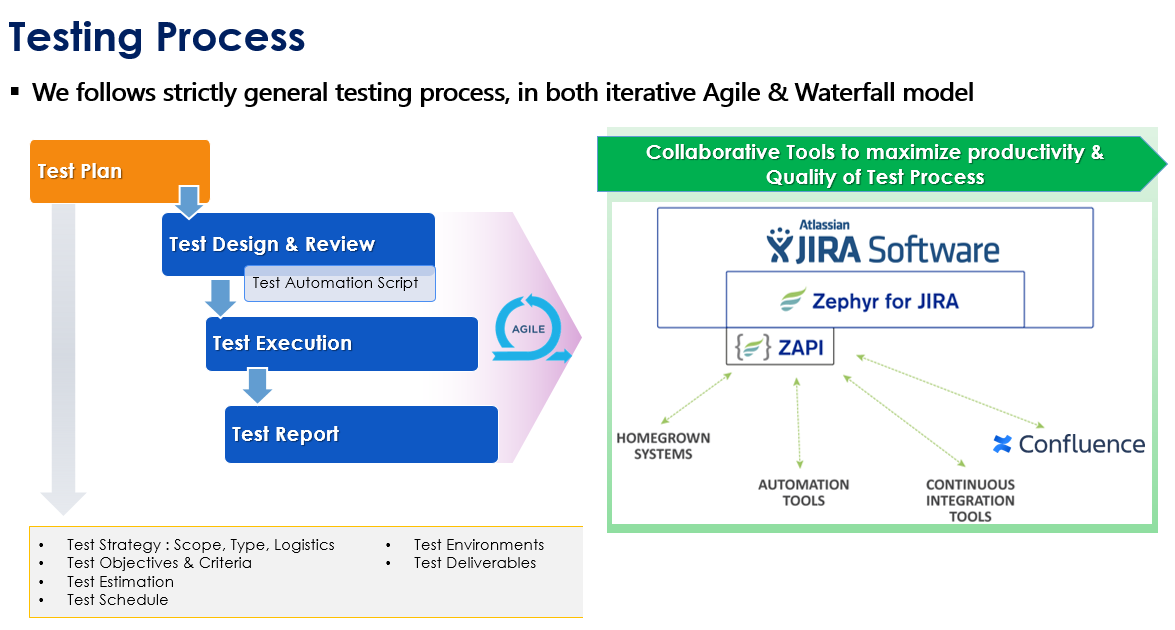
1. **SCHEDULE AND ESTIMATION**

|  |  |  |  |
| --- | --- | --- | --- |
| **No.** | **Task** | **Member** | **Estimation Effort**  **(Man-day)** |
| 1 | Create a test plan | Team | 1 |
| 2 | Study & review requirements |  |  |
| 3 | Create a test design, test cases |  |  |
| 4 | Test execution |  |  |
| 5 | Test report & Quality Control |  |  |

