Waste To Taste

Product Testing Plan

Version 1.0

April 1, 2024

Delbert Li
Lucas Prifti
Saadman Choudhury

Revision History

Date	Description	Author	Comments
4/1/2024	Version 1.0	- Delbert Li - Lucas Prifti - Saadman Choudhury	First Draft of Test document submitted

Document Approval

The following Product Design Specification has been accepted and approved by the following:

Signature	Printed Name	Title	Date
DL	Delbert Li	● Team Lead ● Presentation Lead ● Assistant Documentation Lead ● UI/Frontend Lead	4/10
LP	Lucas Prifti	● Database/Backend Lead ● Presentation Lead ● Note-Taker ● QA Lead	4/10
SC	Saadman Choudhury	● Documentation Lead ● UI/Frontend Lead, Assistant Database/Backend Lead	4/10
	Dr. Seyed Ziae Mousavi Mojab	● Professor	

Table of Content

1. Introduction

1.1 Purpose:

The purpose of this Testing Plan Document is to outline the approach that the testers will take when testing the Waste To Taste' web application. This document will specify testing strategies that the testers will utilize to test the system, the scope that will specify what will and will not be tested. It will also specify the test schedule, environment, entry/exit criteria, pass/fail criteria, risk analysis, and roles/responsibilities. This will lead to an efficient, well guided testing procedure that lays out everything for the testers to follow closely. This plan is designed to ensure that the functional and non-functional requirements are verified for compliance, performance, security, and user preference. The document aims to establish the framework for identifying and resolving issues, consequently enhancing the web application's quality and reliability.

1.2 Objectives:

The primary objectives of this testing plan consist of:

- Ensuring the application functions properly, with all features and functional requirements working in accordance with their specifications.
- Confirming the application's compliance with relevant data protection methods and data security.
- Validating the application's performance under normal and peak load conditions.
- Verifying the application's usability, ensuring a positive user experience.
- Identifying and addressing potential issues before the final deadline.
- Validate the functional/non-functional requirements of the system.
- Verify the system performance.
- Ensure security and reliability.

1.3 Scope:

The scope of this testing plan covers all aspects of the Waste To Taste web application, including:

- User Account Management: Including registration, login, profile management, and password recovery functionalities.
- Recipe Handling: Encompassing creation, editing, deletion, and sharing of recipes.
- Search/Tag Functionality: Testing the efficiency and accuracy of the search and tag functionalities.

- User Interface and User Experience: Assessing the responsiveness of the application across different devices and browsers.
- Security: Ensuring the application's resistance to common web vulnerabilities and data protection compliance.

These are the aspects of the Waste To Taste web application that our scope will not cover.

- Recipe Description Editing
- Recipe Allergen editing
- Admin Panel
- External API's (nodemailer, bcrypt)

1.4 Intended Audience:

This document's intended audience are the System Testers, more specifically for the Waste To Taste Development team as we will conduct all of the testing. This document will outline the testing strategies, environment, scope, schedule, and roles/responsibilities of all of the members performing and managing these tests.

1.5 Document Overview:

This testing document is structured to guide the testing strategy for the Waste To Taste web application, this document is divided into sections detailing the test strategy, objectives, scope, deliverables, and schedule, alongside specific criteria for test execution, defect management, and risk mitigation. It concludes with appendices, and reference documents, to ensure a shared understanding of terms and testing strategies. This structure is intended to provide a clear, comprehensive path for the testing process, from initial planning through to execution and final evaluation.

2. Test Objectives

- **Verify Comprehensive Functionality:**

Action: Systematically test each feature outlined in the project, including account creation, login processes, recipe input, modification, deletion, and user-generated content sharing. Implement a blend of automated scripts for routine actions and manual testing for nuanced user interactions to cover all use cases.
Measurement: Achieve a 100% pass rate on all defined test cases, ensuring every feature works as intended without any bugs.

- **Assess Performance Under Varied Conditions:**

Action: Utilize tools like Apache JMeter to apply stress and load testing, simulating both typical and peak usage scenarios. Focus on critical performance indicators such as server response times and page load speeds under these varied conditions.
Measurement: The application maintains a response time of under 3 seconds for all actions under normal load conditions, and system scalability to support at least a 50% increase in concurrent user load without critical performance degradation.

- **Uphold Security and Data Protection Standards:**

Action: Engage in comprehensive security testing,and code reviews to identify potential security risks.
Measurement: Zero tolerance policy for critical vulnerabilities, with all identified security concerns addressed before launch.

- **Ensure Reliability and Error Handling:**

Action: Implement a suite of regression tests and monitor system stability and error handling mechanisms continuously through automated testing frameworks. This includes testing for system behavior under failure conditions, such as network outages or server errors.
Measurement: The system demonstrates 99% uptime, excluding scheduled maintenance, with all critical errors resolved within a 24-hour window and non-critical issues within a week.

- **Optimize Cross-Platform User Experience:**

Action: Conduct exhaustive cross-platform and cross-browser testing to ensure the application's UI/UX is consistent and responsive across different devices and browsers. This includes using tools like BrowserStack for testing and collecting user feedback through beta testing.
Measurement: The application is fully functional and visually consistent on the latest two versions of major browsers (Chrome, Firefox, Safari, Edge) and on both iOS and Android platforms, with user satisfaction ratings exceeding 85%.

- **Validate Data Integrity and Operational Accuracy:**

Action: Perform detailed testing on all forms of data input, storage, retrieval, and deletion within the application to ensure accuracy and integrity. Use both manual inspection and automated database tests to validate data consistency and correctness.

Measurement: No data corruption or loss instances, with 100% accuracy in data display and manipulation functionalities as per user actions.

