**Kontroller**

Iteration Test Plan

Version 1.0

Revision History

| **Date** | **Version** | **Description** | **Author** |
| --- | --- | --- | --- |
| 29/07/2025 | 1.0 | Initial version of the test plan for functional testing on selected use cases (Sign in, Sign up, Manage Profile). Includes features to be tested, testing strategy, schedule, and resources. | Nguyen Minh Thuan, Nguyen Thanh Dat |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

Table of Contents

1. Introduction 4

1.1 Purpose 4

2. Target Test Items 4

3. Environmental Needs 4

3.1 Hardware Requirements 4

3.2 Software in the Test Environment 4

3.3 Productivity and Support Tools 4

4. Responsibilities, Staffing, and Training Needs 4

4.1 People and Roles 4

**Iteration Test Plan**

# Introduction

This document outlines the test plan for functional testing of three selected use cases in the Kontroller application: Sign in, Sign up, and Manage Profile. It defines the testing objectives, scope, approach, and resources required to ensure that these features work as intended.

The main objectives of this document are:

* To describe the overall strategy and process for functional testing.
* To provide a structured plan for designing, executing, and evaluating test cases for selected features.
* To serve as a communication tool between team members and stakeholders regarding the testing process.
* To provide clear roles, responsibilities, and deliverables for all QA participants.
* To minimize testing risks and ensure effective defect identification and resolution before deployment.

This document is intended to be used by:

* **Quality Assurance Team:** For executing test cases, recording results, and logging defects.
* **Developers:** For understanding test focus areas, fixing identified defects, and collaborating during retests.
* **Project Manager/Stakeholders:** For reviewing progress, ensuring deliverables are on schedule, and tracking software quality.
* **End-users (Indirect):** Benefiting from improved application stability, functionality, and usability.

Key features of this document include:

* A list of features to be tested.
* Test strategy and testing environment.
* Detailed test cases and execution status.
* A summary of test results and identified defects (if any).

# Target Test Items

The following software items and features will be tested in this iteration:

* 1. Sign in Functionality
* Input validation for email/username and password.
* Authentication logic and error handling.
* Redirection and user session management upon successful login.
* Usability (form clarity, error messages, accessibility).
  1. Sign up Functionality
* Input validation (username, email format, password strength).
* Duplicate account detection and error responses.
* Email confirmation (if applicable).
* Usability and user guidance during registration.
  1. Password Reset Functionality
* User submits email to request password reset.
* System verifies that the email is associated with an existing account.
* If valid, system generates a new random password and sends it to the user's email.
* System updates the password in the database.
* User can use the new password to log in.

Additional aspects covered during testing:

* Functional correctness of the features.
* Usability, including layout, form clarity, and feedback to users.
* Basic security, such as unauthorized access prevention to profile management.

1. **Cross-browser behavior** (e.g., Chrome, Firefox).

To ensure consistent functionality and user experience across different platforms, the selected features will be tested on multiple modern web browsers. The following browsers are targeted for compatibility testing:

* **Google Chrome** (latest stable version)
* **CocCoc** (latest stable version)
* (Optional: Add more if applicable, e.g., Microsoft Edge, Safari)

Key aspects to verify across browsers:

* Correct rendering of UI components (forms, buttons, text alignment).
* Functional consistency (form submissions, redirects, profile update behavior).
* Input validation and error message display.
* Responsive behavior (if the UI supports responsive design).

# Environmental Needs

This section outlines the non-human resources and environment configurations required to execute the test plan effectively.

## Hardware Requirements

Development and testing machines with:

* **CPU:** Intel i5 or equivalent (multi-core recommended).
* **Memory:** Minimum 4 GB RAM; recommended 8 GB+ for smoother execution.
* **Storage:** SSD ≥ 256 GB to support database, server, and testing tools.
* **Network:** High-speed internet (minimum 10 Mbps) to support client-server communication.

## Software in the Test Environment

The following base software elements are required in the test environment for this *Test Plan*.

| **Software Name** | **Purpose** | **Version** | **Type and Other Notes** |
| --- | --- | --- | --- |
| Windows 10 | Operating system for backend server | 64-bit | Main OS for backend |
| Google Chrome | Web application testing | Latest stable | Main browser used for functional testing |
| Coc Coc | Web application testing | Latest stable | Secondary browser for cross-browser testing |
| SQL Server | Application database | Any stable version | Local test data used for consistent results |
| Kontroller WinForms Tool | Server management utility | Internal tool | Built with C# WinForms; start/stop/monitor NetCoreServer |
| Google Sheets / Excel | Test case tracking and reporting | Web-based or desktop | Used to manage test cases and summarize results |
| Postman | API testing (future scope) | Latest stable | Kiểm tra API backend (login, profile). |

## Productivity and Support Tools

The following tools will be employed to support the test process for this *Test Plan*.

| **Tool Category or Type** | **Tool Brand Name** | **Vendor or In-house** | **Version** |
| --- | --- | --- | --- |
| Test Case Management | Google Sheets | Google | Web-based (latest) |
| Test Result Recording | Google Sheets | Google | Web-based (latest) |
| Documentation | Google Docs | Google | Web-based |
| Server Management Tool | Server WinForms | In-house | Internal build |
| Defect Tracking | Trello | Atlassian | Cloud-based (latest) |
| Collaboration Tool | Slack | Slack | Cloud-based (latest) |
| Version Control | Github/GitLab | Open-source | Cloud-based (latest) |

# Responsibilities, Staffing, and Training Needs

## People and Roles

This table shows the staffing assumptions for the test effort.

| **Human Resources** | | |
| --- | --- | --- |
| **Role** | **Minimum Resources Recommended**  **(number of full-time roles allocated)** | **Specific Responsibilities or Comments** |
| Test Manager | 1 | Provides overall test planning and coordination. Also handles  Responsibilities include:   * Scheduling test execution * Assigning tasks to testers * Reporting progress and risks to team lead. |
| Tester | 3 | Executes manual test cases for selected use-cases (Sign-in, Sign-up, Manage Profile).  Responsibilities include:   * Logs results in Google Sheets * Reports bugs or inconsistencies. |
| Test System Administrator | 1 | Manages and monitors test environment  Responsibilities include:   * Starts/stops server using Server WinForms Tool * Ensures the browser environment is clean and consistent before tests. |