# Final Project

# Comprehensive test design of OnSite

TIME: 06/13 2023


Group1:
1953067 宋潇歌
2052225 张勤杭
2051840 梁厚
1950389 季艺

# CONTENT


01

# Introduction

# Background

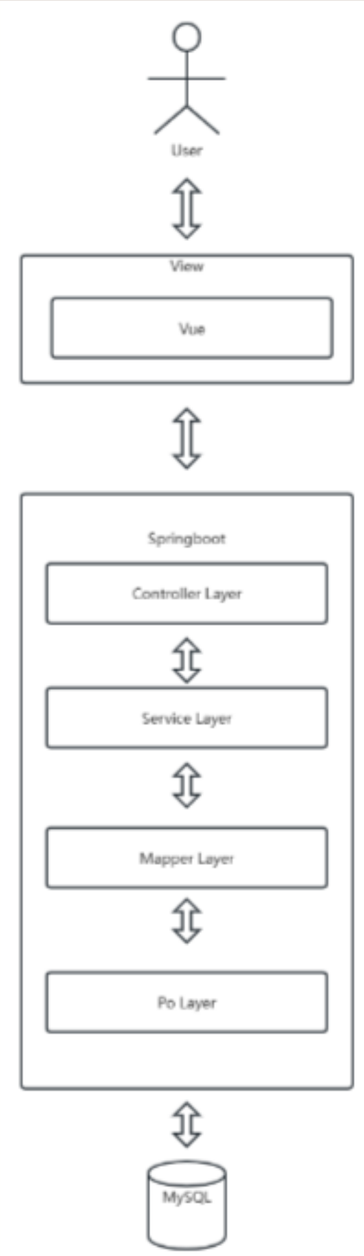

## ONSITE

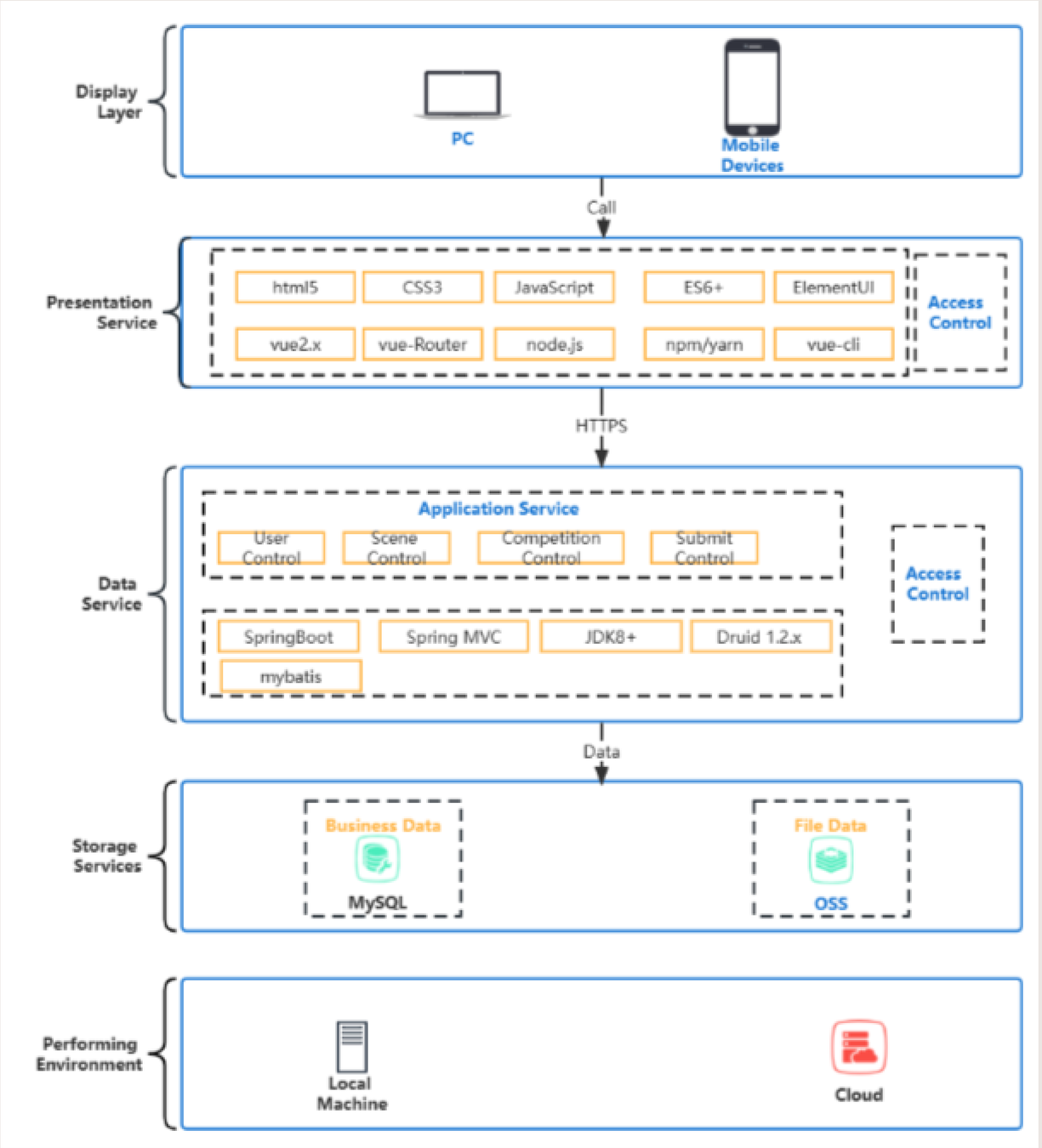
Public Natural Driving Intelligent Vehicle Simulation Test Environment

# General Goals

- Providing a public platform for testing algorithms for highly automated driving vehicles.
- This platform contains tools used for testing, available scenario documents and serves as an official website for an algorithm competition.
- Providing a forum for Q&A.