3. Test Strategy

3.2 Testing Methodology: White-Box Testing

We have decided that White-Box testing will be our principal methodology as the main testers of Waste To Taste will be us, the developers, so we have a very familiar understanding of all of the internal code and structure.

3.2 Testing Levels

- ### Functional Testing

Unit Testing: Utilizing White-Box testing, we look into the application's fundamental components, such as text validation mechanisms and database connection protocols. While exhaustive testing of every unit isn't required for this overview, critical units essential to the application's core functionality will be comprehensively tested. White-Box testing allows for an in-depth analysis of the code, ensuring these elemental functions operate correctly and efficiently.

Integration Testing: Utilizing White-Box testing, this phase assesses the seamless integration of various functionalities within the application. Focusing on essential user paths, such as the processes of registration, login, email verification, and password recovery, this stage verifies that different components interact as expected. The integration testing primarily focuses on the end-to-end workflow, from user input through to the expected outcome, ensuring a qualitative user experience.

System Testing: This is a crucial phase where we evaluate the majority of the web applications in an integrated environment. This stage is designed to verify the seamless operation of the application across all functionalities—such as adding, deleting, searching, and filtering. It simulates real-world usage scenarios to ensure that every component of the application works in harmony. System testing acts as a series of comprehensive integration tests, covering the full spectrum of the application’s functionalities. This phase primarily utilizes White-box testing methods, focusing on how the application responds to various inputs, ensuring that users receive the expected outcomes without the need for testers to delve into the codebase.

Acceptance Testing: This phase marks the culmination of the testing process. It is where the web application undergoes the final verification to ensure it meets all the predefined requirements and is ready for deployment. Acceptance testing is key for assessing the application from the end-user’s and clients' perspectives, focusing on functionality, performance, and compliance.

- Non-Functional Testing

System Testing: This critical phase extends beyond functional requirements, examining the application's performance, security, and usability under conditions that mimic real-world use. Key performance tests include evaluating the application's behavior under significant load, for instance, managing 100 concurrent users, to validate its stability and responsiveness. This comprehensive testing ensures the application is robust, secure, and user-friendly, ready for wide-scale deployment.

Performance Testing: This phase is focused on evaluating how the application behaves under normal and peak load conditions. This testing aims to identify any performance factors that could potentially negatively affect the user experience. We will simulate scenarios such as handling 100 concurrent users, which allows us to measure the application's response times and overall stability under stress. The goal is to ensure that the application remains responsive and stable, providing a seamless experience for users even during periods of high demand.

Security Testing: The security testing phase is equally critical, aimed at uncovering vulnerabilities that could compromise the application's integrity and the safety of user data. Through a combination of automated tools and manual testing techniques, we test the application for common vulnerabilities such as security misconfigurations. This comprehensive security audit ensures that the application adheres to the best practices in web security, safeguarding against potential threats and attacks.

- Special Tools and Techniques:

- **Selenium**: A popular and effective choice for automating tests, especially during system and acceptance testing, Selenium simulates user interactions across diverse environments, ensuring the application's interface is intuitive and responsive.
- **Jest**: The Jest tool is pivotal for White-Box unit testing, allowing for granular testing of individual code units. Its use ensures each function within the application is performing as intended.

- **Apache JMeter**: Employed during the system testing phase for performance testing, JMeter simulates high traffic conditions, assessing the application's handling of heavy user loads to ensure scalability and reliability.

4. Test Scope

The test scope for the Waste To Taste web application is a detailed testing assessment of both functional and non-functional requirements, as outlined in the Software Requirements Specification (SRS) document. This section outlines the specific areas and features targeted for testing to ensure the application's integrity, performance, and user satisfaction.

- Functional Requirements Testing:

User Registration: Testing will verify the process of creating new user accounts, focusing on input validation, data integrity, and the uniqueness of user credentials. This includes ensuring the correct handling and encryption of sensitive information, such as passwords, and confirming that a verification email is correctly generated and dispatched to the user.

User Login: The login functionality will be examined to authenticate user credentials accurately and grant appropriate access to the personalized dashboard. Error handling for incorrect credentials and the effectiveness of the password reset process will also be assessed.

Email Verification: This testing area ensures the email verification process operates as intended, validating users' email addresses through a secure link and activating their accounts successfully upon completion.

Forgot Password: The focus will be on the reliability and security of the password reset process, ensuring users receive clear instructions for resetting their passwords and handling non-existent email addresses appropriately.

Profile Management: Testing will cover the ability of users to update their profile information within specified constraints and time frames. Validation checks for input criteria and feedback mechanisms for update failures will be evaluated.

Personal Dashboard: The personalized dashboard's functionality will be tested to confirm that it accurately reflects user activities, saved recipes, and notifications, with proper error handling for missing data.

Recipe Saving & Recipe Management: The testing will ensure users can effectively manage their recipes—adding, editing, deleting, and saving them with accurate feedback and validation.

- **Non-Functional Requirements Testing:**

Performance: Performance testing will focus on response and processing times under normal and peak loads, ensuring the application meets specified benchmarks for speed and efficiency.

Reliability: Assessments will be made to guarantee system stability, low error rates, and high data integrity, confirming the application's reliability for all users.

Security: Security tests will ensure robust protection mechanisms are in place, including mandatory email verification for new users to maintain system and data integrity.

By outlining these specific testing targets, we aim to comprehensively evaluate the Waste To Taste web application across all outlined functional and non-functional requirements, ensuring readiness for an optimal user experience.

