**YOUR** **LOGO**

**Blockchain Alerting Project Technical Documentation**

| **REVISION HISTORY** | | | |
| --- | --- | --- | --- |
| **DATE** | **VERSION** | **DESCRIPTION** | **AUTHOR** |
| 07/06/2022 | 0.1 | Initial template | Ismail Said |
|  | 0.2 | Blockchain Alerting Project | Samuel Itauma |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |
|  |  |  |  |

[INTRODUCTION 3](#_Toc154045699)

[Purpose 3](#_Toc154045700)

[Objectives 3](#_Toc154045701)

[Background 3](#_Toc154045702)

[GLOSSARY 3](#_Toc154045703)

[TECHNICAL ARCHITECTURE 4](#_Toc154045704)

[Illustration 4](#_Toc154045705)

[DATA SOURCES 5](#_Toc154045706)

[Moralis 5](#_Toc154045707)

[Coin Gecko 6](#_Toc154045708)

[Global Bitcoin Price Index 6](#_Toc154045709)

[CRYPTO news Api 7](#_Toc154045710)

[DATA INGESTION 8](#_Toc154045711)

[Python Script 8](#_Toc154045712)

[DATA STORAGE 8](#_Toc154045713)

[Microsoft Fabric 8](#_Toc154045714)

[Entity Relationship Diagram (ERD) 9](#_Toc154045715)

[Data Dictionary 9](#_Toc154045716)

[Blockchain Alerting 12](#_Toc154045717)

[Email Alerting 12](#_Toc154045718)

[Testing and User-Feedback 14](#_Toc154045719)

# INTRODUCTION

## Purpose

The purpose of this document is to give you a understanding into the technology behind Blockchain and how it can be used in many ways whether its integrating to move, copy or store DATA.

## Objectives

The objective of this project is to source blockchain data, monitor specific conditions, and send conditional alerts to an Email. The project aims to provide real-time insights and notifications based on changes or events in the blockchain data.

## Background

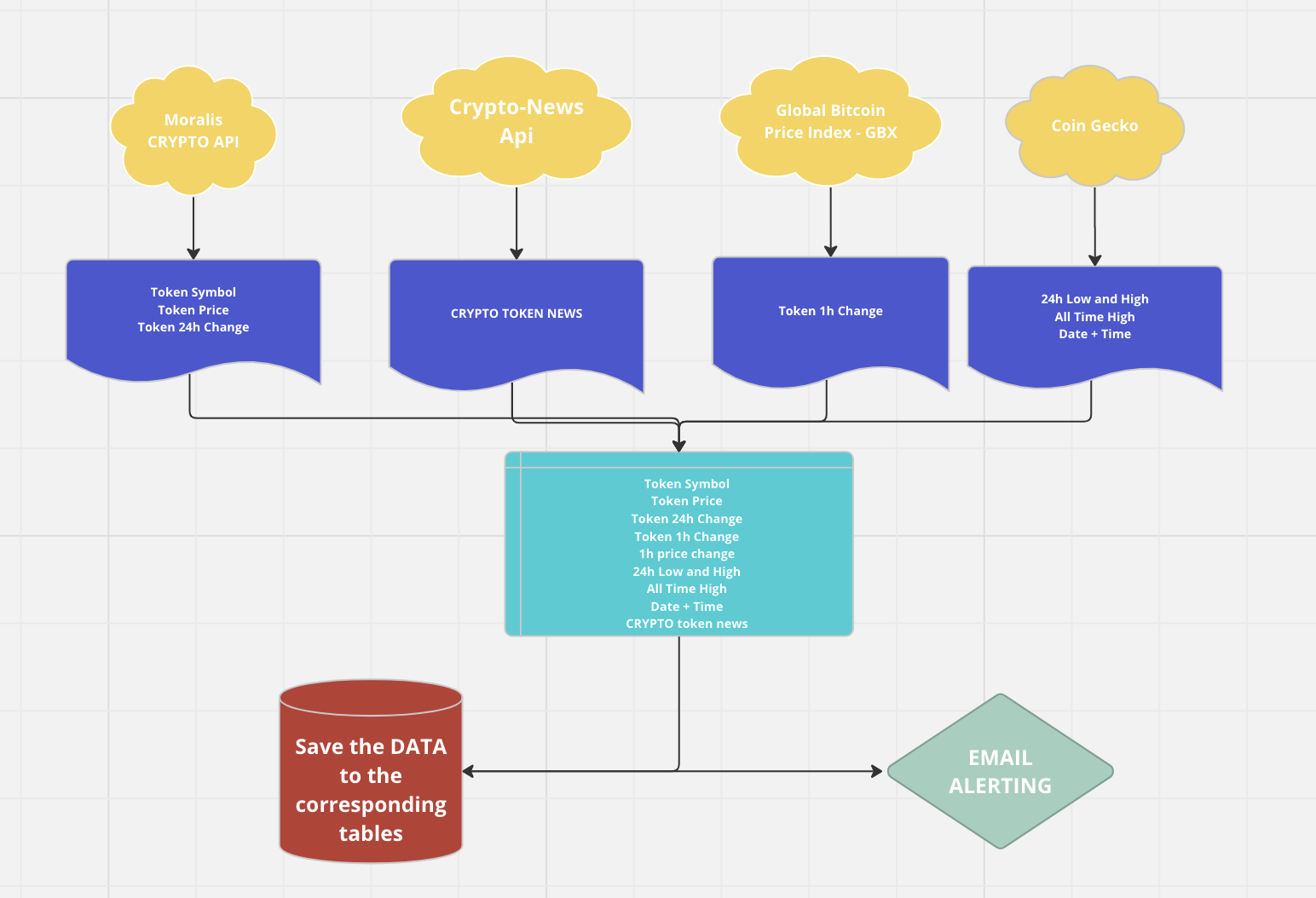
Just like what the name refers to ‘Block chain’ the structure of a Blockchain is an immutable chain of blocks containing data. Once a block is added it cannot be changed unless all succeeding blocks are changed aswell, that’s why a blockchain system commonly has multiple validators that conduct POW (Process of Work). Therefore, the decentralized nature of blockchain, with multiple validators reaching consensus (POW), ensures data integrity, transparency, and security.

