

**CAPSTONE PROJECT REPORT**

**Report 5 – Software Test Documentation**

– Hanoi, November 2023 –

**Table of Contents**

[I. Record of Changes 3](#_heading=h.gjdgxs)

[II. Testing Documentation 4](#_heading=h.30j0zll)

[1. Scope of Testing 4](#_heading=h.ymfzma)

[2. Test Strategy 6](#_heading=h.3im3ia3)

[2.1 Testing Types 6](#_heading=h.1xrdshw)

[2.2 Test Stages 8](#_heading=)

[2.3 Supporting Tools 8](#_heading=h.1c1lvlb)

[3. Test Plan 9](#_heading=h.3w19e94)

[3.1 Human Resources 9](#_heading=h.2b6jogx)

[3.2 Test Environment 9](#_heading=h.qbtyoq)

[3.3 Test Milestones and Deliverable 9](#_heading=h.3abhhcj)

[4. Test Cases 10](#_heading=h.1pgrrkc)

[4.1. Unit test 10](#_heading=h.j49s0vn7av83)

[4.2. Integration Test 10](#_heading=h.2c8hzjzgyshp)

[4.3 System Test 10](#_heading=h.widz5giqfpnh)

[4.4 Acceptance Test 10](#_heading=h.f8gwflj6cfgk)

[5. Test Reports 10](#_heading=h.49gfa85)

[5.1 Unit Test Report 10](#_heading=h.v8fj1zke6vvc)

[5.2 Integration Test Report 11](#_heading=h.g7bcqf2lkbfs)

[5.3 System Test Report 12](#_heading=h.5zm0scesmz5)

[5.3 System Test Report 12](#_heading=h.vak8hl3zsfx)

# I. Record of Changes

| **Date** | **A\* M, D** | **In charge** | **Change Description** |
| --- | --- | --- | --- |
| 18/11/2023 | A | Report Create | Create the report item 1, 2, 3 |
| 13/01/2024 | A | Add | Add item 4,5 |
| 20/01/2024 | M | Modify | Modify report |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

\*A - Added M - Modified D - Deleted

# II. Testing Documentation

## 1. Scope of Testing

This section will describe the main targets in the testing phase of the HealCheck System management project including the main features and functions besides the Stages of testing.

| Feature | Function |
| --- | --- |
| User Management | Create User |
| Edit User |
| Disable User |
| Service Management | Add Department |
| Edit Department |
| Disable Department |
| Add Service Type |
| Edit Service Type |
| Disable Service Type |
| Add Service |
| Edit Service |
| Disable Service |
| Medicine Management | Add Medicine Type |
| Edit Medicine Type |
| Disable Medicine Type |
| Add Medicine |
| Edit Medicine |
| Disable Medicine |
| Patient and Medical record Management | Create Patient |
| Create Medical Record |
| View Medical Record |
| Search Patient |
| Edit Medical Record |
| Create Examination Result |
| View Examination Result |
| Create Prescription |
| Invoice Management | View Invoice |
| Search Invoice |

The test stages will be shown in the following [Table 2](https://docs.google.com/document/d/1X0LYhDp8U3h7i5xrAtJQ1yWt_zpK1mPq/edit#heading=h.2i9l8ns)

*Table 2: Test stage*

| Test Stage | In-Charge | Inputs/Time | Acceptance criteria |
| --- | --- | --- | --- |
| Unit test | Developer | Individual code units or modules | 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. |
| Integration test | Tester | Integrated modules or components | 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. |
| Functional System test | Tester | Complete ITSM System | 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 |

The following assumptions are made for test process:

* Verification from SEP490\_G74 testing team for test execution, documenting and results.
* Project managers 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 hardwares and softwares as defined in Section 3.2
* The user interface requirements were reflected in detail in the application's MOCKUP design document. Testers will rely on it to perform manual testing.
* The interface requirements will be properly reflected in the software's functional documentation. Testers will test in the system testing phase.
* Security test requirements will be reflected in the system design and when programming will be reviewed code, UT to test the requirements.

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
* Business Analysis will analyze the customer's performance requirements to clarify the number of concurrent users, server configuration, and bandwidth.