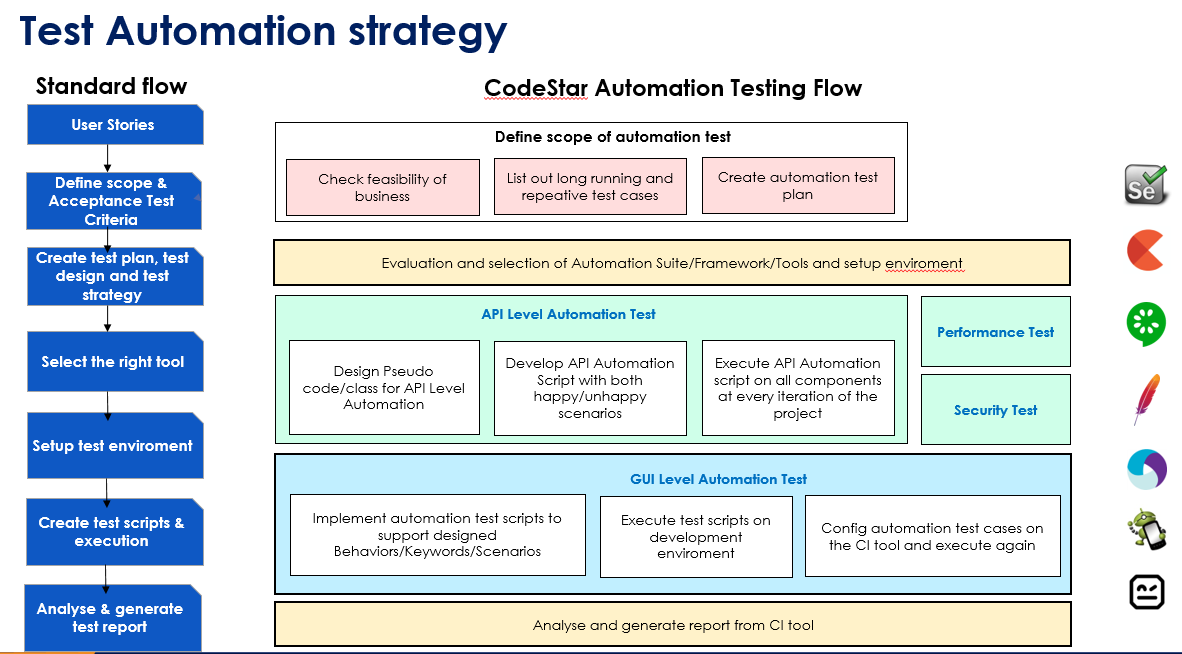
1. **TEST DELIVERABLE**

|  |  |  |
| --- | --- | --- |
| **Phase** | **Deliverable Name** | **How to access** |
| **Before testing** | Test Plan | The test plan will be uploaded to Confluence  <https://conf.codestar.com/display/CDIS/Test+plan> |
| Test-cases documents | All test cases will be uploaded on Confluence and Zephyr, which is an Jira Add-On.  Link Confluence:  <https://conf.codestar.com/display/CDIS/Test+cases>  - Jira filter:  Open Advanced Search, then enter below JQL:  *project = CDIS AND issue type = Test* |
| **During testing** | Test cycles | Each Sprint will have at least 1 test cycle for executing sets of test cases.  Please follow this example to filter the test cycle: |
| **After testing** | Test reports | Test reports will be created after each sprint and published on Confluence: [https://conf. codestar.com/display/CDIS/Test+reports](https://conf.gemvietnam.com/display/CDIS/Test+reports) |
| Bugs | All bugs logged on Jira   * Sprint bugs are sub-tasks of the corresponding User story * Bugs/Improvements are linked to User stories, not added intothe current sprint |

1. **AUTOMATION TESTING TOOL AND STRATEGY**
   1. *Testing process*



* 1. *Automation strategy*



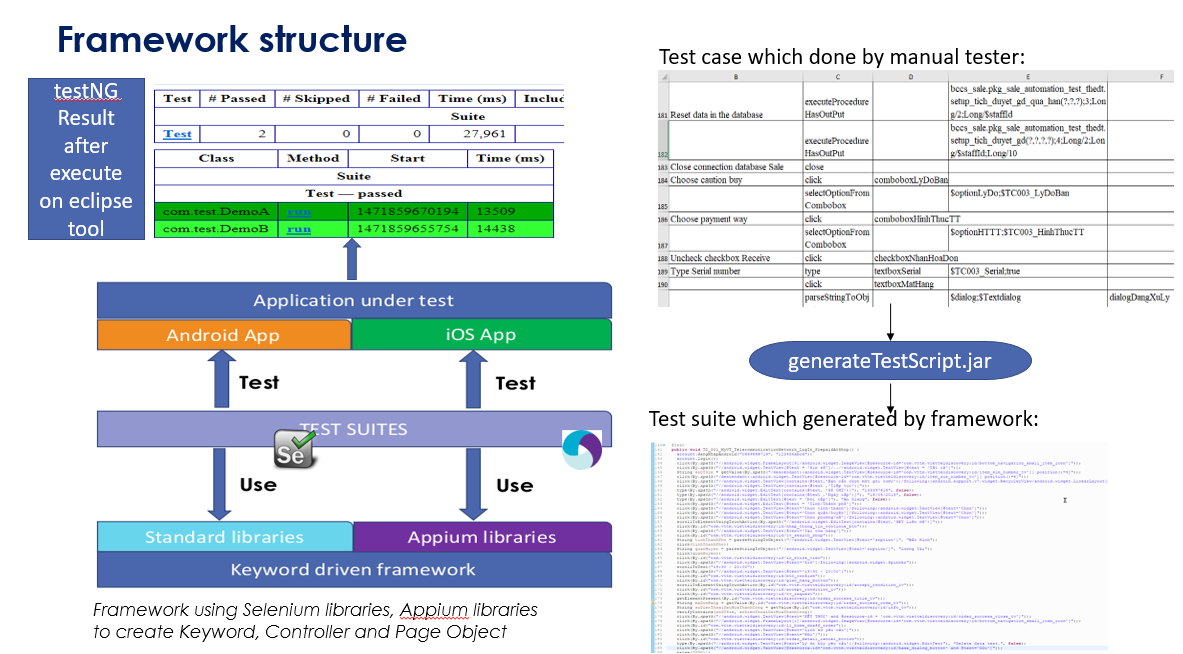
* 1. *Automation testing framework*

**Framework library/ tools:**

* Maven
* TestNG
* Github
* Selenium

**Language**:

* Java
* Python
* C#
* Javascript



1. **LIFECYCLE OF BUG AND RISK**

BUG LIFECYCLE

