Web Application for Cryptocurrencies

System specification

Utility of the application

The theme that we have selected our project is cryptocurrency, because cryptocurrency is a digital currency, that through cryptography ensures transactions and control over the creation of new currency units. One of the key features of cryptocurrency is the creation of the new digital coins so they can be predictable and produced in an even pace. There are around 2000 different cryptocurrencies and price of each unit varies from currency to currency. Bitcoin is the most widely used cryptocurrency, however, it is still at development stage, but it is becoming more and more popular.
The development of this application includes features such as:

- A useful tool for checking prices when they would like to buy and sale cryptocurrencies.
- The application could keep users on the right track and by using simple and convenient structure, users would be able to avoid spending time on countless websites, as we have gathered multiple currencies at one platform.
- The application might help users save time by having a simple design and could provide the most relevant information regarding the cryptocurrencies of interest.

Information Processing

The information is taken from live '*application programming interface*', also known as API, and have presented the most important information associated with five different cryptocurrencies we have selected. As information includes market capitalization, supplies and price change within different time periods. This data gives users a simple and straightforward overview about the information they need to know when trading currencies.

System Architecture

The sense of this project is to accomplish some intended goals regarding the live prices of cryptocurrencies. We have used API data to present prices of cryptocurrencies. Cryptocurrencies are digital currencies in which cryptography techniques are used to regulate the creation of unit currency, verify transactions, and operating independently of a central bank.

The information depends on getting the API data from the website of cryptocurrency services. The information is processed within an application mainly built by the golang programing language.
Our application gets the input by clicking on different buttons on the main page. As a result, output is presented on website. Since there are several types of information, we divide the cryptocurrency data by providing different paths or webpages and each page refers to a certain result.

This project can work wherever we want in case we have access to the internet. It is designed mainly for those who are interested in cryptocurrencies. The project basically gives an overview over the prices of the cryptocurrencies on internet.

By clicking on the name of the cryptocurrency after navigating on the website, we get back its price and other information. Thus, the input is already available in the project, but we activate it by clicking on the button to get the result “prices of the cryptocurrencies” that can also be considered as “output”.

The project can be used through all kinds of computer systems with a browser. It does not demand certain specifications, but as mentioned above, the project cannot work without internet because of getting “live data”.

The top-level software components in the project could be summarized as follow:

Input

The project is built with different kinds of programming languages. Golang programming language is the main one that works as the backend. It builds up a web server, stores API data into RAM, and send it to different website templates. Our major work is based on golang programming, but we have used other languages to add extra features and to design our project smoothly.
Moreover, the external formatting of the project is also built by two programming languages HTML and CSS. The web server provided by golang runs on http://localhost port 8080.

Output / Frontend

Three programming languages (HTML, javascript and CSS) are used in this project to show API data in a pleasant webpage. We also differentiate HTML templates representation from the simple data.

Prediction/Suggestion feature

As per requirement of our project, we have added prediction statements that provide suggestion. The suggestions are based on the market prices and changes in percentage during the last 24 hours.

Test and error handling

Testing and error handling are the most important elements in programming. Therefore, we have considered it seriously and applied these two core elements in our project. The main purpose of testing is to ensure that the whole project is working well and to fix errors if there is need to that. On the other side, through implementation of error handling, to fix errors during each stage. So we could catch them on the right time and give proper reasons why this has happened. Software must be able to detect errors and recover from them. Since bugs cannot be fixed online, so the developers must do that off-line as a part of maintenance.

Conclusion

Our project is to create an application that retrieves live data, so we have selected cryptocurrency as a theme. Since cryptocurrencies becomes more popular, the application will be a helpful tool for the users within trading cryptos. This tool can save time because it collects the useful details and focuses on the most popular cryptocurrencies. Thus, the users avoid spending time on multiple websites since the application focuses only on important information as users may find unnecessary information on other webpages. We have created a simple design that is easy and fast to navigate , this could keep the users on the right track. We have learned how to implement requirements, get reasonable results and how to document, test and drive the program to be compatible to the desired requirements.

Glossary

Http = Hypertext transfer protocol
CSS = Cascading style sheet
HTML = Hypertext markup language
API = Application programming interface
RAM = Random-access memory
Front end = parts of the project a user interacts with--such as the graphical user interface or command line.
Back end = the parts that do the work, but the user is unaware of or cannot see. Databases, services, etc.

Resources

System Spesifikasjon: https://no.wikipedia.org/wiki/Kryptovaluta

https://www.lix.polytechnique.fr/~golden/systems_architecture.html

Info about testing golang codes

To test golang codes, one has to write “go test -v” in the program terminal or powershell. To avoid code duplication, we have only two test functions. One for “Collectdata” and other for “Template”. One does not need to test each function in the main file, as they are the same. If there is need to test other functions, one can change the “handler := http.HandlerFunc(here you can change the function name)” in the test file.