5. Test Deliverables

5.1 Unit Testing

Approach

The Unit Testing phase is crucial in order to examine the smallest, independent components of the Waste To Taste web application. Since we are using a 3-Tier Layered MERN stack for our project, which primarily adopts Javascript as its main programming language, Jest serves to be the most optional tool to use for this phase of testing. White-box testing techniques will be employed for the Unit Testing, since us developers are also the testing team. This will allow us to utilize our in-depth knowledge of the web app's internal structure and conduct precise tests on the internal logic of the web app.

Pass/Fail Criteria

A unit test shall be considered passed if it performs exactly up to the expectations that have been set for it to pass, in accordance with the passing specification of its pertaining test case. Each unit test must execute without any defects or errors and produce the expected outcomes for its predefined inputs for it to be considered as passed. Vice versa, if there is any deviation, non-expected outcomes or errors from the expected results of a test case, it shall be marked as failed

Entry/Exit Criteria

- **ENTRY**

Unit testing is to be initiated once the initial development phase has been completed. We must make sure that each unit to be tested has been fully implemented and completed. By initializing the phase of unit testing before any other testing phases, we are able to receive immediate feedback on the functionality of the segments of the code.

- **EXIT**

The unit testing phase shall be concluded once all the specified unit tests have been comprehensively executed, and have passed. If any defects, or issues are detected within the test, they must be urgently addressed and corrected until the test passes, ensuring the reliability of all the units within the web application.

Suspension/Resumption Criteria

- **SUSPENSION**

If there are any critical defects uncovered during the unit testing phase, which may impede any further testing, or which may deter the unit's functionality, the unit testing phase shall be suspended temporarily. This suspension of unit testing will allow for the defect to be resolved timely, before any other testing is conducted.

- **RESUMPTION**

The unit testing process is to be resumed once the potential critical issues holding any further progress have been identified, resolved and passed. The test environment and units must be verified and ready for the continuation of any further testing.

Risks Analysis

One of the main risks related to the unit testing phase is encountering major bugs within the logic of the specific component. In some cases, a defect like this may trigger a domino effect, negatively impacting the further stages of testing and the web app as a whole. In order to stay clear of this type of risk, we will conduct continuous integration and testing. By doing so we will be able to detect any potential bugs or defects in the early stages of testing, maintaining the timelines and ensuring thorough testing.

Tested Items

- TC-1 (Successful Registration)
- TC-3 (Registering with Invalid Email Format): Testing validation logic for email format ensures proper feedback when users input an invalid email format during registration.
- TC-4 (Registering using Weak Password): Ensuring password complexity rules are enforced to maintain account security.

- TC-5 (Registering with Blank Fields): Verifying that all required fields are validated for input presence to prevent incomplete registrations.
- TC-6 (Successful Login)
- TC-8 (Session Persistence)
- TC-9 (Successful Profile Update)
- TC-10 (Updating Profile Using Blank Fields): Checks for proper handling of empty inputs during profile updates, ensuring that essential user information cannot be left blank.
- TC-11 (Successful Creation)
- TC-14 (Successful Edit)
- TC-15 (Successful Deletion)
- TC-22 (View All Recipes: Share Recipe)
- TC-23 (Explore New Creations: Save Recipe)
- TC-24 (Successful Logout)
- TC-26 (Request Password Reset Link)
- TC-27 (Reset Password With Valid Link)
- TC-29 (View Dashboard)
- TC-30 (Advanced Recipe Search)
- TC-31 (Create Food List)

5.2 Integration Testing

Approach:

Integration testing for the Waste To Taste web application will focus on the interaction between the applications distinct modules an the backend services, focusing on the MERN Stack's connected functionality. The testing team will initialize the database and have predefined datasets to mimic realistic user interactions and scenarios. This will facilitate testing of integrated functionality such as user account creation/management, recipe creation/management, and food list operations in a controlled environment.

Pass/Fail Criteria:

- **PASS:**

A test is considered passed if it executes according to the test plan, and all expected outcomes are observed without deviation. This includes the successful interaction between modules, accurate data retrieval/storage, and maintained session states across different application components

- **FAIL:**

A test is failed if there are deviations within the test from the expected outcomes, including errors in data handling, incorrect application behavior, or a failure in session management.

Entry/Exit Criteria:

- **ENTRY:**

Integration testing will begin upon the successful completion of functional testing, unit testing, and verification that all individual components are functioning as intended in isolation.

- **EXIT:**

Testing will conclude when all integration test cases have been executed with all tests meeting the pass criteria, ensuring that Waste To Taste's application components operate well together.

Suspension/Resumption Criteria:

- **SUSPENSION:**

Testing will be suspended if there is a critical issue encountered that prevents further testing.

- **RESUMPTION:**

Testing will resume once the issue causing suspension has been resolved(through code modifications or testing adjustments during the testing process).

Risks Analysis:

Some risks that come from integration testing with MongoDB and the MERN stack application, include data schema inconsistencies due to MongoDB's schema-less nature, performance issues from improper indexing, and causing consistency issues. MongoDB requires thorough testing to ensure accurate data mapping, and there needs to be a backup and or recovery process to prevent data loss.

Tested Items

Our integration testing execution path will look like this:

1. User registration and authentication flow
2. User account management
3. Recipe Creation, ViewAllRecipe viewing, Recipe Details, Edit Recipe, Delete Recipe
4. Recipe Sharing, ExploreNewCreation Saving
5. Food List creation, Food List viewing, Food List editing.
6. Integration of user interaction with backend services, verify data consistency, and session management.

5.3 System Testing

Approach

System testing for the Waste To Taste web app will be focused on testing the web application as a whole to ensure it meets the outlined functional and non-functional requirements. The approach will include various types of testing such as functional, performance, and security testing. This will aim to simulate real-world usage conditions to identify any potential issues that could affect the user experience or the system.

