TEST PLAN FOR

CRYPTOCURRENCY ANALYSIS WEBAPP

ChangeLog

Version	Change Date	By	Description
V1	04/11/2023	Akanksha Mishra	Fixed testing bugs
V2	06/11/230	Anushraya Sharma	Updated test cases

1 Introduction

The Cryptocurrency Analysis WebApp is developed to provide the facility to present all information and trends of cryptos in a structured and centralized manner ranging from history and analysis of past performance to reliable predictions about future dips and rises along with the market cap and rank of these cryptos in order to provide profitable future investments in cryptocurrencies.

For the Cryptocurrency Analysis WebApp project, the testing strategies will focus on ensuring the application's reliability, performance, and accuracy in providing cryptocurrency-related information and predictions. The following test strategies will be employed:

Functional Testing:

- Test the core functionalities of the application, such as historical data retrieval, analysis tools, prediction algorithms, real-time value updates, and currency conversions.
- Ensure that user data, especially financial information, is stored and transmitted securely.

Security Testing:

- Identify and address security vulnerabilities, such as data breaches and unauthorized access.
- Ensure that user data, especially financial information, is stored and transmitted securely.

Compatibility Testing:

- Test the application on various web browsers and devices to ensure crossbrowser and cross-platform compatibility.
- Ensure that the application is responsive and usable on different screen sizes.

Process and Workflow

The testing process for the Cryptocurrency Analysis WebApp can follow the following workflow:

- Requirement Analysis: Review project requirements and user stories related to the application's features and functionality.

- Test Planning: Define test objectives, test scope, and test criteria. Develop a test plan outlining the testing phases, schedule, and resources required.

- Test Design: Create test cases and test scenarios based on functional requirements. Develop test data, including historical cryptocurrency data for testing analysis tools.

- Test Execution: Execute test cases, including functional, performance, security, compatibility, and usability testing. Log and track defects and issues, if any.

- Test Reporting: Generate test reports that include test results, identified issues, and recommendations. Communicate with the development team to resolve defects.

- Retesting and Regression Testing: After issue resolution, retest affected areas to ensure that defects are fixed. Perform regression testing to verify that the fixes did not introduce new issues.

- Sign-off and Deployment: Fix the remaining bugs and finally release the application for production use.

Methodologies

The testing methodologies used for the project may include a combination of the following:

- Manual Testing: Manual testing will be essential for exploring the application's usability and conducting user acceptance testing. It is a software testing process in which test cases are executed manually without using any automated tool. All test cases executed by the tester manually according to the end user's perspective.

- Automated Testing: Automation can be employed for repetitive tasks, such as performance testing and regression testing. It is the application of software tools to automate a human-driven manual process of reviewing and validating a software product.

1.1 Scope

1.1.1 In Scope

Functional Requirements to be tested:

- User Authentication: Test user registration, login, and password recovery processes. Verify that user accounts are secure and private.

- User Dashboard: Ensure that users have a personalized dashboard to track their favorite cryptocurrencies and access their investment portfolios.

- Search and Filtering: Validate search functionality for finding specific cryptocurrencies or market data. Verify the effectiveness of filters for refining search results.

- Data Visualization: Test the graphical representation of data, including charts, graphs, and visual indicators for historical and real-time data.

Non-Functional Requirements to be tested:

- Performance: Test the application's response times and ensure it can handle a high volume of users and data without slowing down. Verify that real-time data updates are timely and smooth.

- Security: Conduct security testing to identify and address vulnerabilities, including data encryption, user authentication, and secure API connections.

- Compatibility: Test the application's compatibility with various web browsers, operating systems, and mobile devices.

- Usability: Evaluate the user interface for user-friendliness and ease of navigation. Gather feedback from users to improve the application's usability.

1.1.1 Out of Scope

Requirements which will not be included in the scope of testing:

- Financial Advice: Providing financial advice or specific investment recommendations is outside the scope of testing. The application should not be tested for its ability to offer personalized investment advice.

- Regulatory Compliance: Compliance with local or international cryptocurrency regulations and legal requirements is not within the scope of testing. The application should not be tested for regulatory compliance.

- Integration with External Investment Platforms: Any integration with external cryptocurrency trading or investment platforms is out of scope for testing. The application's interactions with third-party services for trading will not be tested.

