

STP Document

Randommer
random stuff here

Mohammed Hussien

Table of Contents

Planned schedule.....	3
Purpose of this document.....	4
Description Of the System	4
Glossary and Abbreviations.....	6
Testing plan	7
Resources.....	7
Software/hardware needed.....	7-8
functional tests	9
non- functional tests	9,10
Starting and existing criteria.....	11
Traceability Table.....	12
Tests Tree	12-13
Hazards table	15-16

Planned Schedule

A step in the project process	Start date	End date
Preparation of STP document	07.03.2024	07.03.2024
Preparation of STD document	07.03.2024	07.03.2024
Round of tests #1	07.03.2024	08.03.2024
Round of tests #2	08.03.2024	08.03.2024
Round of tests #3	08.03.2024	08.03.2024
Preparation of STR document	08.03.2024	09.03.2024

Purpose Of This Document

The purpose of the document is to define a framework program for testing. This framework plan will include all relevant topics for planning and performing the tests, such as the test topics and types of tests, the schedule, and the planned work method for testing the system.

The document will also be used as base for writing a detailed test plan (STD) in which the various tests will be detailed step by step. After that, the tests will be carried out according to the instructions in this document.

Description Of the Randommer System

Randommer is an innovative online platform designed to provide users with a seamless experience in generating random data across various categories. With a simple and intuitive interface, Randommer aims to cater to the needs of individuals, researchers, developers, and businesses requiring randomized data for testing, simulations, statistical analysis, and more.

Purpose: Randommer's primary objective is to offer a reliable and efficient solution for generating random data, addressing the diverse requirements of users across different industries and domains. By providing a wide range of randomization options and customizable parameters, the platform empowers users to obtain precise and tailored datasets suited to their specific use cases and applications.

Randommer seeks to streamline the process of acquiring random data, thereby facilitating research, experimentation, and development in various fields.

Key Features and Functions:

1. **Data Generation:** Randommer allows users to generate random data across multiple categories, including numbers, text, dates, and boolean values, among others.
2. **Customization Options:** Users can customize various parameters such as data type, length, format, and distribution to generate randomized datasets according to their preferences and requirements.
3. **Batch Processing:** Randommer supports batch processing, enabling users to generate large volumes of random data efficiently and effectively.
4. **Export Options:** Users can export generated data in various formats such as CSV, JSON, XML, or plaintext, facilitating seamless integration with other applications and systems.
5. **Randomness Assurance:** Randommer employs advanced algorithms and techniques to ensure the randomness and unpredictability of generated data, adhering to established standards and best practices.

Stakeholders:

1. **Researchers and Scientists:** Randommer caters to researchers and scientists who require randomized datasets for experiments, simulations, and statistical analysis in various fields such as psychology, sociology, and biology.
2. **Developers and Testers:** Randommer serves developers and testers who need random data for software testing, quality assurance, and simulation purposes in application development and debugging processes.
3. **Data Analysts:** Randommer provides data analysts with a tool to generate synthetic datasets for exploratory data analysis, modeling, and visualization, facilitating insights discovery and decision-making.

Glossary and Abbreviations

Glossary

- GUI (Graphical User Interface): The design of user interfaces based on specified requirements.
- Functional Testing: Verification that fundamental system functions operate correctly.
- Maintenance Testing: Examination of the functionality of a modified system following changes, updates, or alterations in the working environment.
- STP (System Test Plan): A comprehensive project planning document encompassing strategy, schedule, and topic tree.
- STD (System Test Design): Detailed documentation outlining the testing plan.
- STR (System Test Results): A concise document summarizing test results after three cycles.
- Traceability Matrix: A document that correlates any two baselined documents that require a many-to-many relationship to determine the completeness of the relationship.

Abbreviations

- QA: Quality Assurance
- CEO: Chief Executive Officer
- HR: Human Resources

Testing Plan

The testing team will communicate the requirements to the development team based on the planned tests. The development team will work according to these requirements to maximize efficiency.

Resources: Team leader, 4 testers. A total of 5 team members will be assigned to this project.

i

1. Computers:

- High-performance desktops or laptops to run test scripts and perform manual testing.

2. Network Equipment:

- Routers and switches to simulate different network conditions for testing randommer performance under various network scenarios.

