SE 3XA3: Software Requirements Specification
CryptoMetrics

Vanshaj Verma, Vermav2
Himanshu Aggarwal, aggarwah
Saif Fadhel, fadhels

February 11, 2022

Contents

List of Tables

List of Figures

Table 1: **Revision History**

Date	Version	Notes
Feb 4, 2021	1.0	Initialize the document
Feb 11, 2021	1.1	Update and finalize the document

1   Project Drivers

1.1   The Purpose of the Project

The purpose of the project is to implement a dashboard to display data visualizations on cryptocurrency prices and their trends.

1.2   The Stakeholders

The stakeholders of the project include the development team, cryptocurrency enthusiasts, the professor, and the teaching assistants of the course.

1.2.1   The Client

The clients of the project includes the TAs and the professor as they have assigned the development team the task of creating software and provide valuable feedback that will directly influence the development of the software.

1.2.2   The Customers

The customers of the project includes beginner and casual cryptocurrency traders, cryptominers, small firms, local businesses and anyone who is interested in cryptocurrency.

1.2.3   Other Stakeholders

The other stakeholders are the development team which is composed of software designers, developers, and testers who are invested in the success of the project.

1.3   Mandated Constraints

C1.

- Description: The project shall be a web application written in JavaScript.

- Rationale: Using a JavaScript framework such as Next.js, the development team can create a user-friendly web application that is modular and allows for rapid feature deployment.

- Fit Criterion: The project shall run on any modern web browser supporting JavaScript such as Google Chrome, Mozilla Firefox, and Microsoft Edge.

1.4   Naming Conventions and Terminology

Acronym/Term	Description
CryptoMiner	The name of the project in development.
Cryptocurrency	Digital currency secured by cryptography that uses a decentralized system to record transactions.
Browser	Software application utilized to access the World Wide Web.
TA	Teaching Assistant who is a client of the project.
UI	User Interface.
API	Application Programming Interface
JS	JavaScript. The language used to create the project.
React	JavaScript library that was used to create the reference project.
NextJS/Next.js	The specific React framework that will be utilized to develop and organize the web application.
URL	Uniform Resource Locator.
FR	Functional Requirements.
NFR	Non-functional Requirements.
Bug	Fault in the code that produces an incorrect/unexpected result.
Database	Organized collection of data.

Table 2: Description of Terms and Acronyms

1.5   Relevant Facts and Assumptions

1.5.1   Relevant Facts

The number of lines of code in the original, existing open-source project is approximately 395.

1.5.2   Assumptions

- The user is interacting with the web application in an up-to-date browser.

- The user is proficient in the English Language.

- The user has a very basic understanding of how to navigate a website.

- The user has background knowledge on Cryptocurrency and how their prices fluctuate.

2 Functional Requirements

2.1 The Scope of the Work and the Product

The scope of the project involves making the existing cryptocurrency dashboard of the original project more modular and extending its functionality to provide users with more options to interact with cryptocurrency data.
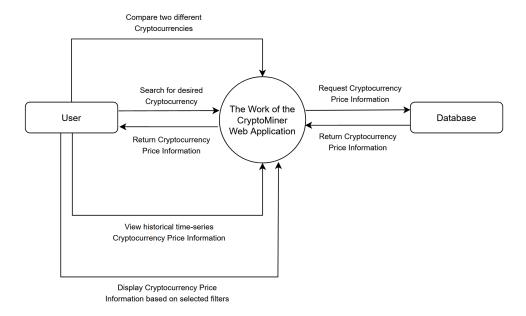
2.1.1 Work Partitioning

Figure 1: Work Context Diagram

Event Name	Input and Output	Summary
1. Search for Desired Cryptocurrency	In: c_name Out: c_name Price Information	The user can search for a desired Cryptocurrency's price information for the system to display.
2. Compare Two different Cryptocurrencies.	In: Cryptocurrency 1 and Cryptocurrency 2 Out: Price Information of both Cryptocurrencies	Current and Historical Price Information of both Cryptocurrencies are directly compared.
3. View Historical Cryptocurrency Price Information	In: Cryptocurrency record Out: Time-Series Cryptocurrency data	Historical data of the desired Cryptocurrency will be displayed by the system in an organized manner.
4. Display Cryptocurrency Price Information Given filters	In: Filter Names Out: Cryptocurrency price Information that pertains to the selected filters.	The list of cryptocurrencies will be reduced down for the system to display only relevant cryptocurrencies based on selected filters.