- Market Manipulation Prevention: The prevention of market manipulation or fraud in the cryptocurrency markets is not the application's responsibility and is not within the scope of testing.

- Support for All Cryptocurrencies: The application may not support testing for all existing cryptocurrencies, as the scope may be limited to specific cryptocurrencies based on the app's design and data sources.

1.2 Quality Objective

MAIN OBJECTIVE:

The main objective of the testing project for the Cryptocurrency Analysis WebApp is to validate that the application meets its intended purpose and functions effectively, while also conforming to specified quality standards and client requirements.

TESTING OBJECTIVES:

Ensure Conformance to Functional and Non-Functional Requirements: Verify that the application adheres to all functional requirements, such as historical data analysis, real-time data updates, and predictive analysis. Confirm that non-functional requirements, including performance, security, and usability, are met.

Validate Quality Specifications: Ensure that the application meets the quality specifications and standards defined by the client, including accuracy, reliability, and data presentation. Identify and Resolve Bugs/Issues: Detect and report any defects, bugs, or issues within the application. Ensure that these identified issues are resolved and validated before the application goes live to provide a smooth user experience.

Enhance User Confidence: Improve user confidence in the application's reliability, accuracy, and security. Help users make informed investment decisions by ensuring that the application provides valuable insights and data.

Support Profitable Investments: Ultimately, the primary objective is to help users make profitable investments in cryptocurrencies by providing them with the necessary information and tools to make informed decisions.

1.3 Roles and Responsibilities

Anushraya: She was responsible for developing the entire front-end of the project including the API integration and data representation. She has also performed the manual testing of the project.

Akanksha: She was responsible for performing the database integration as well as creating and integrating wallet for successful crypto transactions. She has also performed the automated testing of the project.

2 Test Methodology

2.1 Overview

An Agile methodology is the most suitable for my project as it is a highly flexible and collaborative approach that emphasizes continuous improvement and customer feedback. Well-suited for projects with evolving requirements, complex features, and a need for quick adaptation to changes. Given the dynamic nature of the cryptocurrency market and the need to continuously enhance and update the application to stay relevant, Agile could be a strong choice. It enables close collaboration with users and stakeholders for ongoing feedback and feature prioritization.

2.2 Test Levels

The testing to be performed includes **Unit testing** for the individual Login component as well as Integration and System Testing**.**

2.3 Test Completeness

Criteria to check Test Completeness would be

- 100% test coverage
- All Manual & Automated Test cases executed
- All open bugs are fixed or will be fixed in next release
- All content and style transfer tests have been executed, ensuring that various input images have been processed successfully and meet the defined content and style transfer criteria.