## 2. Test Strategy

### 2.1 Testing Types

Using different test types is given in detail which test type to use at this stage in section 2.2

#### 2.1.1. Component Test

| **Test Objective:** | Component testing is focused on testing individual components, modules, or functions in isolation to ensure their correctness and reliability. |
| --- | --- |
| **Technique:** | - Design UNIT Test cases based on the detailed design documents and Use UT frameworks (xUnit for C#.Net) to write and run all the Unit test and Test Cases  - Examine the internal logic, control flow, and data paths of the code to create test cases that cover various code paths. |
| **Completion Criteria:** | - Achieve a predefined level of code coverage, ensuring that critical portions of the code have been tested.  - All unit tests have been executed, and test results show that the code behaves as expected. |
| **Special Considerations:** | N/A |
|  |  |

### 

#### 2.1.2. API Test

| **Test Objective:** | This testing aims to ensure that the APIs correctly handle requests and responses, effectively communicate with external systems, and maintain data integrity. |
| --- | --- |
| **Technique:** | - API testing will be conducted using a combination of manual and automated testing approaches.  - Automated testing tools, such as Postman and Swagger, will be employed to execute test scenarios, simulate various API requests and responses, and validate the expected outcomes.  - Manual testing will also be performed to assess more complex scenarios and to evaluate the human experience of using the APIs. |
| **Completion Criteria:** | -All API endpoints have been tested, and test cases cover a wide range of scenarios including positive, negative, boundary, and error cases.  -The API documentation is accurate and up-to-date, reflecting the correct usage and behavior of each API endpoint |
| **Special Considerations:** | N/A |

#### 2.1.3. User interface test

| **Test Objective:** | Navigation within the subject under assessment accurately mirrors business operations and specifications, encompassing transitions between screens, fields, and utilization of entry points. Elements and attributes like menus, dimensions, arrangement, condition, and emphasis align with guidelines governing user interface and user experience (UI-UX) standards. |
| --- | --- |
| **Technique:** | - Compare the current interface with a previous version or with similar interfaces in the industry to identify areas for improvement.  - Evaluate the graphical user interface (GUI) elements, such as buttons, checkboxes, radio buttons, and dropdowns, to ensure they behave correctly.  - Verify the flow of navigation between different screens, menus, and features to ensure a smooth user journey. |
| **Completion Criteria:** | - Elements, such as menus, buttons, text fields, and graphics, are positioned and styled correctly according to UI/UX design principles.  - Users can seamlessly move between screens, menus, and features without encountering any unexpected behaviors or broken links |
| **Special Considerations:** | N/A |

#### 

#### 2.1.4. Functional Test

| **Test Objective:** | This type of testing is aimed at verifying that all the components, modules and interactions in the system work seamlessly with each other and meet the specified requirements. |
| --- | --- |
| **Technique:** | * Tester will use black box testing technique * Test cases will be created based on the predefined functions * Only care about how the function works, not about the code inside |
| **Completion Criteria:** | The entire software system functions as expected and meets the specified requirements. |
| **Special Considerations:** | N/A |

#### 

#### 2.1.5. Business Process Test

| **Test Objective:** | Verify the application with business processes to ensure the system functions meet Customer's requirements |
| --- | --- |
| **Technique:** | - Testers will create test scenarios against the requirements provided by the customer. Test scenarios will be created based on black box test technique.  - Processes need to function properly |
| **Completion Criteria:** | Validate that the entire business process, from initiation to completion, functions correctly and accurately within the software.  Ensure that the software correctly follows the defined business rules, validations, and decision points throughout the process. |
| **Special Considerations:** | N/A |

#### 2.1.6 Regression Test

| **Test Objective:** | The primary objective of regression testing is to ensure that new code changes, updates, or enhancements do not adversely affect the existing functionality of the software |
| --- | --- |
| **Technique:** | -Execute the same test cases that were run during previous testing cycles to verify that the existing functionality is still working as expected.  - Focus on testing specific areas of the application that are likely to be affected by the code changes. This approach optimizes testing efforts by targeting high-risk areas.  - Prioritize test cases based on their importance and potential impact on the application. High-priority test cases are tested first. |
| **Completion Criteria:** | -All critical functionalities of the application must continue to work without any regressions or new critical defects.  -Regression testing has covered a representative set of test cases that encompass the key functionalities and affected areas.  -Test cases have been reviewed, updated, or modified as necessary to align with the changes in the application. |
| **Special Considerations:** | N/A |

### 2.2 Test Stages

As mentioned, there are 4 test stages, for each stage we have the test types shown in [Table 3.](#_heading=h.4hr1b5p)

*Bảng 3: Test Stages and Test Types*

| **Type of Tests** | **Test Level** | | |
| --- | --- | --- | --- |
| **Unit** | **Integration** | **System** |
| Component Test | X |  |  |
| API Test |  | X |  |
| User interface test |  | X |  |
| Functional Test |  |  | X |
| Regression Test |  | X | X |