Pass/Fail Criteria

- **PASS:**

A test will be considered passed if it executes to the test case specification and the expected outcomes without any deviations.

- **FAIL:**

Any test that does not meet the expected outcome, has any unexpected behavior, or causes errors will be a failed test.

Entry/Exit Criteria

- **ENTRY:**

System testing will begin once functional testing, unit testing, and integration testing has been completed successfully ensuring that all components are working whether in isolation or when integrated.

- **EXIT:**

System testing will conclude when all planned tests have been executed and passed, all critical bugs have been fixed and retested, and the application meets the performance, reliability, and security benchmarks set forth in the non-functional requirements.

Suspension/Resumption Criteria

- **SUSPENSION:**

Testing will be suspended if a critical bug is encountered that prevents further testing or that could significantly affect user experience. We may test non-related modules if needed.

- **RESUMPTION:**

Testing will resume once the critical issues have been addressed and fixes are verified and tested.

Risks Analysis

For the System Testing phase, potential risks may include integration failures that disrupt user workflows and security vulnerabilities that could compromise data integrity. In order to mitigate such risks our system testing phase shall begin after the successful conclusion of unit, functional, and integration testing, ensuring a thoroughly tested application framework. This strategy ensures addressing pivotal concerns, enhancing the application's overall robustness and delivering an efficient user experience overall.

Tested Items

- ITC-1 (Create New Recipe Integration): Tests the integration between the front-end form inputs and the backend API, ensuring that recipe data is correctly handled and stored.

- ITC-2 (Recipe Editing, Deletion Attempt, and Cancellation): Verifies the integration of user actions (edit, attempt to delete, cancel deletion) with the database operations, ensuring the flow operates seamlessly.

- ITC-3 (Recipe Saving & Sharing): Ensures the functionality of sharing a recipe by one user and saving it by another operates correctly, including the integration of front-end actions with backend database updates.

5.4 Performance Testing

Approach:

Performance Testing for the Waste To Taste web application aims to assess the system's stability, responsiveness, and scalability under various conditions of load. This testing phase will simulate real-world usage scenarios, including multiple users accessing the application simultaneously, to identify any potential bottlenecks or performance issues. Apache JMeter will be utilized to generate virtual users and mimic user interactions with the application, focusing on critical functionalities such as the recipe exploration.

Pass/Fail Criteria:

- **PASS:** A performance test is considered passed if the application meets or exceeds the performance benchmarks set for response times, throughput, and error rates under simulated load conditions.
- **FAIL:** A test fails if the application exhibits performance degradation beyond acceptable thresholds, including response times longer than 2 seconds, throughput significantly lower than expected, or error rates exceeding 1% under the simulated load for up to 100 concurrent users.

Entry/Exit Criteria:

- **ENTRY:** Performance testing commences after the completion of unit, integration, and system testing phases, ensuring that the application is functionally stable and ready for load testing.

- **EXIT:** The performance testing phase concludes once all planned tests have been executed, and the application has met the performance benchmarks across all critical functionalities.

Suspension/Resumption Criteria:

- **SUSPENSION:** Testing will be suspended if critical performance issues are identified that severely impact the user experience or system stability, requiring immediate attention from the development team.
- **RESUMPTION:** Testing resumes once the identified performance issues have been addressed and the application modifications have been verified to meet performance expectations.

Risks Analysis:

Potential risks during the Performance Testing phase include discovering significant performance bottlenecks that could delay the project timeline and scalability issues within the application architecture. Mitigation strategies include early load testing, scalable cloud resource allocation, and architecture optimization.

Tested Items:

- **Load Testing for Recipe Exploration Performance (PTC-1):** This test evaluates the performance of the "View and Explore Recipes" feature under varying degrees of user load to ensure the application remains responsive and stable.

5.5 Security Testing

Approach:

Security Testing for the Waste To Taste web application focuses on verifying the web application's mechanisms to prevent unauthorized access and ensure the integrity of the user authentication process. This phase specifically targets testing the application against common security threats related to user authentication and password management, as outlined in the specified test cases.

Pass/Fail Criteria:

- **PASS:** A security test case is considered passed if the application successfully prevents unauthorized access, displays appropriate error messages for invalid inputs, and handles password reset operations securely.
- **FAIL:** A test case fails if the application allows unauthorized access, fails to display appropriate error messages, or mishandles password reset operations, thereby exposing the application to potential security risks.

Entry/Exit Criteria:

- **ENTRY:** Security testing begins after the completion of functional, integration, system, and performance testing phases, ensuring the application is fully functional and performance-optimized.
- **EXIT:** This phase concludes once all security test cases have been executed, with all identified vulnerabilities addressed and mitigated, ensuring the application meets established security standards.

Suspension/Resumption Criteria:

- **SUSPENSION:** Testing may be suspended if critical security vulnerabilities that pose immediate risks are identified, necessitating urgent reparations.
- **RESUMPTION:** Testing resumes once critical vulnerabilities have been effectively mitigated or resolved, and the application's security measures have been re-verified.

Risks Analysis:

Potential risks during the Security Testing phase include the discovery of vulnerabilities that could lead to unauthorized access or data exposure. Mitigation strategies involve rigorous testing of authentication mechanisms and ensuring secure password reset functionalities.

Tested Items:

- **SecurityTC-1: Logging in using Invalid Email** - Tests the application’s ability to prevent login attempts with an email address that is not registered in the system, ensuring unauthorized users cannot gain access.
- **SecurityTC-2: Logging in using Incorrect Password** - Verifies the system's response to incorrect password attempts, ensuring that users cannot access the system without the correct credentials.