3 Test Deliverables

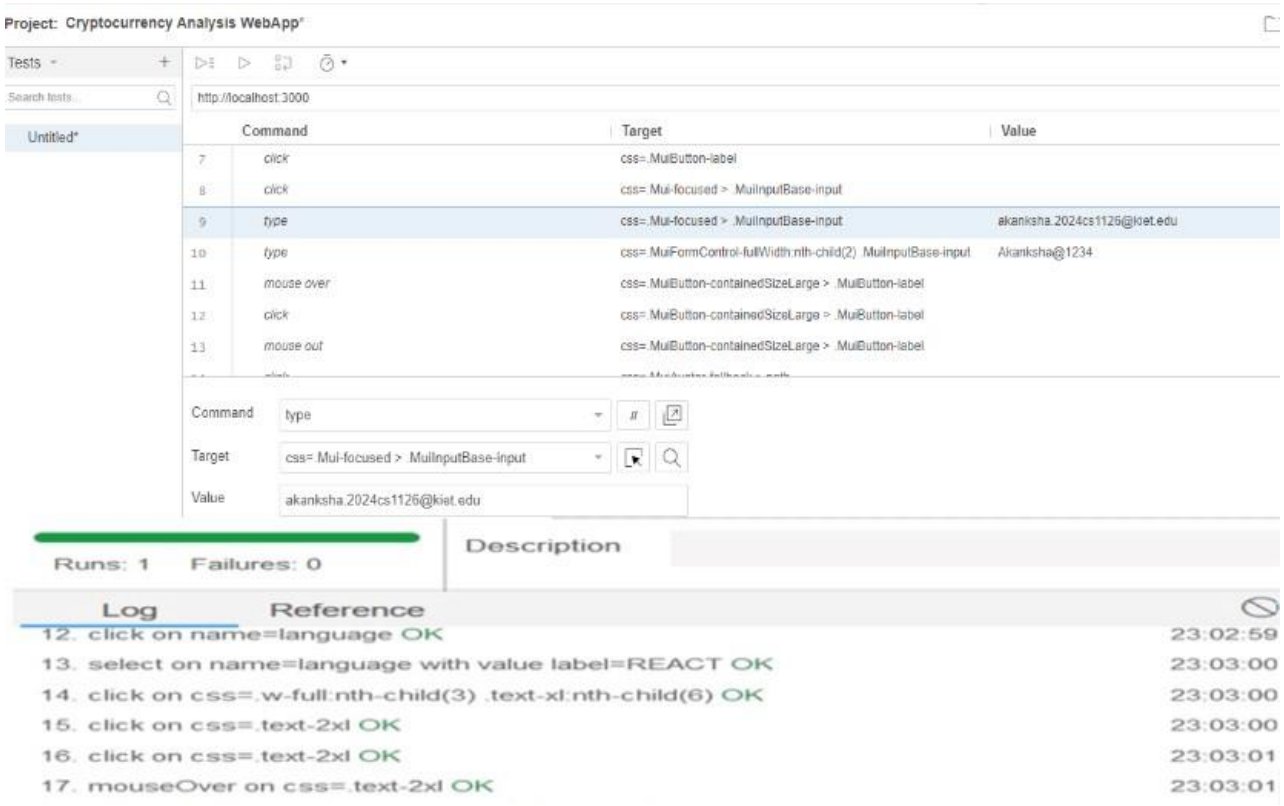
Manual Testing:

1. Boundary Value Analysis for User Login

Range	Username(Number of Characters)	Password(Number of Characters)
Min	4	4
Min+1	5	5
mid	17	7
Max-1	29	9
max	30	10

Testcase ID	Username	Password	Excepted Outcome	Actual Outcome	Pass/Fail
1	Anil	At@1	Login successful	Login Successful	Pass
2	Hin@	An@12	Login successful	Login successful	Pass
3	Rita	Anl@154	Login successful	Login successful	Pass
4	Rit1	Anil@1287	Login successful	Login successful	Pass
5	Riya	Anil@45721	Login successful	Login successful	Pass
6	Avi	An@12	Error	Error	Pass
7	Kavu2	Akq@123	Login successful	Login Successful	Pass
8	Anya@	Ake@6532	Login successful	Login Successful	Pass
9	Aditi	Kaa@964	Login successful	Login Successful	Pass
10	Sanvi	Hi@01	Login successful	Login Successful	Pass
11	Arti8	Ri@	Error	Error	Pass
12	ShanayaTripathi12345@gmail.com	rejE@7	Login successful	Login successful	Pass
13	Shanaya@gmail.com	dey6@9W	Login successful	Login successful	Pass
14	ShanayaTripathi7892@gmail.com	Abrigjw@5Fy	Error	Error	Pass
15	ShanayaTripathi7892@gmail.com	Sh@34	Login successful	Login successful	Pass
16	ShanayaTripathi12347@gmail.com	Weg	Error	Error	Pass
17	Aman	dkE@54	Login successful	Login successful	Pass
18	Sakshi	Iwm@4	Login successful	Login successful	Pass
19	Ayush	dd@4D	Login successful	Login successful	Pass
20	Si	kr@irP	Error	Error	Pass

Automated Testing (Selenium Tool)



3.1 Testing Tools

Automation Tools:

- Selenium: Selenium is a widely used automation tool for testing web applications. In your project, Selenium can automate the testing of various functionalities on the language selection and learning guidance platform, helping ensure functionality and user interface consistency.

3.2 Test Environment

i. Modern operating system (e.g., Windows 10, macOS, or Linux).
ii. Common web browsers (e.g., Chrome, Firefox, Edge).
iii. Testing tools (Mantis for bug tracking, Selenium for automation).
iv. Additional software like an integrated development environment (IDE) and virtualization tools if needed.

4 Terms/Acronyms

Make a mention of any terms or acronyms used in the project

TERM/ACRONYM	DEFINITION
API	Application Program Interface
IDE	Integrated Development Environment
AUT	Application Under Test
UAT	User Acceptance Testing