# GLOSSARY

|  |  |
| --- | --- |
| **Term** | **Definition** |
| Application programming interface (API) | A way for two or more computer programs to communicate with each other. It is a type of software interface, offering a service to other pieces of software |
| Query | Query is a question or a request for information |
| DE-Centralization | Is essentially the process of shifting control from one main group to several smaller ones. (Explain...) |
| URL | A URL is the link which is associated with a site for example Youtube’s url is Youtube.com |
| ERD | It’s a graphical representation of the different ‘objects’ link together. |
| ENDPONT | Endpoint is one end of a communication channel. When an API interacts with another system, the touchpoints of this communication are considered endpoints. |
| SQL | Structured-Query-Language is a programming language which is used to ‘talk and transform’ your data |

# TECHNICAL ARCHITECTURE

## Illustration



# DATA SOURCES

Listed below are the data sources that have been shown in the technical architecture:

## Moralis

Moralis APIs powers crypto and blockchain applications for millions of end users worldwide. Furthermore, Moralis has industry-leading API response times, high reliability, clean, organized data.

Link: <https://moralis.io/>

**Authentication**

To authenticate your API you will simply just need to have a valid API key associated with your account which can be done through your Moralis account.

CODE SNIPPET:

**from moralis import evm\_api**

**api\_key = "YOUR API KEY"**

**params = {**

**"chain": "eth",**

**"include": "percent\_change",**

**"address": "0x7d1afa7b718fb893db30a3abc0cfc608aacfebb0"**

**}**

**result = evm\_api.token.get\_token\_price(**

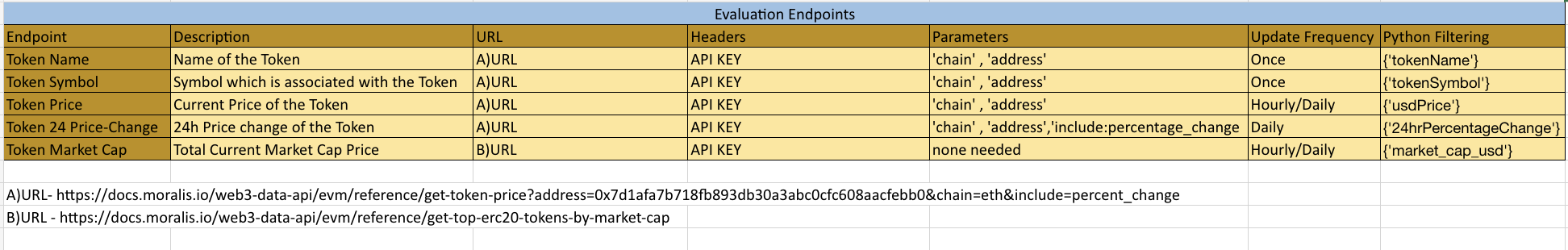
**api\_key=api\_key,**

**params=params,**

**)**

**print(result)**

**Endpoints**



## Coin Gecko

Coin Gecko allows you to power your applications with their own independently sourced crypto data such as live prices, NFT floor prices, trading volume, exchange volumes, trading pairs, historical data, contract address data, crypto categories, crypto derivatives, images and more.