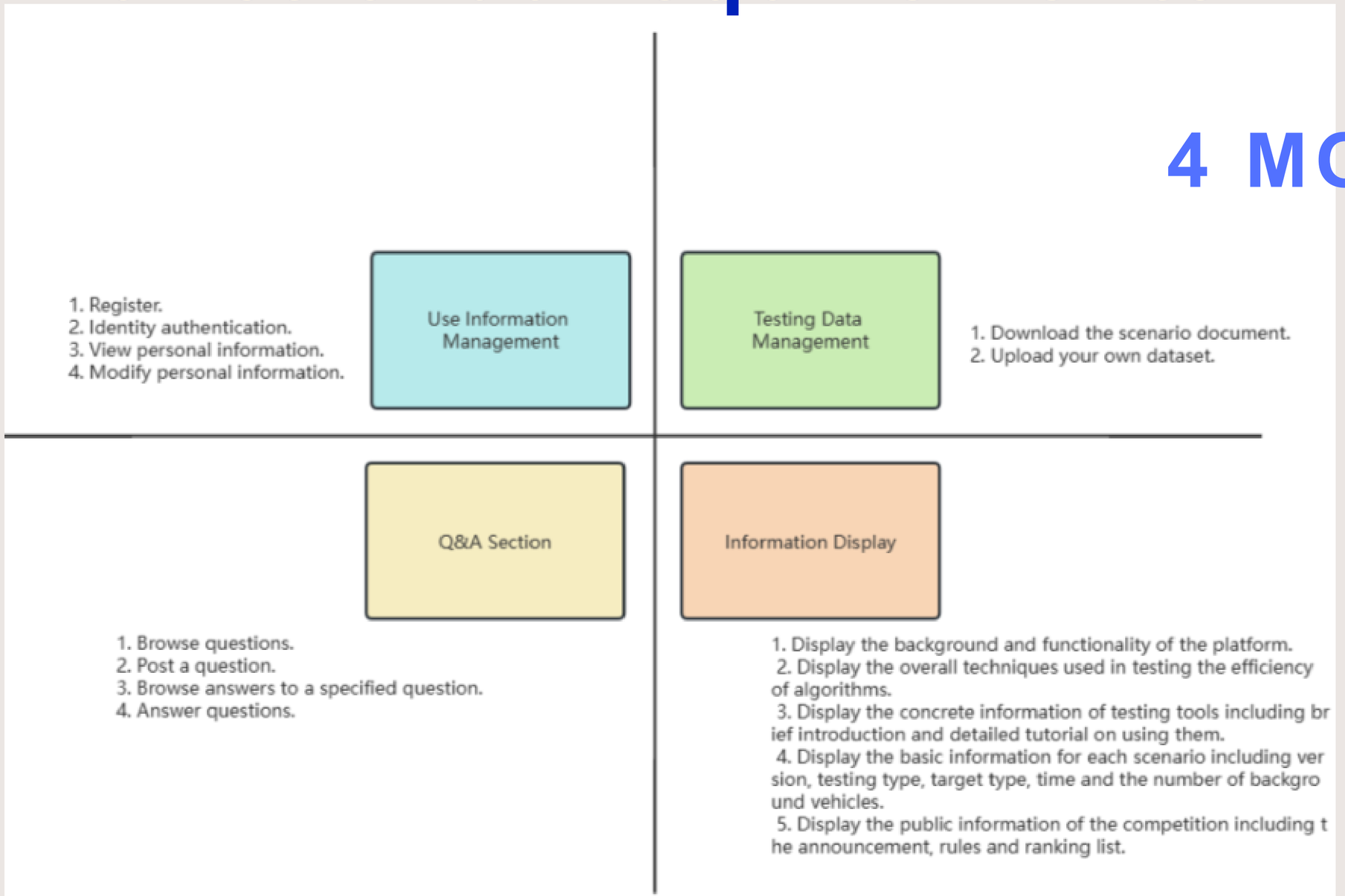
# Software Architecture



# Functional Requirements

## 4 MODULES



# Non-functional Requirements

SAFETY

RELIABILITY

MAINTAINBILITY

USABILITY


02

# Risk Analysis

# 5 MAIN TYPES
# 43 RISKS

**FUNCTIONALITY**
(16 RISKS)

**SECURITY**
(12 RISKS)

**MAINTAINABILITY**
(4 RISKS)

**USABILITY**
(8 RISKS)

**LEGAL AND COMPLIANCE**
(3 RISKS)

| No. | Quality Risk | Tech. Risk | Bus. Risk | Risk Pri. # | Extent of Testing | Tracing |
|---|---|---|---|---|---|---|
| 1 | Functionality | | | | | |
| 1.01 | Browsers unable to access the Web (at all). | 5 | 1 | 5 | Extensive | 2.1 |
| 2.01 | Security Breaches: Unauthorized access or data breaches | 2 | 1 | 2 | Extensive | 2.1.1.b |
| 3.03 | Inconsistent coding standards and practices. | 3 | 3 | 9 | Broad | 2.1 |
| 4.02 | Spam or inappropriate content: Users may post spam or inappropriate content, which can negatively impact the user experience. | 5 | 5 | 25 | Report Bugs | 2.1.3.b |
| 5.01 | The team does not have the right to run an official forum (copyright is not availiable). | 5 | 1 | 5 | Extensive | 2.1 |


03

# Test Plan

# Objectives

- Check whether the functionalities of OnSite are working as expected without any errors or bugs in real business environments
- Check that the external interface of the OnSite such as UI is working as expected and meets the customers' satisfaction; verify the usability of OnSite
- Check whether OnSite meets the functional/non-functional requirements.

# Scope

In the testing:

- **Internet browsers**: Chrome and Edge. Current version and last generally released version of each.
- OS: Windows XP.
- Modules: All the 4 modules.

Not in the testing:

- **Internet browsers**: Older versions or other browsers.
- OS: Other OSs.

# High-level Test Suite Design

- Unit Level (6 test suites)
- System Level (4 test suites)
- Acceptance Level (2 test suites)

| Test Level | Test Target | Test Suite |
|---|---|---|
| Unit Level | Functionality | Test Suite #1-1 |
| | Functionality | Test Suite #1-2 |
| | Functionality | Test Suite #1-3 |
| | Functionality | Test Suite #1-4 |
| | Maintainability | Test Suite #1-5 |
| | Usability | Test Suite #1-6 |
| System Level | Functionality | Test Suite #2-0 |
| | Reliability | Test Suite #2-1 |
| | Maintainability | Test Suite #2-2 |
| | Safety | Test Suite #2-3 |
| | Usability | Test Suite #2-4 |
| Acceptance Lavel | Functionality | Test Suite #3-0 |
| | Usability | Test Suite #3-1 |

# Detailed Test Suite Design

| Test Suite #1-1 | |
|---|---|
| **Objective** | To test the functionality of OnSite's User Information Management module from unit level. |
| **Features to be tested** | 1. user register<br>2. user login<br>3. user logout<br>4. view personal information<br>5. modify personal information |
| **Strategies** | • White-box Testing<br>a. Conduct a static code analysis. Begin by employing a static code analysis tool to detect bugs, vulnerabilities, and code smell concerns within your code, prioritizing these problems based on their severity level. Afterward, produce a comprehensive static code analysis report to document your findings.<br>b. Implement a logic coverage approach. Start by constructing a control flow diagram; subsequently, refer to this diagram while devising sufficient test cases to guarantee that each statement is executed at least once. Should branch statements be present, decision condition coverage (DCC) will be necessary. Additionally, designing basic path coverage is imperative.<br>c. Utilize testing tools to execute the test cases and export the resulting data for further analysis.<br>• Black-box Testing<br>a. Get the initial test cases. Conduct scenario-based evaluations and use state diagrams to assess the events and state alterations occurring while utilizing the function. This allows for the identification of a range of scenario-specific test cases. Utilize a state transition table to compute the coverage of these scenarios and establish comprehensive test cases that guarantee functionality's integrity and accuracy.<br>b. Refine the test cases. Realizing by examining the state diagram and determining if there are multiple input and output possibilities during user interactions. If such possibilities exist, employ black-box testing methodologies like boundary value analysis, equivalence class partitioning, and decision tables to recognize and refine the test cases.<br>c. Eliminate unnecessary test cases. Achieving by synthesizing the test cases and removing redundancies. This process results in a streamlined set of test cases for the Test |

# Detailed Test Suite Design

| | |
|---|---|
| **Tools** | 1. IntelliJ IDEA: run the system.<br>2. SonarQube: Use as the code analysis tool for identifying problematic patterns found in codes<br>3. Edge: view the webpages |
| **Techniques** | • White-box Testing<br>  ◦ Static Code Analysis: Static code examination detects issues like bugs, vulnerabilities, and poor code quality in programming languages.<br>  ◦ Ensuring Full Statement Coverage: This testing method guarantees that all statements within the code are executed at least once.<br>  ◦ Testing Decision Conditions: Decision condition testing confirms that both true and false results are appropriately evaluated in each decision-making process within the code.<br>  ◦ Implementing Program Instrumentation: The core concept of program instrumentation is to strategically place probes within the program without compromising its logic or integrity during testing.<br>• Black-box Testing<br>  ◦ Get the initial black-box test cases. First, tests rooted in specific scenarios are conducted, utilizing state diagrams to evaluate occurrences and shifts in the state throughout the functional usage process. This analysis results in the identification of numerous scenario-based test cases. By employing a state transition table, the coverage of the use case can be ascertained, and a comprehensive test case can be pinpointed. This method ensures the wholeness and accuracy of the function in question.<br>  ◦ Refine the black-box test cases. As per the determined state diagram, it can assess if numerous input and output possibilities exist during the user's operational procedure. Should they be present, black box testing methodologies, like boundary value examination, equivalence category partitioning, and decision tables, will be employed to recognize and enhance the testing scenarios.<br>  ◦ Remove redundant black-box test cases. Finally, the test cases are synthesized and the redundant use cases are removed to obtain a set of test cases in this Test Suite. |

# Detailed Test Suite Design

| | |
|---|---|
| **Reasons** | • Reasons for choosing white-box testing and related techniques.<br>  ◦ Static code analysis: In white-box testing, it is crucial to conduct a manual or tool-assisted static structural analysis.<br>  ◦ Static code analysis and code inspection: A blend of static first and subsequently dynamic methods can be employed in testing. Initially, perform static structure analysis, code examination, and static quality metrics, followed by coverage testing. Utilize the findings from static analysis to direct the process, and further corroborate the results with code inspection and dynamic testing, which enhances the effectiveness of the testing.<br>  ◦ Multiple coverage criteria: The crux of white-box testing lies in coverage testing. Typically, statement coverage criteria can be accomplished through the base path testing approach. For crucial software modules, employ various coverage criteria to assess the code's coverage.<br>  ◦ In distinct testing phases, different focal points should be maintained. During the unit testing phase, the code will be scrutinized, and the primary focus will be on logical coverage.<br>• Reasons for choosing black-box testing and related techniques.<br>  ◦ EP/BV: Begin by examining the equivalence class and selecting appropriate division and boundary values. This approach helps transform infinite tests into finite tests, effectively reducing the workload and enhancing optimization.<br>  ◦ Generate and influence the diagram/decision table: If the dependencies between the software's input and output are highly evident, applying black-box testing can effectively generate and influence the diagram/decision table. |
| **Pass Criteria** | 1. After adding a new user (i.e. user register), the data of the personal information can be found in the database.<br>2. After updating personal information, the data of new personal information is modified in the database successfully.<br>3. All functions tested are correct and complete.<br>4. The test results of the relevant test cases are all passed |