Table 3: Business Event List

2.1.2 Individual Product Use Cases

1. User wants to view time series data for a cryptocurrency

   (a) The user launches the program.

   (b) The system shows the user a table of cryptocurrencies.

   (c) The user selects a cryptocurrency.

   (d) The system navigates the user to a details page where a chart is displayed with all historical time series data related to the selected cryptocurrency.

2. User wants to sort a list of cryptocurrencies

   (a) The user launches the program.

   (b) The system shows the user a table of cryptocurrencies.

   (c) The user clicks on a heading of the table.

(d) The system sorts the cryptocurrency list based on the heading selected.

3. User wants to filter a list of cryptocurrencies

   (a) The user launches the program.
   (b) The system shows the user a table of cryptocurrencies and buttons to add filters.
   (c) The user uses the filter buttons to add specific filters.
   (d) The system applies the filters to the cryptocurrency list and the updated list is displayed.

4. User wants to compare different cryptocurrencies

   (a) The user launches the program.
   (b) The system shows the user a table of cryptocurrencies and a "select" button.
   (c) The user clicks the "select" button.
   (d) The system displays checkboxes beside each cryptocurrency name.
   (e) The user checks all the cryptocurrencies that need to be compared and clicks on the "compare" button.
   (f) The system navigates the user to a comparison page where a chart is displayed with all historical time series data related to the selected cryptocurrencies.

2.2 Functional Requirements

FR-1 **Description**: The product shall display every cryptocurrency's relevant pricing information in a table-like manner.
Rationale: This will ensure that the information pertaining to a cryptocurrency is organized and can be sorted with filters easily.
Priority: High

FR-2 **Description**: The product shall allow users to search for a desired cryptocurrency to retrieve all relevant pricing information about it.
Rationale: This will ensure that viewing the relevant pricing information associated with a specific cryptocurrency is efficient and

quick.
Priority: High

FR-3 **Description**: The product shall display a desired cryptocurrency's historical pricing information data hourly, weekly, monthly and yearly.
Rationale: When a cryptocurrency's historical prices are displayed, it will help the user make more informed investment decisions as they will be able to predict the pricing trajectory easily.
Priority: High

FR-4 **Description**: The product shall allow users to directly compare the current and historical pricing of one cryptocurrency with another.
Rationale: Through giving users the ability to compare the performance metrics of two different cryptocurrencies, they will have a better idea of the scale of the discrepancies between them.
Priority: High

FR-5 **Description**: The product shall allow the users to filter and display cryptocurrencies along with their associated pricing information in a desired order.
Rationale: Providing the user with the ability to filter the cryptocurrencies, for example, from highest to lowest market cap values, will save them much more time navigating the web application to reach their desired cryptocurrency quicker.
Priority: High

FR-6 **Description**: The product shall allow users to manually select which cryptocurrency along with its associated pricing information can be shifted around the web application.
Rationale: Providing the user with the ability to manually rearrange the order in which a cryptocurrency's information is displayed will give them control over specific cryptocurrencies they would like to access more information about quicker and more efficiently.
Priority: High

FR-7 **Description**: The product shall consistently refresh the web application with up-to-date cryptocurrency data every minute.

Rationale: This will allow the user to stay informed about the current prices of cryptocurrencies to make more informed investment decisions.
Priority: High

FR-8 **Description**: The product shall display an error message to the user if an API is unable to load cryptocurrency data.
Rationale: Giving a user direct feedback during a failed API call will allow them to identify an issue as opposed to waiting for a non-existent response from the product.
Priority: High

FR-9 **Description**: The product shall call the API at least two more times after a delay of 10 seconds if the initial API call results in an error.
Rationale: This will ensure that the product will have a chance to correct any initial data loading failures without the user's input.
Priority: High

FR-10 **Description**: The product should display temporary skeleton placeholder graphics while data is being loaded.
Rationale: This function will provide the user with a clear view of the system retrieving information and conveys the state of the product more accurately.
Priority: Low