3. Mobile Devices:

- Various smartphones and tablets to test randommer website across different platforms (iOS, Android).

4. Virtual Machines:

- Set up virtual machines for testing on different operating systems and browser combinations.

5. Storage Devices:

- Sufficient storage space to store test data, logs, and files generated during testing.

Software Needed:

1. Operating Systems:

- Install and configure various operating systems for testing, including Windows, macOS, and Linux distributions.

2. Browsers:

- Latest versions of popular browsers (Google Chrome, Microsoft Edge, Safari) for cross-browser compatibility testing.

3. Mobile Emulators/Simulators:

- Emulators or simulators to test randommer mobile website on different devices and screen sizes.

4. Performance Testing Tools:

- Tools like Apache JMeter, LoadRunner, or Gatling for performance and load testing to simulate high user traffic.

5. Automation Testing Tools:

- Selenium, Appium, or similar tools for automating functional tests across web and mobile platforms.

6. Database Management System:

- Database systems (MySQL, PostgreSQL) for testing randommer data handling and retrieval functionality.

7. Collaboration Tools:

- Communication and collaboration tools (e.g., Slack, Microsoft Teams) for effective communication among the testing team.

8. Test Management Tools:

- Test case management tools (e.g., **TestRail**, **Jira**) to organize, execute, and track test cases.

This document will be approved by the testing team leader and the project manager. Following approval, the senior tester will build the STD document based on it. The STD document will then undergo approval by the testing team leader.

After approval of the STD document, three rounds of testing will be conducted. At the end of these rounds, the STR document will be prepared by the senior tester, undergo approval by the testing team leader, and serve as the final documentation.

Before the start of testing rounds, **functional tests** will be performed, including:

- 1) **Unit Testing:** To test individual units or components of a software application.
- 2) **Sanity Testing:** To verify that the most important functionalities of a software application work correctly.
- 3) **Integration Testing:** To verify the interactions and interfaces between different components or systems within the application.
- 4) **Regression Testing:** To ensure that new code or changes do not affect the existing functionality of the software.
- 5) **API Testing:** To validate the functionality of an API by testing its endpoints and request-response mechanisms.
- 6) **UI Testing:** To validate that the user interface elements and interactions function correctly according to the design.
- 7) **End-to-End Testing:** To evaluate the entire software system's functionality from start to finish, simulating real user scenarios and interactions.

Afterwards, the following **non-functional tests** will be conducted including:

- 1) **Performance Tests (Load + Stress + Volume):** they assess how well a system performs under various conditions, including heavy loads, stressful situations, and large data volumes.
- 2) **Security Tests:** Security tests aim to identify vulnerabilities and weaknesses in a system to ensure protection against unauthorized access.
- 3) **Upgrade and Installation Test:** Upgrade and installation tests verify the smooth installation of software upgrades and updates, ensuring that the system remains stable and functional.
- 4) **Recovery Tests:** they assess how well a system can recover from failures including data loss or system crashes.
- 5) **Localization and Globalization Testing:** Localization testing verifies that a software application adapts to specific regional or cultural requirements, while globalization testing ensures its compatibility with diverse international settings.
- 6) **Usability Testing:** evaluates the user friendliness and overall user experience of a software application to ensure it meets user expectations.
- 7) **Compatibility Testing:** Compatibility testing ensures that a software application works seamlessly across different devices, browsers, operating systems.

Starting and exiting Criteria

- **Criteria for starting the tests:**

- 100% of the planned sanity tests were carried out and passed successfully.
- 100% of planned functional test cases have been created and reviewed.
- A traceability matrix is established, linking each test case to specific requirements.
- The testing environment, including necessary configurations, data, and tools, is prepared and verified.
- Sufficient and accurate test data for both positive and negative scenarios is available.
- The test plan, detailing the testing approach, objectives, and schedules, has been reviewed and approved.

- **Completion/Release Criteria:**

- 100% of planned functional and non-functional tests have been executed, and results have been documented.
- 100% of test cases passed successfully.
- All critical bugs have been fixed at this point.
- The remaining bugs are at low severity levels, with no high-severity issues affecting functionality.