# Test Framework & Tools

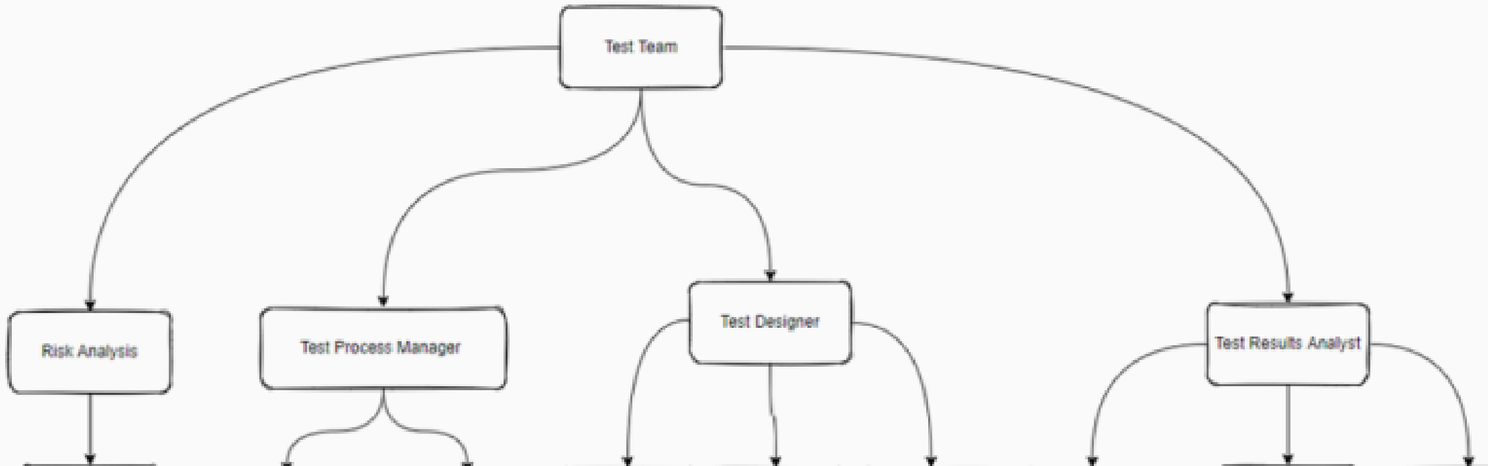
SonarQube is an open-source platform for continuous inspection of code quality to perform automatic reviews with static analysis of code to detect bugs and code smells.

Used to assist to write unit test code and run the code showing the test result.

Postman is a popular collaboration platform for API development. It provides a user-friendly interface for sending HTTP requests, testing APIs, and building API workflows.

# Organizational Chart

| Member | Responsibilities |
| --- | --- |
| Xiaoge Song | • Make the test plan<br>• Collaborate on project requirements and design documents |
| Hou Liang | • Perform risk analysis<br>• Illustrate the system architecture of the project<br>• Test case design |
| Qinhang Zhang | • Test case design<br>• Test tool implementation<br>• Test Results Analysis |



# Cost Estimation

## Task Split

| Task | Sub task |
|---|---|
| Analyze software requirement specification | Investigate the soft requirement specs |
| | Write Risk Analysis Report. |
| Create the Test Specification | Design test scenarios |
| | Create test cases |
| | Review and revise test cases |
| Execute the test cases | Build up the test environment |
| | Execute the test cases |
| | Review test execution results |
| Report the defects | Create the Result Analysis reports |
| | Report the defects |

## Result

| Task | Sub task | Weightage |
|---|---|---|
| Analyze requirement specification | Investigate the soft requirement specs | 3 |
| | Write Risk Analysis Report. | |
| Create the Test Specification | Design test scenarios | 5 |
| | Create test cases | |
| | Review and revise test cases | |
| Execute the test cases | Build up the test environment | 5 |
| | Execute the test cases | |
| | Review test execution results | |
| Report the defects | Create the Result Analysis reports | 1 |
| | Report the defects | |

Weightage: Complex = 5, Medium = 3, Simple = 1

# Cost Estimation

$$FunctionPoints = Weightage * SubTaskNumber$$

| Task | Sub task Number | Weightage | Function Points |
|---|---|---|---|
| Analyze requiremen specification | 2 | 3 | 6 |
| Create the Test Specification | 3 | 5 | 15 |
| Execute the test cases | 3 | 5 | 15 |
| Report the defects | 2 | 1 | 2 |
| Total | | | 38 |

Suppose our project team has estimated defined per Function Points of 3 hours/points. We can estimate the total effort is 38*3 = 114 hours.

Assuming that our team members have an hourly salary of 25RMB/h, the total cost of the test is 114*25=2850RMB.

# Cost Estimation

- **Tools and Software Licenses**

  Testing tools and defect management tools are essential to conducting an efficient testing process. Depending on the type of testing required, the cost of these tools can vary.

  We use JUnit and SonarQube, both of which are free.

- **Hardware and Infrastructure**

  Testing requires servers, desktops, and other devices to conduct various testing types, including performance, load, and compatibility testing. The cost for hardware and infrastructure can vary significantly depending on the organization's needs.

  We use our own laptops as hardware, so there is no extra cost for this.