Link: <https://www.coingecko.com/api/documentation>

**Authentication**

No API key is needed just a direct connection to their request url.

Code Snippet:

**import requests**

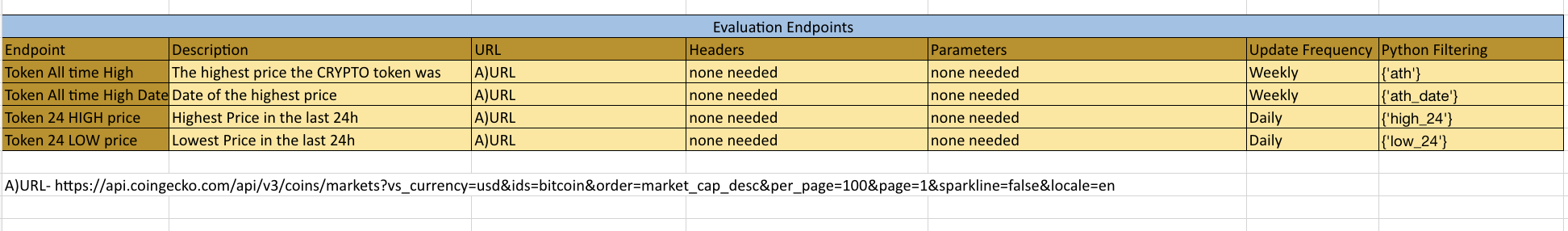
**url = '**[**https://api.coingecko.com/api/v3/coins/markets?vs\_currency=usd&ids=bitcoin&order=market\_cap\_desc&per\_page=100&page=1&sparkline=false&locale=en**](https://api.coingecko.com/api/v3/coins/markets?vs_currency=usd&ids=bitcoin&order=market_cap_desc&per_page=100&page=1&sparkline=false&locale=en)**'**

**response = requests.get(url)**

**jsonRESPONSE = response.json()**

**print(jsonRESPONSE)**

**Endpoints**



## Global Bitcoin Price Index

Global Bitcoin Price Index is the longest running Cryptocurrency price API provider.

These APIs can be used to gather real-time market data, volume and historical price data for many Cryptocurrencies.

Link: <https://rapidapi.com/blockchain-data-ltd-blockchain-data-ltd-default/api/global-bitcoin-price-index-gbx/>

**Authentication**

The authentication for this is a linked API key to a valid account. I did this through Rapid Api.

Code Snippet:

**import requests**

**url = "**[**https://bitcoinaverage-global-bitcoin-index-v1.p.rapidapi.com/indices/global/ticker/BTCUSD**](https://bitcoinaverage-global-bitcoin-index-v1.p.rapidapi.com/indices/global/ticker/BTCUSD)**"**

**headers = {**

**"X-RapidAPI-Key": "3fbeefc941mshde8c6a8d5c3a7b5p1628b8jsn1dfd97d6a49c",**

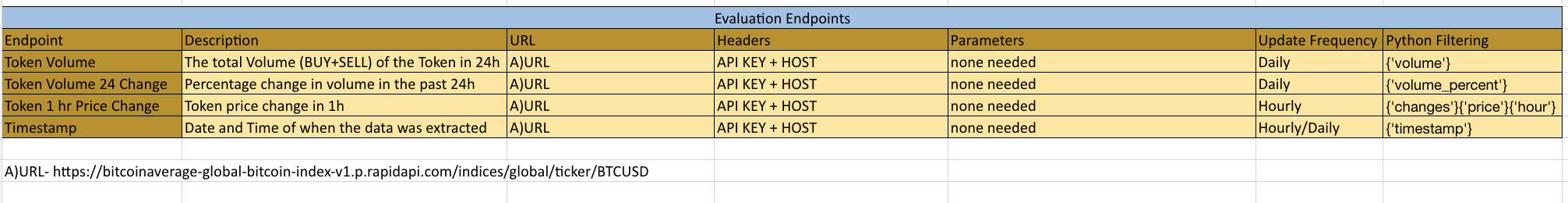
**"X-RapidAPI-Host": "bitcoinaverage-global-bitcoin-index-v1.p.rapidapi.com"**

**}**

**response = requests.get(url, headers=headers)**

**print(response.json())**

**Endpoints**



## CRYPTO news Api

CRYPT news API allows you to get the latest crypto news from the best news sources. Their API has access to clean and relevant crypto news data which can be easily accessed.