### 2.3 Supporting Tools

| **Purpose** | **Tool** | **Vendor/In-house** | **Version** |
| --- | --- | --- | --- |
| Document(Manage test case and test report) | Microsoft office | Microsoft | 2016 |
| Unit Testing | xUnit | Microsoft | 2.5.1 |
| API test | Swagger | SmartBear Software | 1.0 |
| Functional, Business process and UI test | Google Chrome | Google | 116.0.5845.96 |

## 3. Test Plan

### 3.1 Human Resources

| **Worker/Doer** | **Role** | **Specific Responsibilities/Comments** |
| --- | --- | --- |
| Nguyen Khan Son | Tester | Defining goals, determining workload, allocating resources  Test planning, define procedures for test plan  Review and evaluate the tasks of the test process |
| Truong Quang Son | Tester | Analysis and design Test Cases  Prepare for test Data  Execute test  Test Recording |
| Bui Trung Kien | Test Leader | Analysis and design Test Cases  Prepare for test Data  Execute test  Test Recording |
| Bui Trung Thinh | Tester | Analysis and design Test Cases  Prepare for test Data  Execute test  Test Recording |

### 3.2 Test Environment

| **Hardware** | **OS Version** |
| --- | --- |
| Laptop Dell G3 | Window 10, 8GbRam, Core i5 10300H |
| MacBook Air M1 | macOS Big Sur, 8Gb Ram, Apple M1 |
|  |  |

| **Software** | **Version** |
| --- | --- |
| IIS Server(BE) | IIS 10.0 version 1809 |
| VITE Server(FE) | 4.3.9 |

### 3.3 Test Milestones and Deliverable

| **Milestone Task** | **Start Date** | **End Date** |
| --- | --- | --- |
| Test Plan | 01/11/2023 | 05/11/2023 |
| Unit Test | 13/01/2024 | 15/01/2024 |
| Integration Test | 15/01/2024 | 18/01/2024 |
| Functional System Test | 17/01/2024 | 20/01/2024 |

| **No** | **Deliverables** | **Delivered Date** | **Delivered by** | **Delivered to** |
| --- | --- | --- | --- | --- |
| 1 | Test Plan | 01/11/2023 | SEP490\_Group74 | Lecturer |
| 2 | Unit Test report | 15/01/2024 | SEP490\_Group74 | Lecturer |
| 3 | Integration Test report | 17/01/2024 | SEP490\_Group74 | Lecturer |
| 4 | Functional System Test report | 19/01/2024 | SEP490\_Group74 | Lecturer |
| 5 | Test Documentation | 20/01/2024 | SEP490\_Group74 | Lecturer |

## 4. Test Cases

### 4.1. Unit test

Unit Test test cases will be built by developers, Individual code modules will be tested independently

The unit test test case information is shown in the HCS\_Unit Test Report.xlsx.

### 4.2. Integration Test

With this phase, the small components tested in the unit test phase will be integrated together into APIs, the team will use API test in the integration test stage.

The Integration test test case information is shown in the HCS\_Integration Test Report.xlsx