FR-11 **Description**: The product shall timeout API calls after 10 seconds.
Rationale: Timing out API calls for failing after a given threshold of time ensures that the user is not wasting their time waiting longer than they need to for a response.
Priority: Medium

FR-12 **Description**: The product should give the user the ability to manually change the theme.
Rationale: Providing the user with customisation options like changing the theme and font of the website will allow them to better personalise their experience.
Priority: Low

FR-13 **Description**: The product shall allow users to save their preferences and load them automatically upon starting the product.
Rationale: Automatically loading up user preferences allows the user to save time.
Priority: Medium

3 Non-functional Requirements

3.1 Look and Feel Requirements

3.1.1 Appearance Requirements

NFR-1 The product shall be attractive and pleasing to a younger audience.
Rationale: To maintain the attention of the users, it is important that they like the look and feel of the product.
Fit Criterion: Within a minute of their first encounter with it, 85% of a sample of young users shall start using the product without prompt.

NFR-2 The product shall be consistent with the user's device's theme preferences.
Rationale: It is important to provide light and dark modes of the product that switch automatically with the device's preferences as this will affect how many users want to use the product.
Fit Criterion: The product shall adapt to the user's device's theme preferences on first startup 99% of times.

3.1.2 Style Requirements

NFR-3 The product shall feel professional, trustworthy, and informative.
Rationale: To gain the trust of the users, it is important that the product feels professional, trustworthy and informative.
Fit Criterion: After their first encounter with the product, 80% of representative users shall agree that the product feels professional, trustworthy and informative.

NFR-4 The product shall have a consistent font and theme throughout all the components.

8

Rationale: To ensure that the components match each other, it is important that they share the same fonts and styles.
Fit Criterion: 85% of a sample group shall not be able to notice a difference in the font and theme of any of the components.

3.2   Usability and Humanity Requirements

3.2.1   Ease of Use Requirements

NFR-5   The product shall be intuitive and easy to use by anyone above the age of 10.
Rationale: For the product to reach mass audience, it is important for it to be intuitive and easy to use.
Fit Criterion: 90% of sample representative users above the age of 10 shall be able to successfully complete specified tasks within a set amount of time.

NFR-6   The product shall require no prior training or education, except for a basic understanding of the English language and familiarity with navigating around websites.
Rationale: For the product to reach a wide range of users with different backgrounds in education, it must be not require any previous education or experience to use the product.
Fit Criterion: 80% of a sample English-speaking group shall be able to navigate through different pages of the product without any training or education.

3.2.2   Personalization and Internationalization Requirements

NFR-7   The product shall allow the user to save preferences.
Rationale: Users should be able to save their preferences, close the web page and come back another time to the same preferences they had initially set. This will help avoid time wasting in re-adjusting the product.
Fit Criterion: 95% of sample representative users shall be able to save their preferences without errors at least 9 out of 10 times.

3.2.3 Learning Requirements

NFR-8 The product shall be able to be used by anyone without any training.
Rationale: Users should be spending time on learning and understanding different trends in the cryptocurrency market, not on learning how to navigate through the product.
Fit Criterion: 75% of a sample group shall be able to complete a set of tasks within a specified amount of time.

3.2.4 Understandability and Politeness Requirements

NFR-9 The product shall use icons that are universally recognized by the users.
Rationale: For the product to be usable by users around the world, it is important for the icons to be universally recognizable.
Fit Criterion: 95% of users in a diverse sample group shall be able to understand what the icons represent.

3.2.5 Accessibility Requirements

NFR-10 The product shall be usable on both desktop and mobile devices.
Rationale: Many users around the world do not have a laptop/desktop. It is important that they be able to view the product on their phone or any other device of their choice.
Fit Criterion: The product shall automatically adjust to the size of the user's screen 95% of times.

3.3 Performance Requirements

3.3.1 Speed and Latency Requirements

NFR-11 The product shall initialize and setup swiftly.
Rationale: The product should load up fast to avoid users from getting bored or distracted.
Fit Criterion: The product shall take no more than 10 seconds to initialize and setup.

NFR-12 The product shall have a maximum loading time of 5 seconds between any of the pages.