- **SecurityTC-3: Reset Password With Expired Link -** Assesses the application's handling of expired password reset links, ensuring that expired or invalid links cannot be used to compromise account security.

6. Test Schedule

Test Case document submission

- 04/01/2024 at 5pm
- 4/09/2024: Have all functional requirements tested and passed
- 4/11/2024: Have all non-functional requirements tested and passed

Test Case ID	Test Case Title	Description	Responsible Tester	Planned Test Completion Date
TC-1	Registration	Testing successful case of user registration	Entire Team	4/09/2024
TC-2	Registration	Attempting to register using an already in use email	Entire Team	4/09/2024
TC-3	Registration	Registering with an Invalid Email Format	Entire Team	4/09/2024
TC-4	Registration	Registering using a Weak Password	Entire Team	4/09/2024
TC-5	Registration	Registering with Blank Fields	Entire Team	4/09/2024
TC-6	User Login	Testing successful case of user login	Entire Team	4/09/2024
TC-7	User Login	Logging in using Blank Fields	Entire Team	4/09/2024
TC-8	User Login	Session Persistence	Entire Team	4/09/2024
TC-9	User Profile	Testing successful user profile update	Entire Team	4/09/2024
TC-10	User Profile	Updating User Profile with blank fields	Entire Team	4/09/2024
TC-11	Create New Recipe	Testing successful case of recipe creation	Entire Team	4/09/2024
TC-12	Create New Recipe	Testing failed recipe creation from Missing Title	Entire Team	4/09/2024
TC-14	Recipe Details	Testing successful case of recipe editing	Entire Team	4/09/2024

TC-15	Recipe Details	Testing successful deletion of a recipe	Entire Team	4/09/2024
TC-16	Recipe Details	Testing successful cancelation of recipe deletion	Entire Team	4/09/2024
TC-17	View All Recipes	Testing the fetching recipe function of View All Recipes	Entire Team	4/09/2024
TC-18	Recipe/ Shared Recipe Details	Testing the View All Recipes/Explore New Creation: Filter By Tag and Search Bar	Entire Team	4/09/2024
TC-19	Recipe/ Shared Recipe Details	Testing the View Recipe function of View All Recipes and Explore New Creation	Entire Team	4/09/2024
TC-20	View All Recipes/ Explore New Creation	Testing the View All Recipe/Explore New Creation: Recipe Card Description Quick View	Entire Team	4/09/2024
TC-21	View All Recipes/ Explore New Creation	Testing the View All Recipe/Explore New Creation: Recipe Card Allergen Quick View	Entire Team	4/09/2024
TC-22	View All Recipes	Testing the View All Recipes: Share Recipe function	Entire Team	4/09/2024
TC-23	Explore New Creations	Testing the Explore New Creations: Save Recipe Function	Entire Team	4/09/2024
TC-24	Logout	Testing case of Successful Logout	Entire Team	4/09/2024
TC-25	Logout	Testing successful case of session termination	Entire Team	4/09/2024
TC-26	Password Reset	Testing the request for password Reset Link	Entire Team	4/10/2024
TC-27	Password Reset	Testing Reset password with a valid link	Entire Team	4/10/2024
TC-28	Recipe Management	Testing successful case of embedding a YouTube video into the recipe.	Entire Team	4/10/2024
TC-29	User Dashboard	View User Dashboard	Entire Team	4/10/2024
TC-30	Search/Tag filter	Testing recipe searching function	Entire Team	4/10/2024
TC-31	Foodlists	Testing successful case of creating a new foodlist	Entire Team	4/10/2024
TC-32	Foodlists	Testing successful case of editing an existing foodlist	Entire Team	4/10/2024
TC-33	Foodlists	Testing successful case of deleting an existing foodlist	Entire Team	4/10/2024
TC-34	User Profile	Testing Successful deletion of an existing user profile	Entire Team	4/10/2024

ITC-1	Create New Recipe	Testing how different components integrate the process of the creation of a new recipe.	Entire Team	4/10/2024
ITC-2	Recipe Details	Testing how different components integrate the process of recipe editing, deletion attempt, and deletion cancellation processes within the application	Entire Team	4/10/2024
ITC-3	Recipe Saving & Sharing	Testing how different components integrate the process between the recipe sharing functionality from one user's account and the recipe saving functionality into another user's account, ensuring the integrity of the features and the integrity of the shared data.	Entire Team	4/10/2024
PTC-1	Load Testing for Recipe Exploration Performance	Evaluate the performance of the "View and Explore Recipes" feature under different degrees of user load.	Entire Team	4/10/2024
Security TC-1	User Login	Testing Logging in using an unverified Email	Entire Team	4/10/2024
Security TC-2	User Login	Testing Logging in using Incorrect Password	Entire Team	4/10/2024
Security TC-3	Password Reset	Testing Reset Password With Expired Link	Entire Team	4/10/2024
STC-1	System Test	End-To-End Application functionality	Entire Team	4/10/2024

6.1 Roles and Responsibilities

- Test Manager: Oversees the testing process, ensures the test plan is followed, and manages resources.
    - Delbert Li will fulfill the role of Test Manager.
- Test Lead: Designs test cases, monitors the execution of tests, and is the primary point of contact for testing matters.
    - Lucas Prifti will fulfill the role of Test Lead.
- Test Engineers: Execute test cases, both automated and manual, and document the results.
    - Delbert Li, Lucas Prifti, and Saadman Choudhury will all fulfill the role as the Test Engineers.
- Defect Manager: Manages the defect life cycle, from detection to closure.
    - Saadman Choudhury will fulfill the role as the Defect Manager.

7. Test Environment

Hardware and Software Requirements:

- **Operating System Compatibility:** Tests will be run in environments that support the application's server and client-side software, including Windows, macOS, and Linux distributions for comprehensive coverage.