### 4.3. Functional System Test

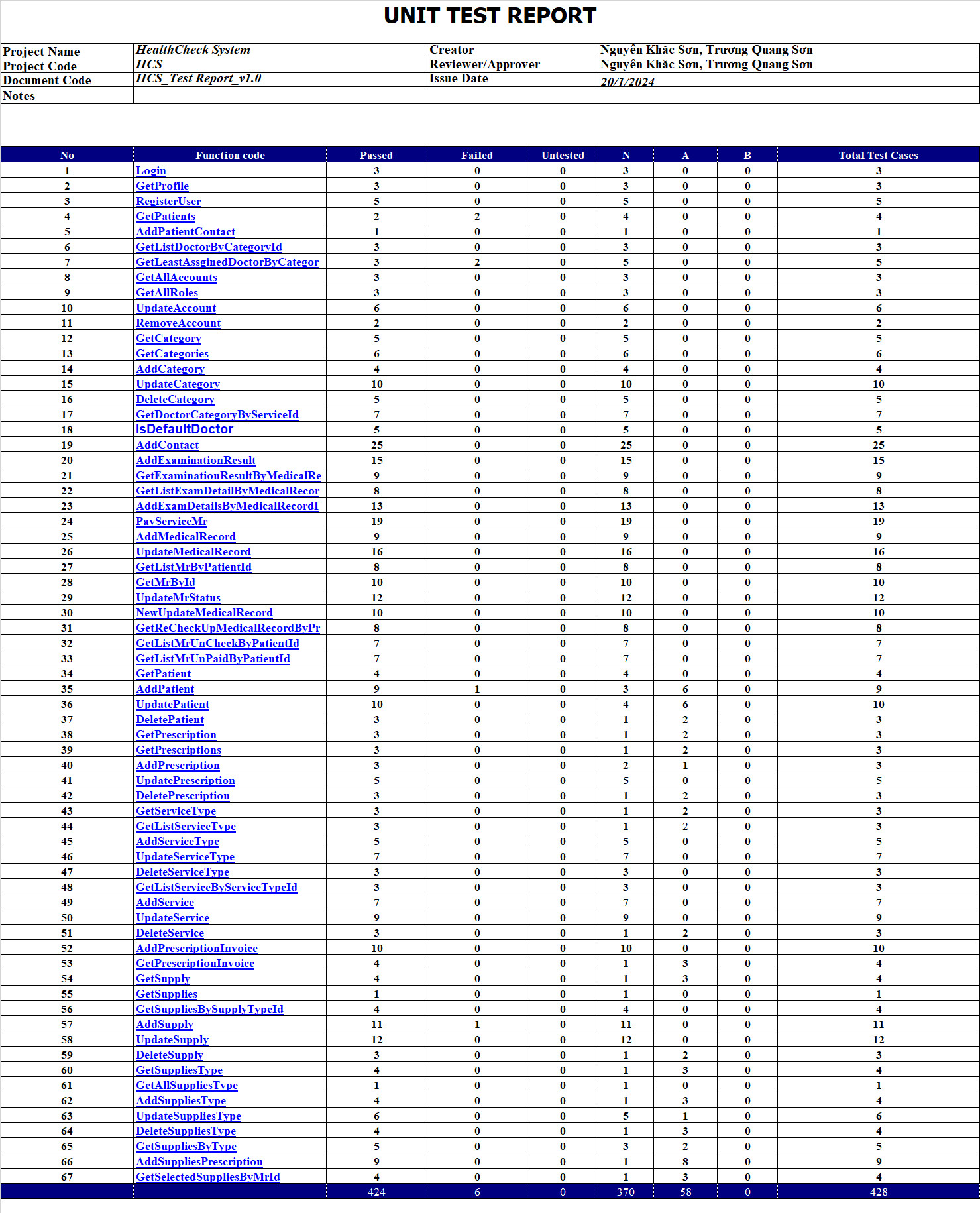
With Stage System testing, the test cases will be built by the testers, then verified by the test leader with the use of Functional Test test type and black box test technique, the tester will mainly focus on testing the system based on the requirements without caring about how the code works.

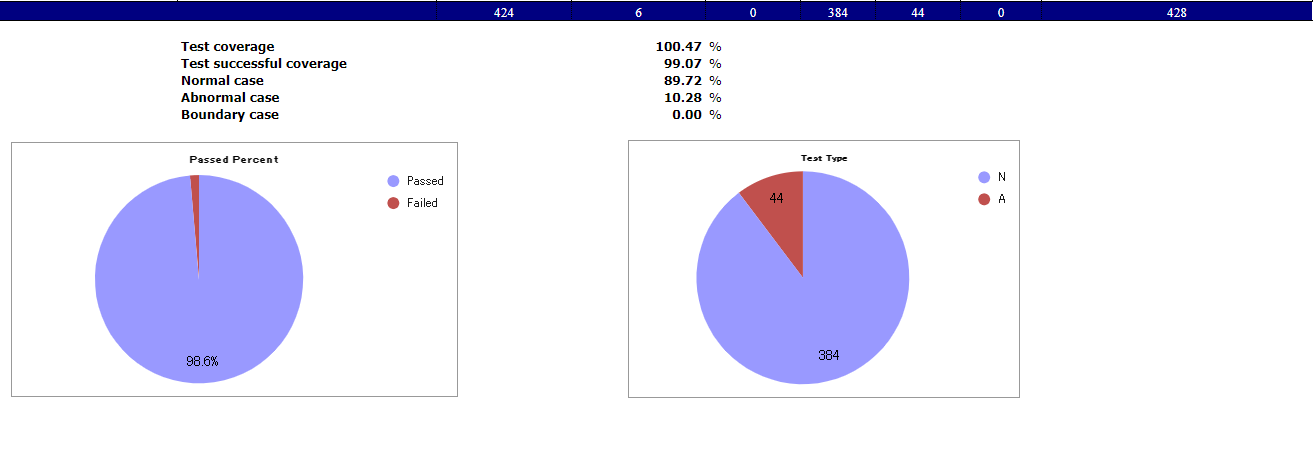
The System test test case information is shown in the HCS\_System Test Report.xlsx