Rationale: The user should not have to wait long times to navigate between pages.
Fit Criterion: The product shall take less than 1 second 95% of times, and less than 5 seconds rest of the times to load different pages.

NFR-13 The product shall be fast enough to keep the user from losing interest.
Rationale: It is important for the product to be quick and responsive to avoid losing the user's interest.
Fit Criterion: A survey of a sample of users should indicate that at least 90% users found the product quick and responsive.

3.3.2 Safety-Critical Requirements

N/A

3.3.3 Precision or Accuracy Requirements

NFR-14 The product shall display prices accurate to 2 decimal places.
Rationale: The product may lose users' trust and not be taken seriously if the prices shown are not precise and accurate.
Fit Criterion: The displayed values shall be accurate to two decimal places 100% of the times.

3.3.4 Reliability and Availability Requirements

NFR-15 The product shall be available 24 hours per day.
Rationale: The product will lose to competitors if not available 24 hours a day.
Fit Criterion: The product shall achieve availability of 24 hours for at least 85% of days in a year.

NFR-16 The product shall meet or exceed 98% uptime.
Rationale: A product with long downtimes will not be taken seriously.
Fit Criterion: The product shall not be down for more than 2% of times.

3.3.5 Robustness or Fault-Tolerance Requirements

NFR-17 The product shall log any errors and continue to operate as usual.
Rationale: The errors should be logged to get notified of any potential bugs that might affect the product. The product should not crash even in case of errors or else users may lose trust.
Fit Criterion: A survey of users shall indicate that no more than 1% of users faced any crashes.

3.3.6 Capacity Requirements

NFR-18 The product shall accommodate at least 100 users using the product simultaneously.
Rationale: A user shall not have to wait if other users are using the product.
Fit Criterion: A sample group of 100 users shall be able to use the product at the same time with no failures or wait time.

3.3.7 Scalability or Extensibility Requirements

N/A

3.3.8 Longevity Requirements

NFR-19 The product shall be easy to update and maintain with low budget availability.
Rationale: A project that is easy to maintain will last a long time.
Fit Criterion: Internal surveys should indicate that majority of developers working on the product feel that the product is easy to maintain.

3.4 Operational and Environmental Requirements

3.4.1 Expected Physical Environment

N/A

3.4.2 Requirements for Interfacing with Adjacent Systems

NFR-20 The product shall interface with external databases and servers to fetch data.
Rationale: The product will not be able to provide up-to-date data if the data is not fetched from a database or a server on every load.
Fit Criterion: Users shall see up-to-date information at least 99% of times whenever they open the product.

3.4.3 Productization Requirements

NFR-21 The product shall be deployed to a web server and be accessible to the users via a secure URL.
Rationale: It is important for the product to be deployed on a server so that users can access it quickly and on the device of their choice.
Fit Criterion: A sample group of users shall be able to verify that they can access the product using the URL. A security firm shall certify that the URL is secure.

3.4.4 Release Requirements

NFR-22 The product shall be updated weekly to patch any bugs.
Rationale: To keep the users away from seeing faulty data or face unexpected errors, it is important to update the product frequently to patch any bugs.
Fit Criterion: The product shall be updated weekly for at least 95% of weeks when there is a known bug.

NFR-23 The product shall be updated monthly to add/improve features.
Rationale: To ensure that users don't get bored of the existing functionality, the product shall be updated frequently to add new features.
Fit Criterion: The product shall be updated monthly with new features for at least 60% of months.

3.5 Maintainability and Support Requirements

3.5.1 Maintenance Requirements

NFR-24 The product shall be able to update frequently for bug fixes.
Rationale: Frequent updates and patches will help the product achieve more trust and credibility from users in the long term.
Fit Criterion: An internal survey shall indicate that at least 80% of developers feel that the product is easily updatable.

3.5.2 Supportability Requirements

N/A

3.5.3 Adaptability Requirements

NFR-25 The product shall run on mobile and desktop web browsers.
Rationale: Users that only have one type of device or the other should not be left out.
Fit Criterion: The product shall adapt to the user's screen size 95% of times.