Database Configuration:

- **MongoDB Atlas:** The primary database service used in both production and testing environments, facilitating data storage, retrieval, and management tasks essential for the application's functionality.
- **Management Tools:** MongoDB Compass will be installed for direct database interaction, allowing testers to inspect, manipulate, and manage test data effectively.

Application & Testing Tools:

- **React and Node.js Architecture:** The test environment mirrors the application's design, with React for the Presentation Layer and Node.js & Express.js for the Server Logic Layer. This alignment is critical for accurately assessing user interactions and backend processes.
- **Selenium WebDriver for Automated Testing:** Selenium WebDriver is selected for its capability to automate browser compatibility and responsive design tests. It plays a vital role in validating the application's user interface across various browsers and screen dimensions, ensuring consistent performance and a seamless user experience.
- **Jest for Unit Testing:** Jest is employed for unit testing within the application's logic and functional units. By using this specific tool, each component of the application is subjected to rigorous testing, verifying that individual parts perform as expected independently of the wider system.
- **Apache JMeter for Performance Testing:** Apache JMeter is utilized to simulate high-traffic conditions and assess the application's performance. This tool helps identify potential bottlenecks and ensures the application can handle anticipated user loads efficiently.

8. Test Entry and Exit Criteria

Entry Criteria:

Before the testing procedure begins, the following conditions must be met to confirm the readiness of the test environment and the testing team:

Test Environment Validation:

- **Software Installation and Compatibility:** Verify the installation of all necessary software across testing devices and servers. This includes ensuring web browsers are up-to-date for Selenium testing and that the operating systems are compatible with Jest.
- **Database Setup:** MongoDB Atlas and Compass must be correctly configured for use in the test environment. This includes preparing test databases with data structures and samples reflective of actual application use cases.
- **Testing Tools Configuration:** Tools such as Selenium, Jest and JMeter should be installed and configured with preliminary test scripts, cases, and parameters aligned with the application's functionality and performance expectations.
- **Performance and Security Benchmarks:** Preliminary benchmarks for application performance and security must be established, setting the standards against which testing outcomes will be evaluated.
- **Test Plan and Cases Review:** The comprehensive test plan, including all test cases and scripts, must be reviewed and approved by the testing team and key stakeholders to ensure completeness and alignment with project objectives.

Exit Criteria:

To exit the testing procedures, the following criteria must be satisfied, indicating that testing objectives have been achieved and the application is ready for deployment or further development stages:

Test Coverage Completion: A minimum of 90% of all planned test cases have been executed, covering critical paths, functionality, performance, and security aspects of the application.

Defect Resolution: All identified defects, especially those categorized as critical and high priority, have been addressed, fixed, and verified through subsequent testing rounds. Documentation of these defects and their resolutions should be complete.

Performance Benchmark Achievement: The application consistently meets or exceeds the established performance benchmarks across all test scenarios, demonstrating its ability to handle expected operational loads.

Security Standards Compliance: Security testing confirms the application adheres to industry-standard security practices, with no critical vulnerabilities unresolved.

9. Test Pass and Fail Criteria

Test Pass Criteria:

A test case is considered to have passed if it meets all of the following conditions:

1. **Expected Outcomes:** The actual results of the test align perfectly with the expected outcomes documented in the test case. This includes correct application behavior, data processing, and user interface display.
2. **Performance Benchmarks:** The application meets or exceeds specified performance benchmarks, including load times, response times, and throughput rates, under various testing conditions as outlined in the test plan.
3. **Security Compliance:** The test case does not uncover new security vulnerabilities, and the application adheres to established security protocols and practices, ensuring data protection and application integrity.
4. **Error-Free Execution:** The test executes without unexpected errors or interruptions, demonstrating the stability and reliability of the application feature or functionality being tested.
5. **Functional Completeness:** All aspects of the test case, including edge cases and interaction with other application components, are covered and performing as intended without any workarounds or issues.

Test Fail Criteria:

A test case is considered to have failed if any one of the following conditions is met:

1. **Mismatched Outcomes:** There is a discrepancy between the actual results of the test and the expected outcomes, indicating a bug, error, or flaw in the application's functionality, user interface, or data handling.
2. **Performance Shortfalls:** The application fails to meet the established performance benchmarks, such as longer than acceptable load or response times, indicating potential performance issues under normal or peak loads.
3. **Security Vulnerabilities:** The test uncovers security vulnerabilities or the application demonstrates non-compliance with the required security standards, posing a risk to user data or application integrity.

4. **Unexpected Errors:** The occurrence of unexpected errors, crashes, or interruptions during the test execution, pointing to stability or reliability issues within the application or its environment.
5. **Incomplete Functionality:** The application feature or functionality under test does not fully meet the requirements or fails to handle edge cases appropriately, necessitating further development or adjustment.

10. Test Suspension and Resumption Criteria

Suspension Criteria:

1. Critical Bug Encountered: If a test case uncovers a critical bug that impedes any further testing, such as a crash or data loss, testing should be suspended until a fix is deployed.
2. Environment Unavailability: If the test environment becomes unstable or unavailable, testing will be paused until the environment is restored to the required state.
3. Resource Unavailability: Should key personnel or critical testing tools become unavailable, testing will be put on hold.
4. Major Requirement Changes: Significant changes to requirements may necessitate a re-evaluation of the test plan and suspension of current testing activities.

Resumption Criteria:

1. Critical Bug Resolved: Once the critical bug has been addressed, and a fix has been verified, testing can resume.
2. Environment Stability: After the test environment is confirmed stable and mirrors production conditions, testing activities can proceed.
3. Resource Availability: Testing can continue once all necessary team members and tools are available again.
4. Updated Test Plan: If requirements change, the test plan should be updated accordingly, and testing can restart with the new or revised cases.