## 5. Test Reports

### 5.1 Unit Test Report

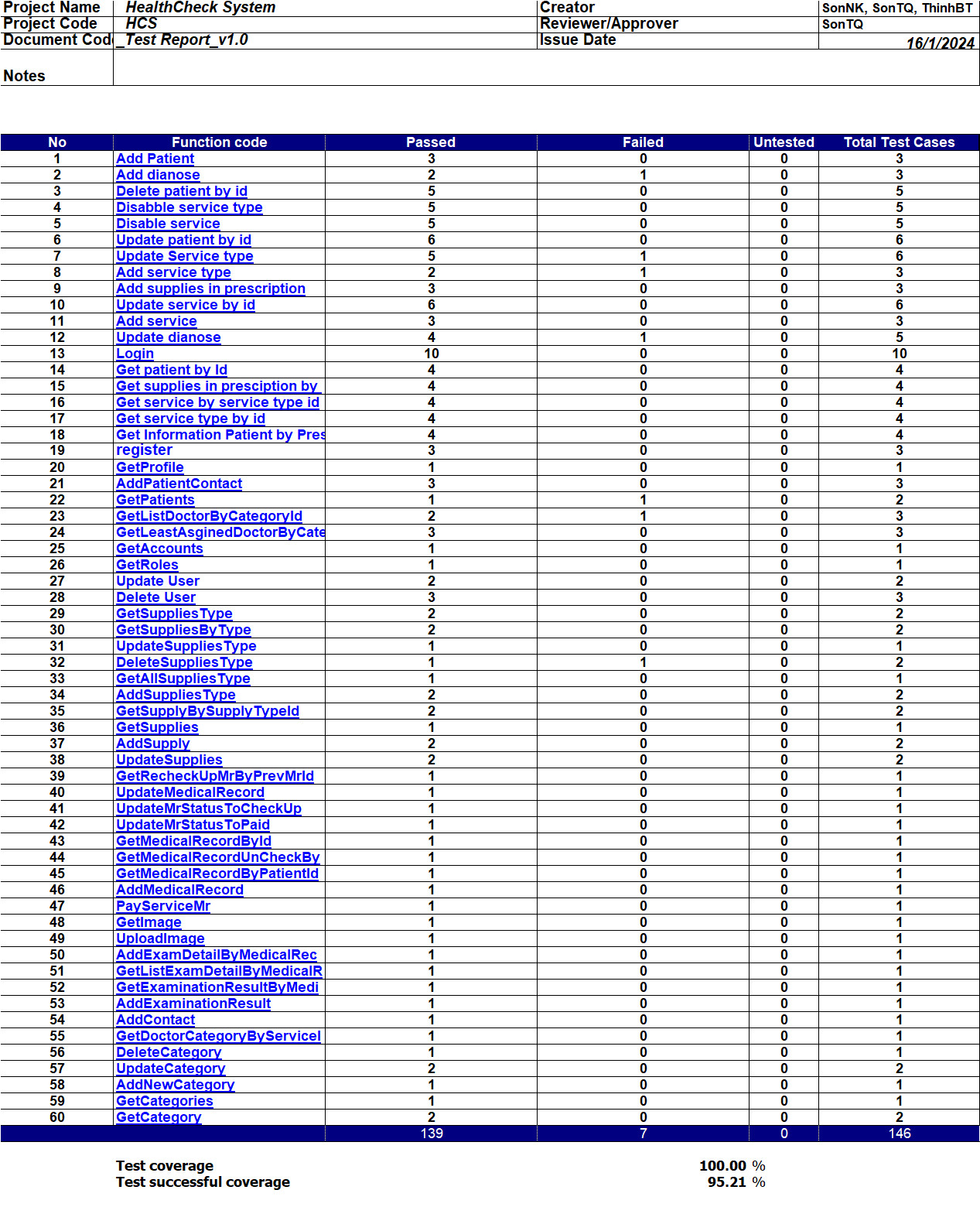
The information about the Unit test report is clearly stated in the document HCS\_Unit Test Report.xlsx.





### 5.2 Integration Test Report

The information about the Integration test report is clearly stated in the document HCS\_Integration Test Report.xlsx.



### 

### 5.3 Functional System Test Report

The information about the System test report is clearly stated in the document HCS\_System Test Report.xlsx.