3.6 Security Requirements

3.6.1 Access Requirements

NFR-26 The product shall be accessible to all users worldwide.
Rationale: To gain a wider range of users, it is important for the product to be accessible worldwide.
Fit Criterion: Users from around the world shall be able to access the product at least 99% of times.

NFR-27 Only the developers shall have access to logs related to bug and crash reports.
Rationale: The user should be able to focus on viewing cryptocurrency prices and should not have to worry about bugs.
Fit Criterion: A security firm shall certify that only the developers have access to bug and crash reports.

3.6.2 Integrity Requirements

NFR-28 Users shall not be able to alter data in the databases.
Rationale: It can be potentially risky if users are given write access to the data stored in the databases.
Fit Criterion: A security firm shall certify that users are unable to alter data in the databases.

3.6.3 Privacy Requirements

NFR-29 The product shall not collect any data from the users.
Rationale: Users shall not have to worry about their personal data being collected.
Fit Criterion: A security firm shall certify that no hidden trackers are present to collect personal data.

3.6.4 Audit Requirements

N/A

3.6.5 Immunity Requirements

N/A

3.7 Cultural Requirements

3.7.1 Cultural Requirements

N/A

3.7.2 Political Requirements

N/A

3.8 Legal Requirements

3.8.1 Compliance Requirements

NFR-30 The product shall comply with all government policies related to cryptocurrencies.

Rationale: The developers may face fines if the product doesn't comply with government laws and policies.
Fit Criterion: A legal team shall certify that the product complies with the appropriate laws and policies.

3.8.2 Standards Requirements

N/A

3.9 Health and Safety Requirements

N/A

4 Project Issues

4.1 Open Issues

Modifying the user interface such that it runs more efficiently and cleanly. The newer version of the project would be using NextJS instead of React so as to have server-side rendered pages.

4.2 Off-the-Shelf Solutions

There are few other available products like TradingView, Yahoo Finance, CoinMarketCap etc that are complicated and more geared towards professionals/experienced traders. Whereas our product is more simple and focused on students and entry-level investors.

4.3 New Problems

- Translating some old code and structure to fit in with the NextJS framework.

- Implementing some additional features such as graphs, comparing cryptocurrencies etc to the UI.

4.4 Tasks

The tasks for this project are set by the deliverable schedule in the course outline for Software Engineering 3XA3. These tasks include:

- The Problem Statement
- Development Plan
- Requirements Document
- Proof of Concept Demonstration
- Test Plan
- Design Documents
- Revision 0 Demonstration
- Final Demonstration
- Test Report

4.5 Migration to the New Product

- Converting the tech stack to NextJS
- Adding new Graphs and other visual features to the UI
- Feature to compare cryptocurrencies
- Ability to filter cryptocurrencies
- More modularity in the codebase

4.6 Risks

- Delay in updating the prices of cryptocurrencies.
- Not having enough data to have better accuracy on the graphs.
- The APIs used by the project might cease to exist.
- Completing the project within the given time frame (course deadline). This could risk the quality of deliverables and the final product.

4.7 Costs

Since the project is using all free-to-use technologies and services, there are no costs associated with the project unless we take into account the development time of the project by the developers.

4.8 User Documentation and Training

The documentation of different modules and files would be provided by the developers which will include its operation, inputs/outputs and purpose of the function. The user of the final product would require no such special training or documentation to get started. The UI should be relatively easy for anyone to navigate without any confusion.

4.9 Waiting Room

N/A

4.10 Ideas for Solutions

- Using free online resources for getting started with the tech stack: NextJS, JavaScript

- The user would be able to view the price chart of the cryptocurrencies to get a better understanding.

- Modular components of the project to add new features and to test the functions.

5   Appendix

The project's updated Gantt chart is located here.

5.1   Symbolic Parameters

Symbol	Definition
c_name	The variable name of the desired cryptocurrency to be utilized.

Table 4: Description of Symbolic Parameters

References

[1] David L. Parnas and P.C. Clements. A rational design process: How and why to fake it. *IEEE Transactions on Software Engineering*, 12(2):251–257, February 1986.

[2] James Robertson and Suzanne Robertson. *Volere Requirements Specification Template*. Atlantic Systems Guild Limited, 16 edition, 2012.