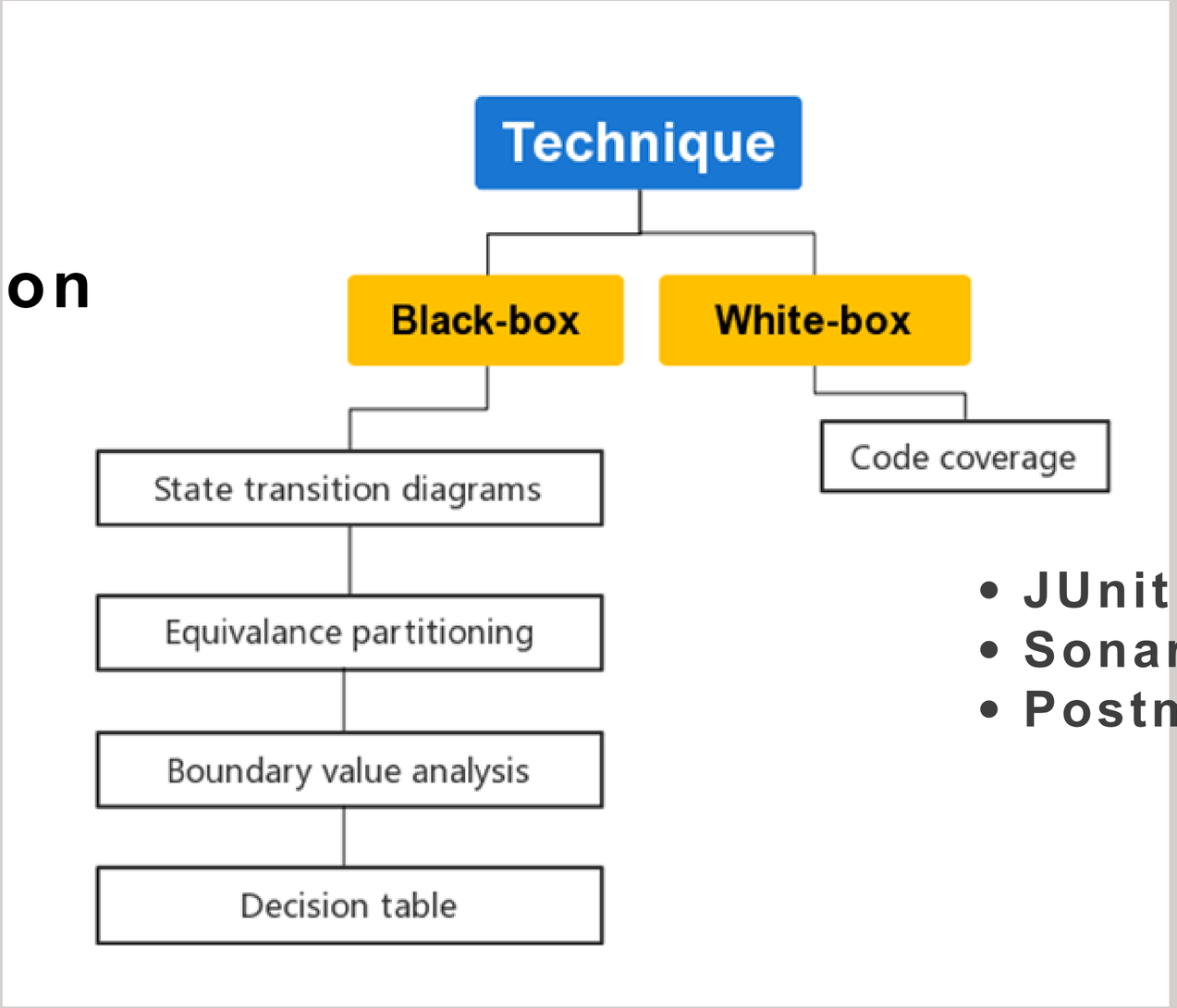

04

# Detailed Tests

# Introduction

Module Selected-

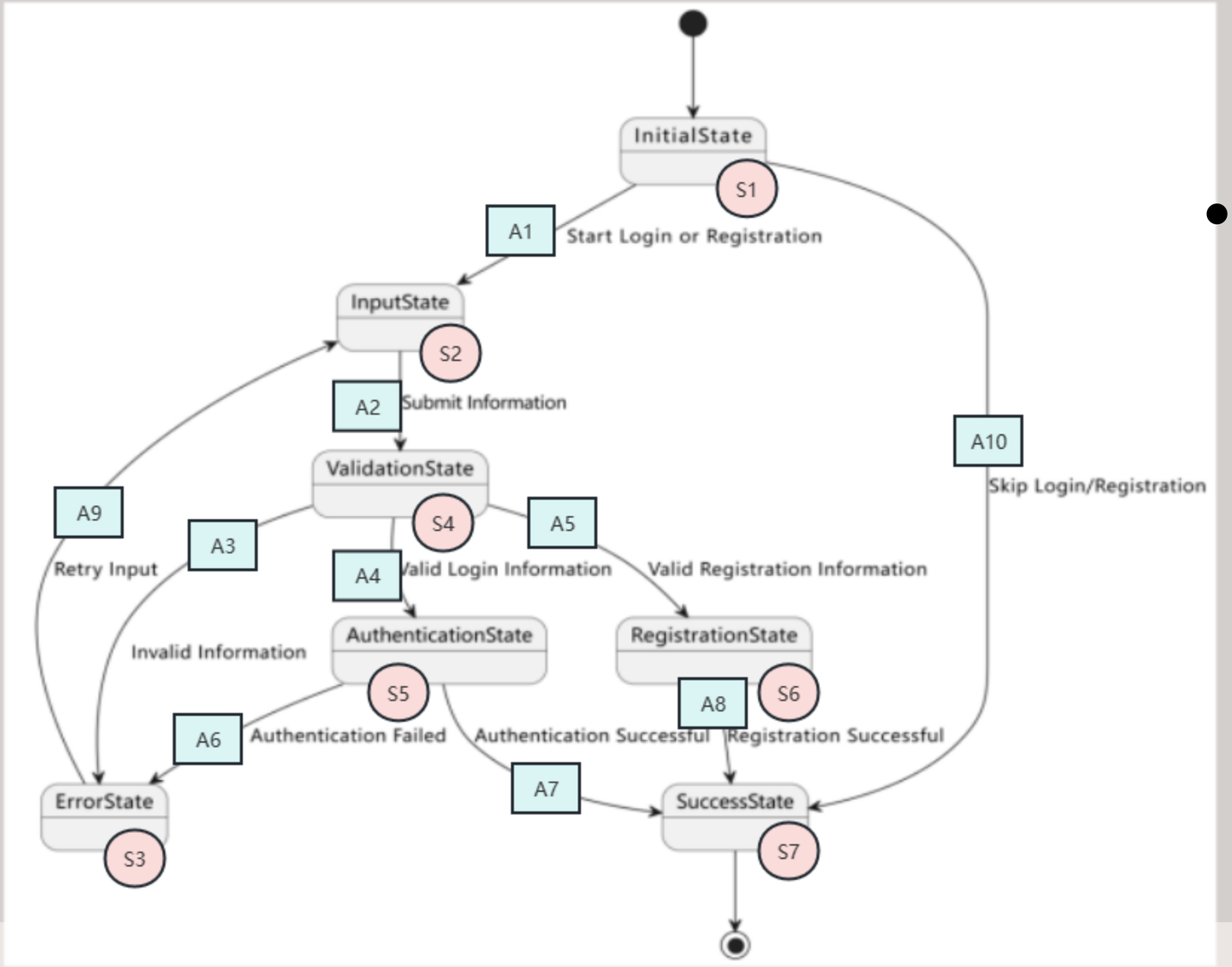
**User Information Management**




- JUnit
- SonarQube
- Postman

# Black-box

## UNIT_1 LOGIN & REGISTER




- **STATE DIAGRAM**

# Black-box

## UNIT_1 LOGIN & REGISTER

## SCENARIO ANALYSIS

- Registration
- Login
- Forgot Password
- Account Lockout
- Security
- User Experience
- Skip Login/Registration

| No. | Test Scenario | Description | Inputs | Expected Outcome |
|---|---|---|---|---|
| #1 | Registration | Verify that a user can successfully register with valid and unique credentials. | Valid and unique username, email, password, etc. | User is successfully registered and redirected to the login page. |
| #2 | | Verify that registration | Usename or email that | Registration fails, and an |

# Black-box

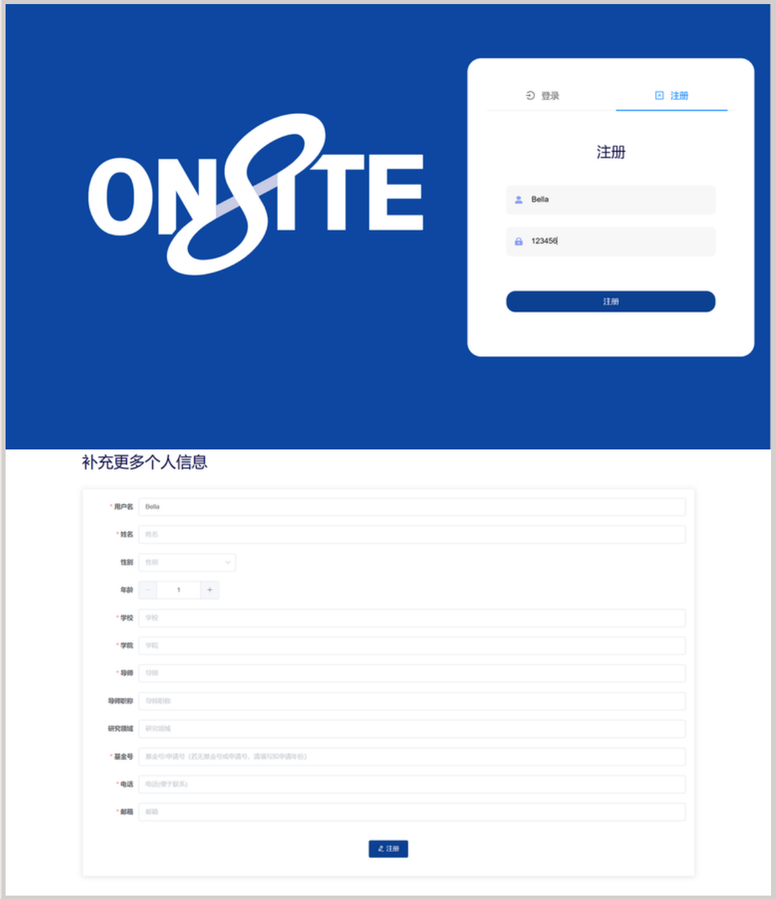
## UNIT_1 LOGIN & REGISTER

| No. | #1 | #2 | #3 | #4 | #5 | #6 | #7 | #8 | #9 | #10 |
|---|---|---|---|---|---|---|---|---|---|---|
| Start State | S1 | S1 | S2 | S3 | S4 | S4 | S4 | S5 | S5 | S6 |
| End State | A1 | A10 | A2 | A9 | A3 | A4 | A5 | A6 | A7 | A8 |
| Action#1 | S2 | S7 | S3 | S2 | S3 | S5 | S6 | S3 | S7 | S7 |
| Contain Scenario | 1,2,3, 4,5,etc . | 27 | 1,2,3, 4,5,etc . | 15,16 | 2,3,4,6 ,7,8,et c. | 5 | 1 | 10 | 5 | 1 |