Traceability Table

Business Requirement	REQ ID	Functional/non Functional tests	Test Case ID	Defects?	Status
Phone number	1	Verify Getting x phone numbers of the chosen country.	1.a	None	Verified
		Verify that the numbers we got belong to the country we choose.	1.b	None	Verified
		Verify validating a phone number for a random country chosen from the countries list	1.c	None	Verified
Credit card	2	Verify getting all card types successfully	2.a	None	Verified
		Verify that we can get a random card detail of a random card type we choose	2.b	None	Verified
password	3	Verify getting a strong password	3.a	None	Verified
		Verify that the password we got contains one Upper letter	3.b	None	Verified
		Verify that the password we got contains at least one digit.	3.c	None	In-progress
		Verify that the password we got contains at least one special char.	3.d	None	Verified
text	4	Verify posting a new Json file with all x words replaced by y words	4.a	None	Verified
		Verify posting a new Json file with all upper-case letters changed to lower case	4.b	None	In-progress

Testing Tree

1. Password

Functional testing:

- a. Get password.
 - i. Verify successfully getting a strong password with the wanted conditions.
 - ii. Verify successfully getting a strong password with at least one special char.
 - iii. Verify successfully getting a strong password with at least one Upper-case letter.
 - iiii. Verify successfully getting a strong password with at least one digit.
 - iiiii. Verify successfully getting a strong password with at least one special char.

Non-functional:

performance Testing

- b. Evaluate the time taken to generate a strong password.

Scalability Testing

- c. Determine the ability of the system to handle simultaneous requests for password generation.

2. Credit Card:

Functional testing:

- a. credit card:
 - i. Verify successfully retrieving details of all card types.
 - ii. Confirm that the system can retrieve random card details for any chosen card type.
 - iii. Verify that that the cvv of the card detail we got is valid.

Non-functional:

Scalability Testing:

- b. Determine the system's ability to handle a large volume of card detail requests.

performance Testing:

- c. Evaluate the time taken to retrieve card details.

Security Testing:

- d. Assess the security measures in place for handling card details.

3. Text:

Functional testing:

- i. Verify successfully replacing all occurrences of 'x' words with 'y' words in a given JSON file.
- ii. Verify successfully replacing all upper letters with small letters.
- iii. Verify successfully replacing all lower letters with upper letters.
- iiii. Verify successfully humanizing a text.

Non-functional:

performance Testing:

- b. Evaluate the time taken to perform text manipulation operations.

Compatibility Testing:

- d. Determine the system's ability to handle increasing loads of text manipulation requests.

Security Testing:

- e. Assess the security of text manipulation operations, particularly regarding data integrity and confidentiality.

Hazard Table

Responsible	Description	Action	Hazard Description	Risk Level	Damage	Chance	Hazard	#
	-Description Of prevention method	Enclose Monitoring Acceptance	What will happen in case of the hazard	Chance Multiplied by Damage	10-1	The probability of the hazard occurring, ranging from 0 to 1.		
System	NA	Monitoring	Unable to connect as a user	5	10	0.5	Bad Internet Connectivity	1
	Finding a tester for the project length	Enclose	Bad testing and coverage	1.6	8	0.2	New Testers	2
System		Monitoring	Unable to login and retrieve info from the DB	7	7	0.5	Server Crash	3
	Postponing / finding replacement	Enclose	Vacations	7	7	1.0	Vacations	4
System	Adding servers	Enclose	System Crash	5	10	0.5	Weak Server	5

HR	Hiring a stable worker	Monitoring	Lowering Team Morale	5	10	0.5	Employee quitting	6
QA Lead	Hiring Experienced Testers	Enclose	Failure to meet the schedule	1	10	0.1	Inexperienced Testers	7
CEO	More flexible customer	Monitoring	No income for the company	3	10	0.3	Contract Termination	8
CEO	QA Lead bad Management	Enclose	Unsatisfied Customer	10	10	1	Failure to meet the schedule	9
CEO	An appointment must be made with the customer and it should be noted to him that it will not be possible to make changes after the system is established	Monitoring	Failure to be prepared for changes by the customer will not ensure a professional, accurate and correct inspection	3.5	7	0.5	Customer requirements document changes frequently during the project	10