Link: <https://cryptonews-api.com/>

**Authentication**

The authentication is a linked API key to a valid account. This can be set up through their main page where you can select the type of subscription that fits your requirements.

**import requests**

**url = f'https://cryptonews-api.com/api/v1?tickers=BTC&items=3&page=1&token=g3qop9boib27sweon7jqevkelhgmfoeasmteqbzq'**

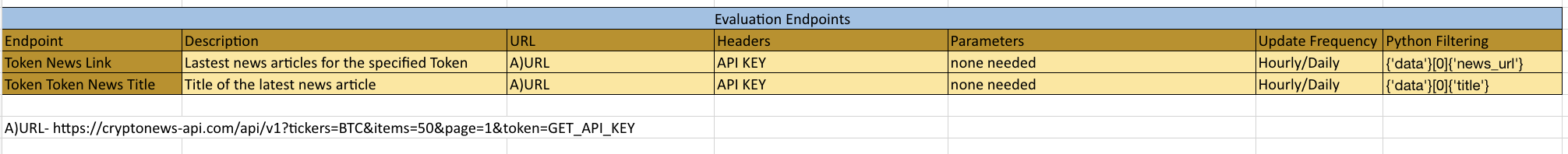
**response = requests.get(url)**

**JSONresponse = response.json()**

**News = JSONresponse['data'][0]['title']**

**title\_news\_link = JSONresponse['data'][0]['news\_url']**

**Endpoints**



# DATA INGESTION

Data is collected and transformed by using Microsoft Fabric Notebooks. Furthermore, Notebooks can be used for far more as this can perform data science tasks such as data cleaning and transformation, data visualization, statistical modelling, machine learning, and deep learning.

In addition, within the notebooks I'll be using a mixture of SQL, PYHTON AND DELTA coding which is possible thanks to the flexibility and integration of Microsoft Fabric.

## Python Script

All the DATA is stored within my Microsoft Fabric Workspace, however I will copy the contents to a GitHub repository. (Link to GitHub) Everything will be explained in the GitHub repository.