## 0 – SWITCH COVERAGE

# Black-box

## UNIT_1 LOGIN & REGISTER




| Property | Effective Equivalence Class | No. | Invalid Equivalence Class | No. |
|---|---|---|---|---|
| Username | Strings of length greater than 0 and less than or equal to 20 | 1 | Strings of length greater than 20 | 12 |
| | | | Null | 13 |
| Passowrd | Strings of length greater than 6 and less than or equal to 50 without any Chinese character and blank. | 2 | Null | 14 |
| | | | Strings of length less than 6 | 15 |
| | | | Strings of length greater than 6 and less than or equal to 50 with some Chinese characters or blanks. | 16 |
| | | | Strings of length greater than 50 | 17 |
| Name | Strings of length greater than | 3 | Strings of length greater than 50 | 18 |

## REGISTER TESTING

## EP & BVA

# Black-box

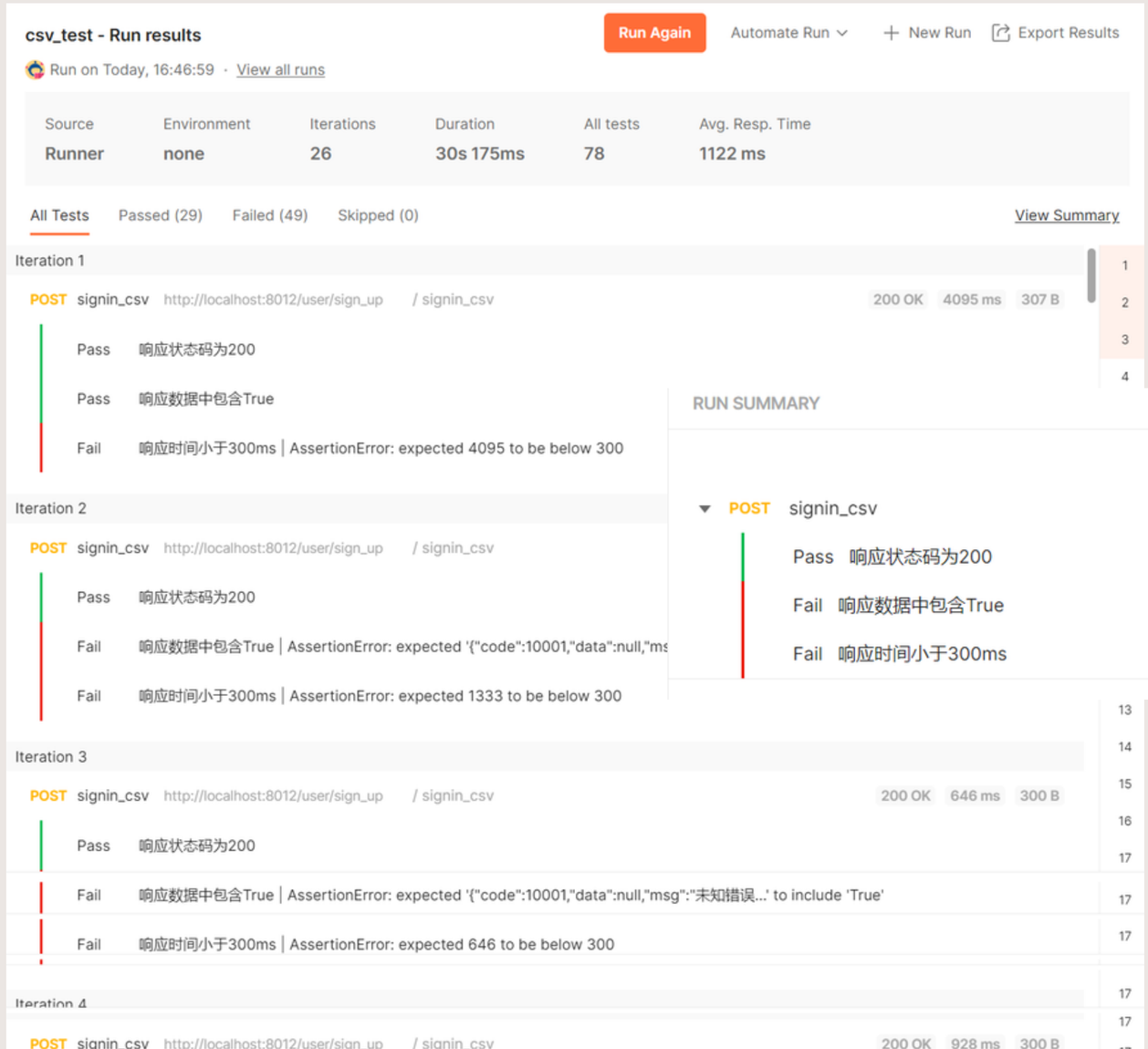
## UNIT_1 REGISTER

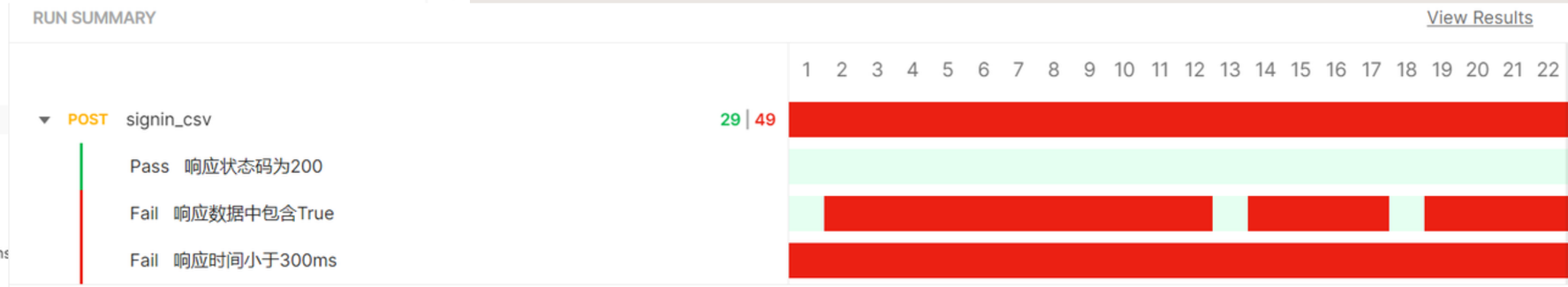
| | No. | EP&BVA | username | password | mobile | email | age | sex | school | departme | supervisornam | supervisor | researchfi | works | name |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| 2 | Test Scenario #1 | | | | | | | | | | | | | | |
| 3 | 1 | 1,2,3,4,5,6,7,8,9,10,11 | Bella | 123456 | 17758689102 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2023 | 王二虎 |
| 4 | | | | | | | | | | | | | | | |
| 5 | Test Scenario #2 | | | | | | | | | | | | | | |
| 6 | 2 | | River | 123456 | 17758689103 | 1306715855@qq.com | 21 | 女 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2023 | 霍林飞 |
| 7 | | | | | | | | | | | | | | | |
| 8 | Test Scenario #3 | | | | | | | | | | | | | | |
| 9 | 3 | BVA 12 | 01234567 | 123456 | 17758689103 | 1306715855@qq.com | 21 | 女 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2023 | 霍林飞 |
| 10 | 4 | BVA 15 | Eve_11 | 12345 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 11 | 5 | BVA 17 | Eve_12 | 123456789012 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 12 | 6 | 16 | Eve_13 | 123456你好 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 13 | 7 | 16 | Eve_14 | 123456 你好 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 14 | 8 | 16 | Eve_15 | 123 456 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 15 | 9 | BVA 18 | Eve_16 | 123456 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 123456789 |
| 16 | 10 | BVA 20 | Eve_17 | 123457 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 12345678 | 901234567 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 17 | 11 | BVA 22 | Eve_18 | 123458 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 12345678 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 18 | 12 | BVA 24 | Eve_19 | 123459 | 15877160554 | 2051849@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 123456789012 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 19 | 13 | BVA 26 | Eve_20 | 123460 | 15877160554 | 2051850@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 12345678 | 自动驾驶 | 2022 | 王二虎 |
| 20 | 14 | BVA 27 | Eve_21 | 123461 | 15877160554 | 2051851@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 12345678 | 2022 | 王二虎 |
| 21 | 15 | BVA 28 | Eve_22 | 123462 | 15877160554 | 2051852@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 12345678 | 王二虎 |
| 22 | 16 | BVA 30 | Eve_23 | 123463 | 123456789012 | 2051853@tongji.edu.cn | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 23 | 17 | BVA 32 | Eve_23 | 123463 | 15877160554 | 123456789012345678901 | 21 | 女 | 同济大学 | 交通学院 | 黄杰 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 24 | | | | | | | | | | | | | | | |
| 25 | Test Scenario #4 | | | | | | | | | | | | | | |
| 26 | 18 | BVA 13 | | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 27 | 19 | 14 | Eve_03 | | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 28 | 20 | BVA 31 | Eve_04 | 123456 | | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 29 | 21 | BVA 33 | Eve_05 | 123456 | 15877160550 | | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 30 | 22 | BVA 21 | Eve_08 | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 31 | 23 | BVA 23 | Eve_09 | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | | 史杨 | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 32 | 24 | BVA 25 | Eve_10 | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | | 副教授 | 自动驾驶 | 2022 | 王二虎 |
| 33 | 25 | BVA 29 | Eve_13 | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | | 王二虎 |
| 34 | 26 | BVA 19 | Eve_14 | 123456 | 15877160550 | 2051001@tongji.edu.cn | 21 | 男 | 同济大学 | 交通学院 | 史杨 | 副教授 | 自动驾驶 | 2022 | |