11. Test Design and Execution

Testing will be a combination of automated and manual efforts:

1. Automated Testing: For repetitive and regression tests, automated scripts will be developed using tools like Selenium for UI tests and Jest for unit tests.
2. Manual Testing: Scenarios involving complex user interactions or subjective evaluation, such as usability and ad hoc testing, will be manually conducted.

Process:

1. Creating Test Cases: Based on the requirements and use cases, a comprehensive set of test cases will be created, specifying the actions, inputs, expected outcomes, and pass/fail criteria.
2. Executing Test Cases: Testers will perform the tests as per the test cases. Automated tests will be scheduled and monitored, while manual tests will be carried out systematically.
3. Documentation: Test results will be recorded at the bottom of this document, noting any deviations from expected results and providing evidence where appropriate.

12. Test Data & Defect Management

- Test Data Creation: Test data will be generated that reflects realistic usage scenarios, ensuring both normal and boundary cases are covered.
- Management: Test data will be maintained in a secure and structured manner, with the ability to roll back to initial states if needed.
- Defect Tracking: All defects discovered will be logged into a defect tracking system with severity, steps to reproduce, expected and actual results.
- Defect Resolution: The development team will address defects based on priority, and retesting will occur to confirm resolution.

13. Risk Analysis

1. **Database Deletion Risk:** Certain functionalities, like batch operations in the admin panel, could inadvertently lead to massive data deletion if improperly handled, presenting a high risk of irreversible data loss.

2. **Performance Bottleneck Risk:** The search functionality, without optimized queries or proper indexing, might become a performance bottleneck under heavy load, severely impacting user experience.

15. Test Cases

For each identified use case, provide a corresponding set of test cases, including an ID, description, pre-conditions, test steps, expected results, priority, and pass/fail criteria for each test case.

User Registration

Test Case ID	Title	Description
TC-1	Successful Registration	Successfully register with a valid email, password, and username.
TC-2	Registering with Duplicate Email	Attempt to register with an email that is already in use.
TC-3	Registering with Invalid Email Format	Attempt to register with an email in an invalid format.
TC-4	Registering using Weak Password	Attempt to register with a password that does not meet complexity requirements.
TC-5	Registering with Bank Fields	Attempt to register without filling out all required fields.

User Login

Test Case ID	Title	Description
TC-6	Successful Login	Successfully log in with the correct email and password.
TC-7	Logging in using Blank Fields	Attempt to log in without entering email or password.
TC-8	Session Persistence	Verify the user session is persisted across page reloads.

User Profile Update

Test Case ID	Title	Description
TC-9	Successful Profile Update	Successfully update user profile information.
TC-10	Updating Profile Using Blank Fields	Attempt to update profile leaving required fields empty.

Create New Recipe

Test Case ID	Title	Description
TC-11	Successful Creation	Successfully create a new recipe with all required fields filled.
TC-12	Missing Title	Attempt to create a recipe without a title.
TC-13	Missing Description	Attempt to create a recipe without providing any descriptions.

Edit Recipe

Test Case ID	Title	Description
TC-14	Successful Edit	Successfully edit existing recipes details and save changes.

Delete Recipe

Test Case ID	Title	Description
TC-15	Successful Deletion	Successfully delete an existing recipe.
TC-16	Cancel Deletion	Cancel the deletion operation after initiating it.

Viewing and Searching Recipes

Test Case ID	Title	Description
TC-17	View All Recipes	Successfully fetching all recipes the user has created or saved from the database and seeing them within the View All Recipe page.
TC-18	View All Recipes: Filter By Tag and search	Successfully filtering recipes by selecting or searching a specific tag in the tag bar or search bar.
TC-19	View All Recipes: Recipe Details	Successfully view the details of a specific recipe by clicking the “View Recipe” button on the recipe cars

TC-20	View All Recipes: Description Quick View	Successfully viewing the description of the recipe by clicking on the description box on the recipe card and having the description modal create a pop up for the user to view and close after viewing.
TC-21	View All Recipes: Allergen Quick View	Successfully viewing the allergens of the recipe by clicking on the allergen tabs on the bottom of the recipe card and having the allergen modal create a pop up for the user to view and close after viewing.

Recipe Sharing/Saving

Test Case ID	Title	Description
TC-22	View All Recipes: Share Recipe	Successfully share a recipe to the Explore New Creations page by clicking the share button towards the bottom of the View All Recipes recipe card.
TC-23	Explore New Creations: Save Recipe	Successfully save a recipe to the user's profile through clicking the save recipe button on the bottom of the Shared Recipe Details page.

Logout

Test Case ID	Title	Description
TC-24	Successful Logout	Successfully log out of the

		application.
TC-25	Session Termination	Verify the user session is terminated after logout.

Password Reset

Test Case ID	Title	Description
TC-26	Request Password Reset Link	Successfully request a password reset link.
TC-27	Reset Password With Valid Link	Successfully reset the password using the valid link.

Recipe Tag/Video Management

Test Case ID	Title	Description
TC-28	Add Video Instruction	Successfully add a youtube video link, which will be embedded into the page.

Dashboard and User Interactions

Test Case ID	Title	Description
TC-29	View Dashboard	Successfully view the user's dashboard with personalized content.

Search and Explore

Test Case ID	Title	Description
TC-30	Advanced Recipe Search	Successfully perform an advanced searching utilizing the search bar.

