PPIT Application

Akeem Jokosenumi

Supervisor: Daniel Cregg

Github: [keemo01/PPIT (github.com)](https://github.com/keemo01/PPIT)

Contents

Submission link..........................................................................................................................3

GitHub: ..................................................................................................................................3

Chapter 1....................................................................................................................................3

Introduction............................................................................................................................3

1.1 Module Information: ...................................................................................................3

1.2 Objective .....................................................................................................................3

1.3 Brief.............................................................................................................................3

Chapter 2: Tech we used............................................................................................................5

2.1 Framework .......................................................................................................................5

2.2 Database Connection .......................................................................................................6

2.3 API’S................................................................................................................................6

2.4 Web/ Mobile ....................................................................................................................6

Chapter 3: System Architecture .................................................................................................7

....................................................................................................................................................9

Chapter 4: Design Methodology..............................................................................................11

4.1 User Interface.................................................................................................................11

4.2 Ease of use .....................................................................................................................11

4.3 Clients............................................................................................................................11

4.4 Maintenance...................................................................................................................12

Chapter 4: Software Development Life Cycle:........................................................................12

Chapter 5 Testing Plans: ..........................................................................................................12

5.1 Platforms........................................................................................................................12

5.2 Validation.......................................................................................................................13

5.3 Security ..........................................................................................................................13

Chapter 6: Planned Future Development.................................................................................13

6.1 Deploy to iOS/Android..................................................................................................13

6.2 Expansion.......................................................................................................................13

6.3 User Feedback................................................................................................................13

Chapter 7: Learning outcomes/ Conclusion.............................................................................14

7.1: React and Next.js..........................................................................................................14

7.2: Conclusion.........................................................14

Submission Link

Github: [keemo01/PPIT (github.com)](https://github.com/keemo01/PPIT)

*Chapter 1*

*Introduction*

Thank you for visiting my Design document for the Professional Practice in IT module. We were each assigned a team member for this module, and we were free to choose the type of project we wanted to complete based on what we had learned in the previous semester's modules. For our project, I made the choice to use the React native framework for my cryptocurrency price tracker.

I got the chance to create a software project for my senior project using the technologies we had studied over the previous three academic years. I made the decision to create a multiuser crypto (Cryptocurrency is any type of money that exists digitally or virtually and employs cryptography to safeguard transactions.

Cryptocurrencies use a decentralized mechanism to track transactions and create new units rather than a central authority to issue or regulate them. Users can log in, register, and update their portfolios on a website that tracks real-time coin prices. The ability to create, read, update, and delete their holdings is included in this.

Over this document you will see how I Implemented this into my project and areas I faced difficulties in.

* 1. Module Information:

• Module: Professional Practice in IT

• Due Date: *29th/04/2022*

• Team: Akeem Jokosenumi

• Version Control Git-hub

• Documentation: Word

1.2 Objective

The objective of this project is to create a dynamic e-commerce site in which a user can go to the app see a list of currencies then when clicked on it will prompt how the coin is doing in the last year also display information on the coin. We wanted to create an application that could be launched into the real world and produce a service to a business.

* 1. Brief

Many of the best practices that we learned about in previous semesters are supposed to be collected in this module. It offers the students the chance to create, refine, and implement a project either on their own or in a group setting. It offers us the opportunity to produce software in a timely and standardized manner. This is the first module we've been given with this much independence during my time at ATU. I hope to provide this project in an appropriate manner like in a company environment like in regards with work progress, version control of code using Git-hub, Documentation, Communication via MS Teams/Meetings throughout the module. Below is a table listing each component and its owners.

SYSTEM REQUIRMENTS

* UP TO DATE PC
* Internet Access
* NodeJS
* React
* Next.js
* MongoDB
* Visual Studio Code
* Google Chrome
* Command Prompt
* Windows 10

**The requirements for this app are as follows:**

• Client-side routing

• Website must have a minimum of four components

• The server must be able to read from the database

• The server must be able to write data to the database

• The server must be able to delete and update data in the database

• User must be able to view their portfolio.

• User Must be able to Update/Delete/Add/Update data from their portfolio.

• User Must be able to view live cryptocurrency prices.