*detailed test cases*

# Black-box

## UNIT_1 REGISTER

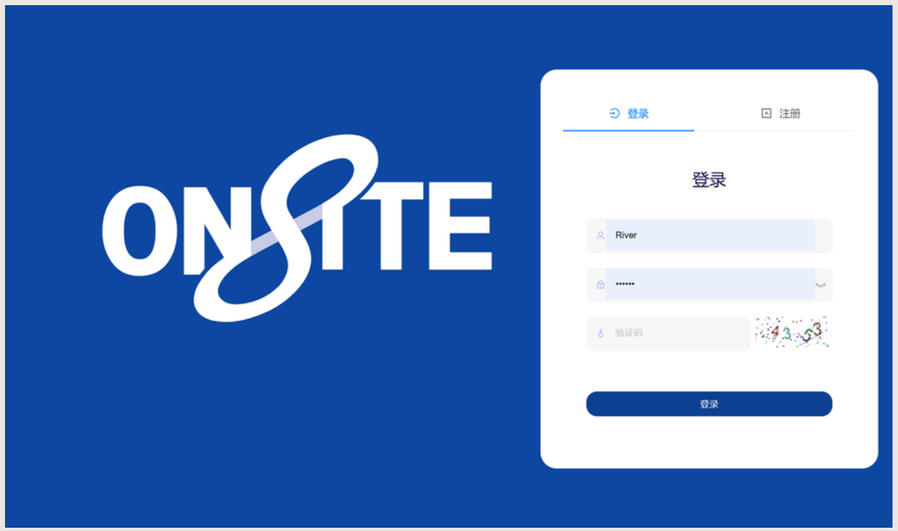



***Running tests & results summary***

# Black-box

## UNIT_1 LOGIN & REGISTER




## LOGIN TESTING

| No. | Scenario No. | username | password | Validcode | Expect Result | Result | Test Pass ? |
|---|---|---|---|---|---|---|---|
| 1 | 5,9 | River | 123456 | 5713 [valid] | PASS | PASS | Y |
| 2 | 8 | - | 123456 | 3616 [valid] | FAIL | FAIL | Y |
| 3 | 6 | @#$% | 123456 | 4521 [valid] | FAIL | FAIL | Y |
| 4 | 7 | River | 123457 | 4353 [valid] | FAIL | FAIL | Y |
| 5 | 8 | River | - | 7143 [valid] | FAIL | FAIL | Y |
| 6 | - | River | 123456 | 6666 [invalid] | FAIL | FAIL | Y |

***detailed test cases & testing results***

# Black-box

# UNIT_1 LOGIN & REGISTER

## OTHER SCENARIOS

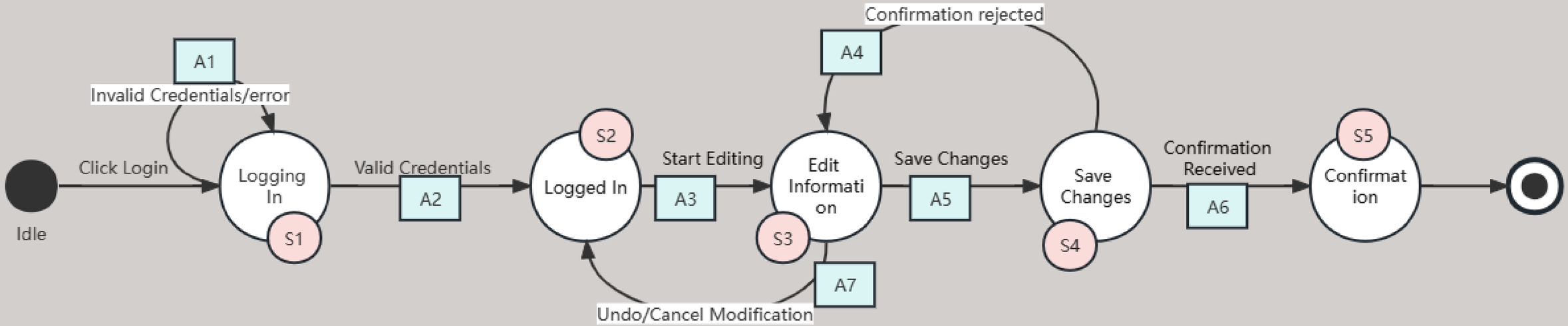
| No. | Scenario No. | Describe | Expect Result | Test Pass ? |
|---|---|---|---|---|
| 1 | 20 | Verify that password fields are masked (hidden) while entering the password. | Password fields are masked (hidden) while entering the password for security reasons. | PASS |
| 2 | 21 | Verify that password fields are case-sensitive. | Password fields are case-sensitive, so "Password" and "password" are considered different. | PASS |
| 3 | 22 | Verify that the system enforces session timeouts to prevent unauthorized access. | The system enforces session timeouts to prevent unauthorized access, automatically logging out the user after a specified period of inactivity. | PASS |
| 4 | 23 | Verify that the system logs failed login attempts for security auditing purposes. | Failed login attempts are logged for security auditing purposes. | FAIL |
| 5 | 24 | Verify that appropriate error messages are displayed for various failure scenarios. | Appropriate error messages are displayed for various failure scenarios, guiding the user on how to resolve the issues. | PASS |
| 6 | 25 | Verify that the user can easily switch between the login and registration forms. | The user can easily switch between the login and registration forms, either through tabs or separate pages. | PASS |

*detailed test cases & testing results*

# Black-box

## UNIT_2 MODIFY PERSONAL INFORMATION

- **STATE DIAGRAM**



# Black-box

# UNIT_2 MODIFY PERSONAL INFORMATION

## SCENARIO ANALYSIS

- Successful Modification
- Invalid Credentials
- Empty Fields
- Restricted Field Modification
- Invalid Input Format
- Maximum Field Length
- Concurrency Handling
- Undo/Cancel Modification

| No. | Test Scenario | Description | Inputs | Expected Outcome |
|---|---|---|---|---|
| #1 | Successful Modification | Users successfully modify their personal information. | • Valid userid<br>• Valid password (token)<br>• Updated personal information (e.g., name, email, phone number) | Personal information is updated successfully, and the changes are reflected in the user's profile. |

# Black-box

## UNIT_2 MODIFY PERSONAL INFORMATION

| No. | #1 | #2 | #3 | #4 | #5 | #6 | #7 |
|---|---|---|---|---|---|---|---|
| Start State | S1 | S1 | S2 | S3 | S3 | S4 | S4 |
| End State | S1 | S2 | S3 | S2 | S4 | S3 | S5 |
| Action#1 | A1 | A2 | A3 | A7 | A5 | A4 | A6 |
| Contain Scenario | 2 | 1 | 1 | 8 | 1 | 3,4,5,6 | 1 |

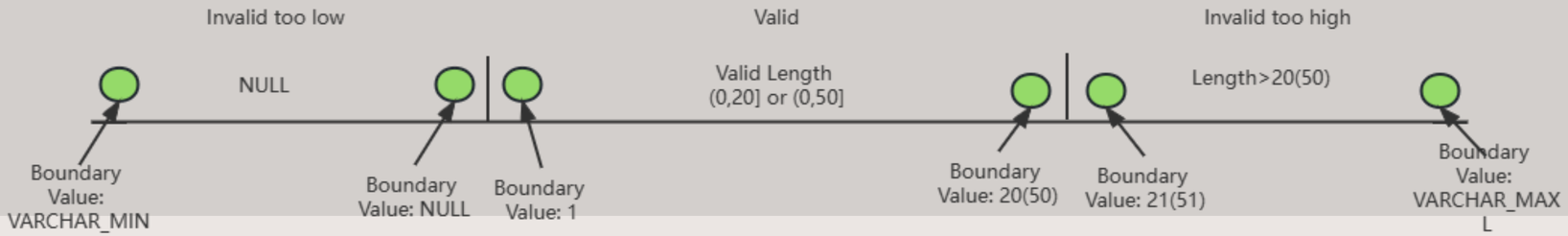
## 0 – SWITCH COVERAGE

# Black-box

# UNIT_2 MODIFY PERSONAL INFORMATION

## *EP & BVA*

| Property | Effective Equivalence Class | No. | Invalid Equivalence Class | No. |
|---|---|---|---|---|
| Username | Strings of length greater than 0 and less than or equal to 20 | 1 | Strings of length greater than 20 | 11 |
| | | | Null | 12 |
| Name | Strings of length greater than 0 and less than or equal to 50 | 2 | Strings of length greater than 50 | 13 |
| | | | Null | 14 |
| School | Strings of length greater than 0 and less than or equal to 50 | 3 | Strings of length greater than 50 | 15 |
| | | | Null | 16 |
| Faculty | Strings of length greater than 0 and less than or equal to 50 | 4 | Strings of length greater than 50 | 17 |
| | | | Null | 18 |