Food Lists Management

Test Case ID	Title	Description
TC-31	Create Food List	Successfully create a Food List with selected recipes and a description.
TC-32	Edit Food List	Successfully edit an existing Food List (add/remove recipes, change title/description).
TC-33	Delete Food List	Successfully delete an existing food list with confirmation prompt to give the user the chance to correct an incorrect click.

Delete User Profile

TC-34	Delete User Profile	Successfully delete the user profile once they have confirmed they want to delete their account.

Integration Testing

ITC - 1	Create New Recipe	Testing how different components integrate the process of a new recipe creation.

ITC - 2	Recipe Details	Testing how different components integrate the process of recipe editing, deletion attempt, and deletion cancellation processes within the application
ITC - 3	Recipe Saving & Sharing	Testing how different components integrate the process between the recipe sharing functionality from one user's account and the recipe saving functionality into another user's account, ensuring the integrity of the features and the integrity of the shared data.

System Test

STC-1	System Test	End-To-End Application Functionality..

Performance Test

PTC-1	Load Testing for Recipe Exploration Performance	Evaluate the performance of the "View and Explore Recipes" feature under different degrees of user load.

Security Testing

SecurityTC-1	User Login	Logging in using an unverified email.
SecurityTC-2	User Login	Logging in using an incorrect password format
SecurityTC-3	Password Reset	Reset Password with an expired link

Test Pass/Fail Report

Test Case ID	Test Case Title	Description	Planned Test Completion Date	PASS/FAIL
TC-1	Registration	Testing successful case of user registration	4/09/2024	PASS
TC-2	Registration	Attempting to register using an already in use email	4/09/2024	PASS
TC-3	Registration	Registering with an Invalid Email Format	4/09/2024	PASS
TC-4	Registration	Registering using a Weak Password	4/09/2024	PASS
TC-5	Registration	Registering with Blank Fields	4/09/2024	PASS
TC-6	User Login	Testing successful case of user login	4/09/2024	PASS
TC-7	User Login	Logging in using Blank Fields	4/09/2024	PASS
TC-8	User Login	Session Persistence	4/09/2024	PASS
TC-9	User Profile	Testing successful user profile update	4/09/2024	PASS
TC-10	User Profile	Updating User Profile with blank fields	4/09/2024	PASS
TC-11	Create New Recipe	Testing successful case of recipe creation	4/09/2024	PASS
TC-12	Create New Recipe	Testing failed recipe creation from Missing Title	4/09/2024	PASS
TC-14	Recipe Details	Testing successful case of recipe editing	4/09/2024	PASS
TC-15	Recipe Details	Testing successful deletion of a recipe	4/09/2024	PASS
TC-16	Recipe Details	Testing successful cancelation of recipe deletion	4/09/2024	PASS
TC-17	View All Recipes	Testing the fetching recipe function of View All Recipes	4/09/2024	PASS
TC-18	Recipe/ Shared Recipe Details	Testing the View All Recipes/Explore New Creation: Filter By Tag and Search Bar	4/09/2024	PASS
TC-19	Recipe/ Shared Recipe Details	Testing the View Recipe function of View All Recipes and Explore New Creation	4/09/2024	PASS
TC-20	View All Recipes/	Testing the View All Recipe/Explore New Creation:	4/09/2024	PASS

	Explore New Creation	Recipe Card Description Quick View		
TC-21	View All Recipes/ Explore New Creation	Testing the View All Recipe/Explore New Creation: Recipe Card Allergen Quick View	4/09/2024	PASS
TC-22	View All Recipes	Testing the View All Recipes: Share Recipe function	4/09/2024	PASS
TC-23	Explore New Creations	Testing the Explore New Creations: Save Recipe Function	4/09/2024	PASS
TC-24	Logout	Testing case of Successful Logout	4/09/2024	PASS
TC-25	Logout	Testing successful case of session termination	4/09/2024	PASS
TC-26	Password Reset	Testing the request for password Reset Link	4/10/2024	PASS
TC-27	Password Reset	Testing Reset password with a valid link	4/10/2024	PASS
TC-28	Recipe Management	Testing successful case of embedding a YouTube video into the recipe.	4/10/2024	PASS
TC-29	User Dashboard	View User Dashboard	4/10/2024	PASS
TC-30	Search/Tag filter	Testing recipe searching function	4/10/2024	PASS
TC-31	Foodlists	Testing successful case of creating a new foodlist	4/10/2024	PASS
TC-32	Foodlists	Testing successful case of editing an existing foodlist	4/10/2024	PASS
TC-33	Foodlists	Testing successful case of deleting an existing foodlist	4/10/2024	PASS
TC-34	User Profile	Testing Successful deletion of an existing user profile	4/10/2024	PASS
ITC-1	Create New Recipe	Testing how different components integrate the process of the creation of a new recipe.	4/10/2024	PASS
ITC-2	Recipe Details	Testing how different components integrate the process of recipe editing, deletion attempt, and deletion cancellation processes within the application	4/10/2024	PASS
ITC-3	Recipe Saving & Sharing	Testing how different components integrate the process between the recipe sharing functionality from one user's account and the recipe saving functionality into another user's account, ensuring the integrity of the features and the integrity of the shared data.	4/10/2024	PASS

PTC-1	Load Testing for Recipe Exploration Performance	Evaluate the performance of the "View and Explore Recipes" feature under different degrees of user load.	4/10/2024	PASS
Security TC-1	User Login	Testing Logging in using an unverified Email	4/10/2024	PASS
Security TC-2	User Login	Testing Logging in using Incorrect Password	4/10/2024	PASS
Security TC-3	Password Reset	Testing Reset Password With Expired Link	4/10/2024	PASS
STC-1	System Test	End-To-End Application functionality	4/10/2024	PASS