***TEST SPECIFICATION***

Ben Siao (2848022), Alex Orban (2821580), Joe Norton (2837372), Allen Jiang (2846054)

4/27/2025

**1.0 Introduction**

This section provides an overview of the entire test document. This document describes both the test plan and the test procedure.

**1.1 Goals and objectives**

Overall goals and objectives of the test process are described.

The goal of the testing process was to guarantee that all the core functionalities of the ChoreHero web application were fully operational and that the software met the requirements we created during the development phase. The test process was important as it helped us verify the usability, correctness, and stability of the system as well as identify any remaining bugs or defects. The objectives were to validate that users can successfully register a new account, log in to created accounts, manage tasks, keep a score count of completed tasks, and interact with the system as intended without any major errors.

**1.2 Statement of scope**

A description of the scope of software testing is developed. Functionality/features/behavior to be tested is noted. In addition, any functionality/features/behavior that is not to be tested is also noted.

Testing focused on verifying that the main functionality of the ChoreHero system worked as intended. This included the following functionalities:

1. User registration for new accounts
2. User login for existing accounts
3. Task creation and deletion
4. Viewing tasks
5. Viewing names of existing accounts in the system
6. Task completion
7. Updating user score counts based on task completion

Features not tested include:

1. Security vulnerabilities beyond basic form validation
2. Stress testing under high concurrent user loads (since it is a local deployment)
3. Large-scale scalability of the system for production-level usage

**2.0 Test Plan**

This section describes the overall testing strategy and the project management issues that are required to properly execute effective tests.

**2.1 Software to be tested**

The software to be tested is identified by name. Exclusions are noted explicitly.

The software tested was the ChoreHero web application. This includes:

1. Frontend pages developed with HTML, CSS, and JavaScript to provide a user interface
2. MySQL database storing user, task, and reward information
3. PHP backend scripts responsible for processing user actions and database communication

All main functionalities were included in testing. No major feature areas were excluded, except for areas that were specifically noted in the scope such as security testing beyond form validation, stress testing under high user loads, and large-scale scalability of the system as a whole.

**2.3 Testing tools and environment**

A description of the test environment, including tools, simulators, specialized hardware, test files, and other resources is presented here.

Testing was done on a local development environment using the following:

1. MAMP as the local server setup (Apache, MySQL, PHP)
2. phpMyAdmin for database creation, management, and verification
3. Various web browsers such as Google Chrome, Mozilla Firefox, and Microsoft Edge for user interface and functionality testing
4. Manual functional testing methods were also used for verifying the behavior of the system

No specialized testing frameworks or automation tools were required given the size and scope of the project.

**2.4 Test schedule**

A detailed schedule for testing is described.

|  |  |  |  |
| --- | --- | --- | --- |
| **Phase** | **Timeframe** | **Activity** | **Description** |
| 1 | April 6 to April 10 | Unit Testing | After each individual feature was implemented (user registration, login, task management, etc.), initial unit tests were run to verify that each feature worked correctly in isolation. |
| 2 | April 11 to April 17 | Integration Testing | After verifying individual components, integration tests were run to make sure that all components such as login, database access, and task management interacted correctly and without issue. |
| 3 | April 18 to April 24 | System Testing | Full-system level testing was done using the local server environment. This included workflow tests that covered login, task creation, task completion, and score updates. This testing was run across multiple browsers (Chrome, Firefox, and Edge) to guarantee consistency. |
| 4 | April 25 to April 27 | Defect Retesting and Final Verification | After fixing any identified issues in previous testing phases (such as invalid input handling or other identified issues), all features were retested. Final checks helped us make sure that all functional requirements were met before project submission. |

**3.0 Test Cases**

This section enumerates a complete list of test cases for the software. A template for test cases is as follows.

|  |  |
| --- | --- |
| ID |  |
| Test Input |  |
| Expected Output |  |
| Description |  |

Below shows a table containing each of our test cases including all sections required (ID, Test Input, Expected Output, and Description) from the test case template shown above:

|  |  |  |  |
| --- | --- | --- | --- |
| **ID** | **Test Input** | **Expected Output** | **Description** |
| TC\_01 | Valid registration information (unique email, password) | New user account is created, and user is logged in | Tests user registration functionality |
| TC\_02 | Invalid registration information (existing email, weak password) | Error message is displayed; account is not created | Tests validation and error handling for registration |
| TC\_03 | Valid login credentials (existing account) | User successfully logs in and accesses the dashboard | Tests successful user login |
| TC\_04 | Invalid login credentials (wrong email, wrong password) | Error message is displayed; login fails | Tests login error handling for invalid credentials |
| TC\_05 | New task input (name, optional description, due date) | Task appears in the user's task list | Tests adding a new task to the system |
| TC\_06 | Selecting a task to delete | Selected task is removed from the user's task list | Tests deletion of an existing task |
| TC\_07 | Viewing tasks after login | All assigned tasks for the user are displayed correctly | Tests viewing of user tasks |
| TC\_08 | Viewing list of user accounts | Names of all registered users are displayed correctly | Tests retrieval and display of user account names |
| TC\_09 | Submitting a form with missing or empty fields | Error message is displayed; task is not created | Tests form validation for task creation |
| TC\_10 | Completing a task | Task status changes to "Complete" and user score increases | Tests marking a task as completed and updating score |
| TC\_11 | Completing multiple tasks | User’s score increments appropriately for each completed task | Tests cumulative score updating after multiple task completions |