# Black-box

# UNIT_2 MODIFY PERSONAL INFORMATION

## *Decision Table*

| Decision Table | | TC1 | TC2 | TC3 | TC4 | TC5 | TC6 | TC7 | TC8 | TC9 | TC10 | TC11 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Conditions | Username valid? | No | - | - | - | - | - | - | - | - | - | Yes |
| | Name valid? | - | No | - | - | - | - | - | - | - | - | Yes |
| | School valid? | - | - | No | - | - | - | - | - | - | - | Yes |
| | Faculty valid? | - | - | - | No | - | - | - | - | - | - | Yes |
| | Tutor valid? | - | - | - | - | No | - | - | - | - | - | Yes |
| | Title of Supervisor valid? | - | - | - | - | - | No | - | - | - | - | Yes |
| | Field of Research valid? | - | - | - | - | - | - | No | - | - | - | Yes |
| | Fund Number valid? | - | - | - | - | - | - | - | No | - | - | Yes |
| | Phone valid? | - | - | - | - | - | - | - | - | No | - | Yes |
| | Email valid? | - | - | - | - | - | - | - | - | - | No | Yes |
| Actions | Reject operation | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | Yes | - |
| | Save changes | - | - | - | - | - | - | - | - | - | - | Yes |

# Black-box

## UNIT_2 MODIFY PERSONAL INFORMATION

| No. | EP&BVA | userid | username | mobile | email | age | sex | school | department | supervisorname | supervisorrank | researchfield | works | name | token |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| Test Scenario #1 | | | | | | | | | | | | | | | |
| 1.01 | 1,2,3,4,5,6,7,8,9,10 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| Test Scenario #2 | | | | | | | | | | | | | | | |
| 2.01 | | 20220207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 2.02 | | | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 2.03 | | 0 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 2.04 | | @#$%^&*() | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| Test Scenario #3 | | | | | | | | | | | | | | | |
| 3.01 | 12,14,16,18,20,24,26,28 | 20230207023944 | | | | | | | | | | | | | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.02 | 12 | 20230207023944 | | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.03 | 26 | 20230207023944 | River | | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.04 | 28 | 20230207023944 | River | 17786605331 | | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.05 | 16 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.06 | 18 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.07 | 20 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.08 | | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.09 | | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.10 | 24 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 3.11 | 14 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| Test Scenario #4 | | | | | | | | | | | | | | | |
| 4.01 | | 20230207023945 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| Test Scenario #5 | | | | | | | | | | | | | | | |
| 5.01 | BVA | 20230207023944 | 0 | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.02 | | 20230207023944 | River | 123 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.03 | BVA | 20230207023944 | River | 0 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.04 | | 20230207023944 | River | 17786605331 | 12345 | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.05 | BVA | 20230207023944 | River | 17786605331 | @ | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.06 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 1 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.07 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 1 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.08 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 1 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.09 | BVA | 20230207023944 | River | 0 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 1 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.10 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 1 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.11 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 1 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 5.12 | BVA | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 1 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| Test Scenario #6 | | | | | | | | | | | | | | | |
| 6.01 | 11 | 20230207023944 | 123456789012345678901 | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.02 | 25 | 20230207023944 | River | 123456789012345678901 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.03 | 27 | 20230207023944 | River | 17786605331 | 12345678901234567890 | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.04 | 15 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 12345678901234 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.05 | 17 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 12345678901 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.06 | 19 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 12345678901234 | 教授 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.07 | 21 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 12345678901234 | 自动驾驶 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.08 | 22 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 12345678901234 | 2023 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.09 | 23 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 12345678901234 | 赵佳 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |
| 6.10 | 13 | 20230207023944 | River | 17786605331 | 1306715855@qq.com | 20 | 男 | 同济大学 | 交通学院 | 涂辉招 | 教授 | 自动驾驶 | 2023 | 12345678 | eyJ0eXAiOiJKV1QiLCJhbGciOiJIUz |

mof_msg_tc

## *detailed test cases*

# Black-box

## UNIT_2 MODIFY PERSONAL INFORMATION

csv_test - Run results

Run Again | Automate Run | New Run | Export Results

Run on Today, 16:32:29 · View all runs

| Source | Environment | Iterations | Duration | All tests | Avg. Resp. Time |
|---|---|---|---|---|---|
| Runner | none | 40 | 3s 767ms | 120 | 50 ms |

All Tests | Passed (110) | Failed (10) | Skipped (0) | View Summary

Iteration 1

POST modify_msg_2 http://localhost:8012/user/modify_msg / modify_msg_2 200 OK 62 ms 307 B

Pass 响应状态码为200

Pass 响应数据中包含True

Pass 响应时间小于300ms

Iteration 2

POST modify_msg_2 http://localhost:8012/user/modify_msg / modify_msg_2

Pass 响应状态码为200

Fail 响应数据中包含True | AssertionError: expected '{"code":10001,"data":null,"msg":"未

Pass 响应时间小于300ms

Iteration 3

POST modify_msg_2 http://localhost:8012/user/modify_msg / modify_msg_2 200 OK 29 ms 300 B

Pass 响应状态码为200

Fail 响应数据中包含True | AssertionError: expected '{"code":10001,"data":null,"msg":"未知错误..." to include 'True'

Pass 响应时间小于300ms

Iteration 4

POST modify_msg_2 http://localhost:8012/user/modify_msg / modify_msg_2 200 OK 30 ms 300 B

Pass 响应状态码为200

RUN SUMMARY View Results

1 2 3 4 5 6 7 8 9 10 11 12 13 14 15 16 17 18 19 20 21 22

POST modify_msg_2 110 | 10

Pass 响应状态码为200

Fail 响应数据中包含True

Pass 响应时间小于300ms

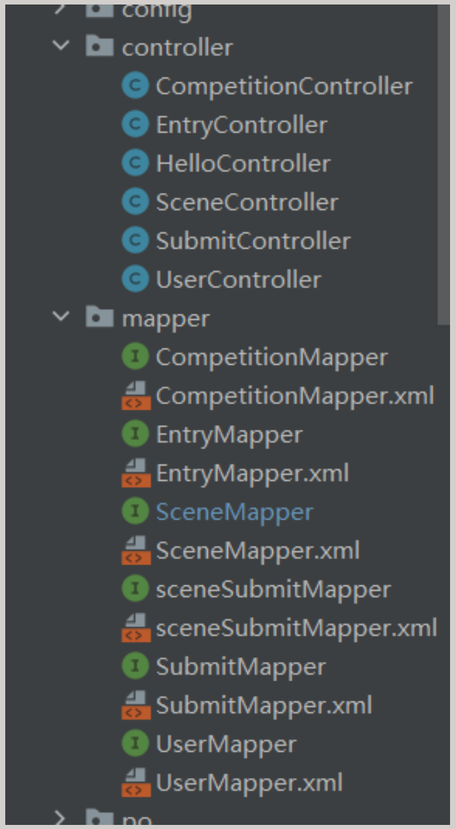
## *Running tests & results summary*

# Black-box

## UNIT_2 MODIFY PERSONAL INFORMATION

| No. | Scenario No. | Test Case | Expect Result | PASS? |
|---|---|---|---|---|
| 41 | #7 Concurrency Handling | Two users simultaneously attempt to modify personal information for the same account. | The system should handle the concurrency scenario gracefully, ensuring that only one user's changes are applied while providing appropriate feedback to the other user. | Y |
| 42 | #8 Undo/Cancel Modification | User initiates the modification process but decides to cancel or undo the changes. | The system should discard the changes made by the user and revert back to the original personal information. | Y |

# White-box




White box testing focuses on examining the code structure and internal logic to ensure comprehensive coverage of statements and branches. The goal is to achieve 100% statement coverage and conditional coverage, ensuring that all code paths and decision points are executed and validated during testing. By analyzing the code structure and conducting white box testing, we can assess the quality and integrity of the test suite and identify any potential gaps or areas that require additional testing.