Project Overview

Crypto Home Page: Akeem

Api: Akeem

Webpage Layout: Akeem

Chapter 2: Tech we used

2.1 Framework

We decided to continue with REACT after holding a few meetings and considering alternative systems like java net bean. A JavaScript framework called React is used to create natively rendered, real-world mobile applications. Making interactive web pages is possible using the client-side and server-side programming language JavaScript. JavaScript transforms a static web page into an interactive one, improving the user experience. We looked for REACT frameworks for this application and discovered Next.js, a React framework that provides building blocks to develop online applications. Framework translates to the tooling and setup required for React are handled by Next.js, which also offers extra structure, capabilities, and optimizations for our application. One of the biggest features that we liked was server-side rendering, this feature showed that it increased applications performances

2.2 API’S

A collection of procedures, protocols, and instruments used to create software applications is known as an application program interface (API). The interaction between software components is essentially described by an API.

Programming graphical user interface (GUI) components also makes use of APIs.

A Restful API was one of the APIs we studied and used in our project. A REST API, commonly referred to as a RESTful API, is a web API that complies with the restrictions of the REST architectural style and enables communication with RESTful web services.

Computer scientist Roy Fielding came up with the acronym REST, which stands for representational state transfer.

2.3 Web/Mobile

The platform flexibility of React and next.js is one of their biggest advantages.

Using a free hosting service like GitHub, I also intend to host this program as a website. I was able to build both mobile and online measures using Next.js and give compatibility with screen measurements for mobile devices. Providing a mobile/web version of this program in an effort to expand its potential user base.

Chapter 3: Design Methodology

3.1 User Interface

We made an effort to adhere to current standards and conventions for the UI. Numerous websites and programs use a grid style. Information is arranged into grids in grid layouts, making it simple to peruse and allowing individuals to pause at certain areas of interest. Grid layouts enable an even distribution of text, images, and video on websites, allowing users to specify the relative significance of each component.

3.2 Ease of use

We always intended for our app to be simple to use and to provide little on-screen instructions to users unless they were entering data. Additionally, we kept in mind that just because something is simple to use on a website doesn't necessarily guarantee it will work well on a mobile device.

3.3 Clients

Keep user inputs to a minimal was the consistent response when we asked customers what they would expect from a website when buying a product. This meant that the user just needed to input the most important data and minimal navigational inputs. Don't overwhelm them with tabs and pop-up windows full of information about your items and orders. which can cause them to get irritated.

Chapter 4: Software Development Life Cycle:

I would say that our approach to the Software Development life cycle was similar to the waterfall paradigm. There is a linear sequential flow in the waterfall model. In which development is viewed as flowing continuously through the stages of software implementation (like a waterfall). This implies that the start of every phase in the development process depends on the conclusion of the phase before it.

***Project Management:***

• **Project Initiation:** This is where we came up with ideas and what sort of project to

pursue.

• **Project Planning:** Once we knew what project to do, we started doing research into

what we need to complete it.

• **Project Execution:** This was where we started executing and building deliverables to

complete assigned tasks.

• **Project Monitoring and control:** This was where we would try and prevent scope

creep and keep up to date with our assigned tasks.

• P**roject Closure:** Is where we delivered all important deliverables and were able to

close the project.

Chapter 5 Testing Plans:

5.1 Platforms

We plan to have this application on mobile devices and as a web app. This requires us to test all the features and have the sizing right for both platforms.

5.2 Security

If there are several security breaches, security is a crucial issue that could spell the end for your application. One security check we should perform is to make sure there are no loopholes through which a user could access our orders database via the website and the administrative portion of the app.

Chapter 6: Planned Future Development

6.1 Deploy to iOS/Android

After fully completing the app, we plan on deploying the app onto the mobile market. Android has the biggest share of the market.

6.2 Expansion

If our website expands, we might need to stop using an online service and instead utilize our own database, which is kept on a different system. To further improve the user experience, we can also add new features.

6.3 User Feedback

The best apps may be made with the support of user feedback, thus I would aim to collect more of it in the future.

Chapter 7: Learning outcomes / Conclusion

7.1 React and Next.js

I appreciated how adaptable and dependable the Next.js framework and React were. The documentation was excellent and made it easy for me to integrate various popular features. Their error handling mechanism is one area where I believe they could make improvements. Compared to Java's IDE eclipse, React Errors aren't particularly good at describing what the problem is. It's nice that the app responds quickly and can support numerous platforms at once thanks to server-side rendering.

7.2 Conclusion

Overall, I was pleased with the website I made and the obstacles I faced learning this new framework like react and a programming language like JavaScript which I don’t have the most experience with was challenging I read the documents and used many videos I seen online on YouTube to help me make the project.

I enjoyed using JavaScript I think it’s one of my strong and favourable coding languages its simple & easy to use compared to C++, its high level and can easily be embedded with HTML.

References:

<https://www.binance.com/en>

https://www.coinbase.com/