# White-box

**The object of this white box test is a Java class named UserController.java, which contains nine methods as follows:**

| Function Name | Description | Included in Test Cases? | Branches within Function? |
|---|---|---|---|
| loginValidate() | Validates user login credentials and performs authentication. | Yes | No |
| signUp() | Registers a new user and adds them to the system. | Yes | No |
| checkMsg() | Checks for new messages for the logged-in user. | Yes | No |
| modifyMsg() | Modifies user messages, such as marking them as read. | Yes | No |
| modifyPassword() | Allows users to change their account password. | Yes | No |
| validateUsername() | Validates the uniqueness and format of a username during registration. | Yes | No |
| validateMobile() | Validates the uniqueness and format of a mobile number during registration. | Yes | No |
| validateEmail() | Validates the uniqueness and format of an email address during registration. | Yes | No |

# White-box

## Database connection test

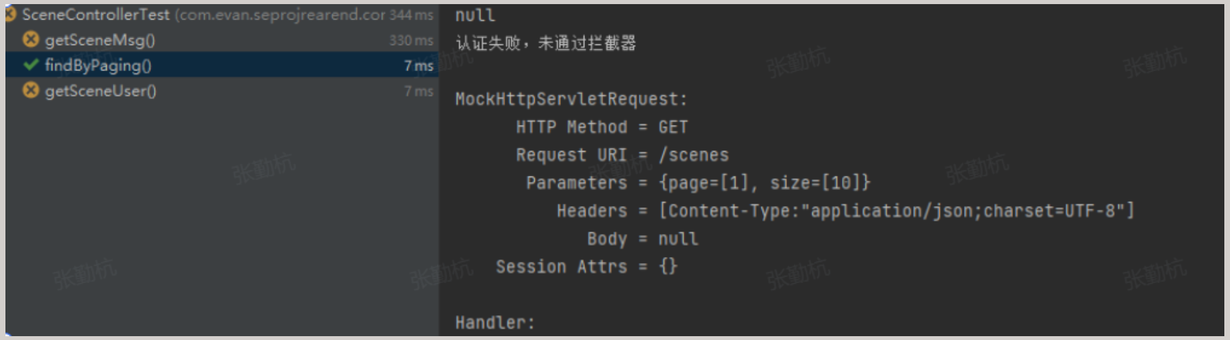
# White-box

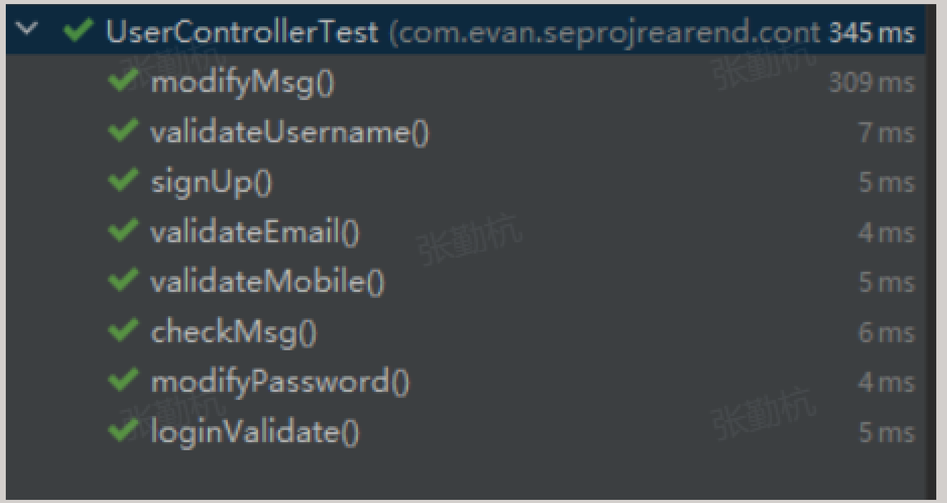
## Code example:

The purpose of the MockMvc in the provided code is to simulate HTTP requests and test the functionality of the UserController class without actually making real network connections or accessing the database. It allows for isolated testing of the controller's endpoints by providing a controlled environment where the controller can be tested independently of other components, such as the web server or the database. By mocking the HTTP requests and responses, it enables the execution of specific test scenarios and assertions on the expected behavior of the controller methods, ensuring that they handle the requests correctly and return the expected responses.

```C++
import org.junit.jupiter.api.Test;
import org.springframework.beans.factory.annotation.Autowired;
import org.springframework.boot.test.autoconfigure.web.servlet.AutoConfigureMockMvc;
import org.springframework.boot.test.context.SpringBootTest;
import org.springframework.http.MediaType;
import org.springframework.test.web.servlet.MockMvc;
import org.springframework.test.web.servlet.request.MockMvcRequestBuilders;
import org.springframework.test.web.servlet.result.MockMvcResultMatchers;

import static org.springframework.test.web.servlet.result.MockMvcResultHandlers.print;

@SpringBootTest
@AutoConfigureMockMvc
class SceneControllerTest {

    @Autowired
    private MockMvc mockMvc;

    @Test
    void findByPaging() throws Exception {
        mockMvc.perform(MockMvcRequestBuilders.get("/scenes")
                        .param("page", "1")
                        .param("size", "10")
                        .contentType(MediaType.APPLICATION_JSON))
                .andExpect(MockMvcResultMatchers.status().isOk())
                .andDo(print());
    }

    @Test
    void getSceneMsg() throws Exception {
        mockMvc.perform(MockMvcRequestBuilders.get("/scenes/1"))
                .andExpect(MockMvcResultMatchers.status().isOk())
                .andExpect(MockMvcResultMatchers.jsonPath("$.id").value(1))
                .andExpect(MockMvcResultMatchers.jsonPath("$.name").value("Test Scene"))
                .andDo(print());
    }

    @Test
    void getSceneUser() throws Exception {
        mockMvc.perform(MockMvcRequestBuilders.get("/scenes/1/users"))
                .andExpect(MockMvcResultMatchers.status().isOk())
                .andExpect(MockMvcResultMatchers.jsonPath("$[0].id").value(1))
                .andExpect(MockMvcResultMatchers.jsonPath("$[0].name").value("User1"))
                .andExpect(MockMvcResultMatchers.jsonPath("$[1].id").value(2))
                .andExpect(MockMvcResultMatchers.jsonPath("$[1].name").value("User2"))
                .andDo(print());
    }
}
```

# White-box Result




05

# Test Result Analysis

# Test Result Analysis

## 01.Register

- In almost all 26 test cases in register, the response time suspended 300ms, it may cause lower usability.
- There are no mechanisms to check whether email or phone are in the correct format. It may cause some problems when fake or fraudulent registrations occur (risk 1.05) or inadequate validation of user input during the registration process(risk 1.06), which means that in this feature, potential risks are not handled properly.

## 02.Login

- It will be a serious problem that there is no mechanism for "Forgot Password" and "Account Lockout" handling. Users can not find their password in an effective way which leads to low usability. On the other hand, no mechanism for "Account Lockout" means the system can not deal with brute-force attacks (malicious actors can attempt to guess or crack user passwords), so that the user's personal information can be exposed.

## 03.Mod-Inf

- The modifying personal information page can not handle wrong input (empty field, invalid format, etc) properly.
- It is not reasonable to allow users to change their username, which is defined when registering. In other words, strict areas, such as username or email can not be modified easily.

# Test Result Analysis

**During the white box testing process for the User management module, we have reached the following conclusions:**

During the white box testing process for the User management module, we have reached the following conclusions:

1. Functional tests: All functional test cases for the UserController.java file have passed. This indicates that the functions and logic in the file are functioning correctly under different input conditions. The requests are processed accurately, and the expected results are returned.

2. Boundary tests: By testing the boundary cases of input, we have confirmed that the UserController.java file handles boundary conditions correctly for different ranges of input values. This demonstrates that the file is robust and can handle all possible input values effectively.

3. Exception handling: The test results indicate that the UserController.java file adequately captures and handles various exception situations. Whether incorrect parameters are provided or other contingencies occur, the file handles them in a reasonable manner to avoid application crashes or unexpected errors.

4. Code coverage: In the white box test, our test cases have achieved 100% coverage of the code in the UserController.java file. This means that we have tested every statement and branch in the file, ensuring code integrity and reliability.

5. However, one area for improvement is the usage of int data for the annotation's result instead of following HTTP standard codes. This can pose challenges in test design and may lead to inconsistencies. It would be beneficial to align the result annotations with the HTTP status codes to enhance clarity and conformity with industry standards.