## 

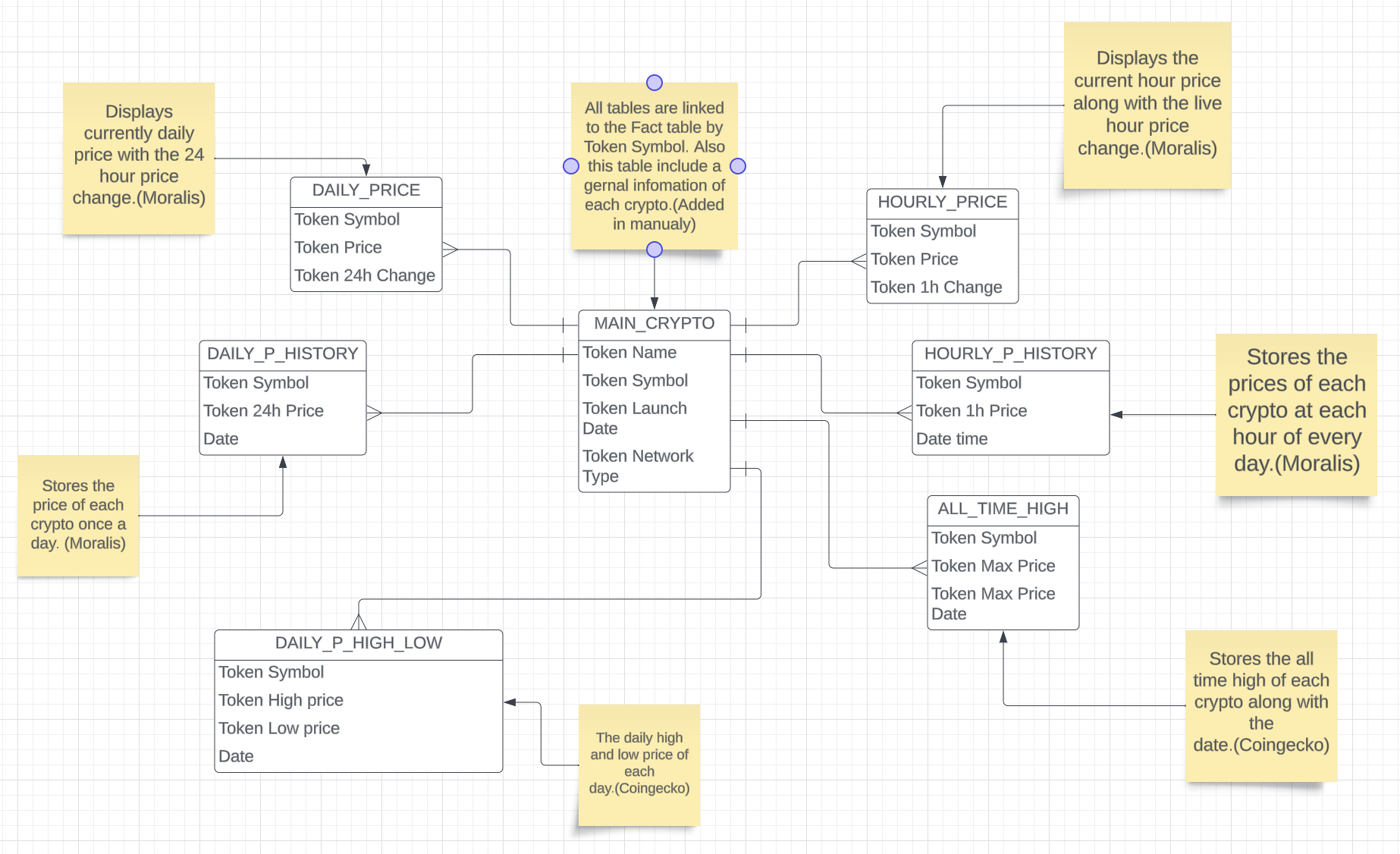
# DATA STORAGE

## Microsoft Fabric

In Microsoft fabric all operations are in harmony, you can instantly query or transform data right at its source. Essentially, it’s an all-in-one analytics solution for enterprises that covers everything from data movement to data science, Real-Time Analytics, and business intelligence.

With Fabric, you don't need to connect different services from multiple platforms. Instead, you can enjoy a highly integrated, end-to-end, and easy-to-use product that is designed to simplify your analytics needs.

## Entity Relationship Diagram (ERD)



## Data Dictionary

Information about the tables

|  |  |  |
| --- | --- | --- |
| **Table Name: MAIN\_CRYPTO** | | |
| **Table Description: This is the Fact Table where all the dimensions tables are linked to by. They are all linked by the Token Symbol** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Name | varchar | This includes the name of the token |
| Token Symbol | varchar | This includes the associated token symbol |
| Token Launch Date | date | The date which the token was launched |
| Token Network Type | varchar | The type of network the blockchain is ran on |

|  |  |  |
| --- | --- | --- |
| **Table Name: DAILY\_P\_HIGH\_LOW** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token High Price | int | This is the highest price the token was on that day |
| Token Low Price | int | This is the lowest price the token was on that day |
| Date | date | This is the date of when the data was extracted |

|  |  |  |
| --- | --- | --- |
| **Table Name: ALL\_TIME\_HIGH** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token Max Price | int | This is the highest price the token ever been in its history |
| Token Max Price Date | int | This is the date at which the token reached its highest price |

|  |  |  |
| --- | --- | --- |
| **Table Name: HOURLY\_P\_HISTORY** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token 1h Price | int | This is the price captured of a token at a specific hour of the day |
| Datetime | datetime | This is to help understand the date and the hour which the Token 1h price was captured |

|  |  |  |
| --- | --- | --- |
| **Table Name: DAILY\_P\_HISTORY** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token 24h Price | int | This includes the token price in a day when it was retrieved (if the code was run at 12:00 then the daily price will be the price at 12:00) |
| Date | date | This is the Date which the Token 24H Price was extracted |

|  |  |  |
| --- | --- | --- |
| **Table Name: DAILY\_PRICE** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token Price | int | This includes the token price in a day when it was retrieved (if the code was run at 12:00 then the daily price will be the price at 12:00) |
| Token 24h Change | int | This is the 24h price percentage change from the previous day |

|  |  |  |
| --- | --- | --- |
| **Table Name: HOURLY\_PRICE** | | |
| **Table Description:** | | |
| **Field Name** | **Data Type** | **Description** |
| Token Symbol | varchar | This includes the associated token symbol |
| Token Price | int | This is the price of the token for the hour |
| Token 1h Change | int | This is the 1h price percentage change from the previous hour |

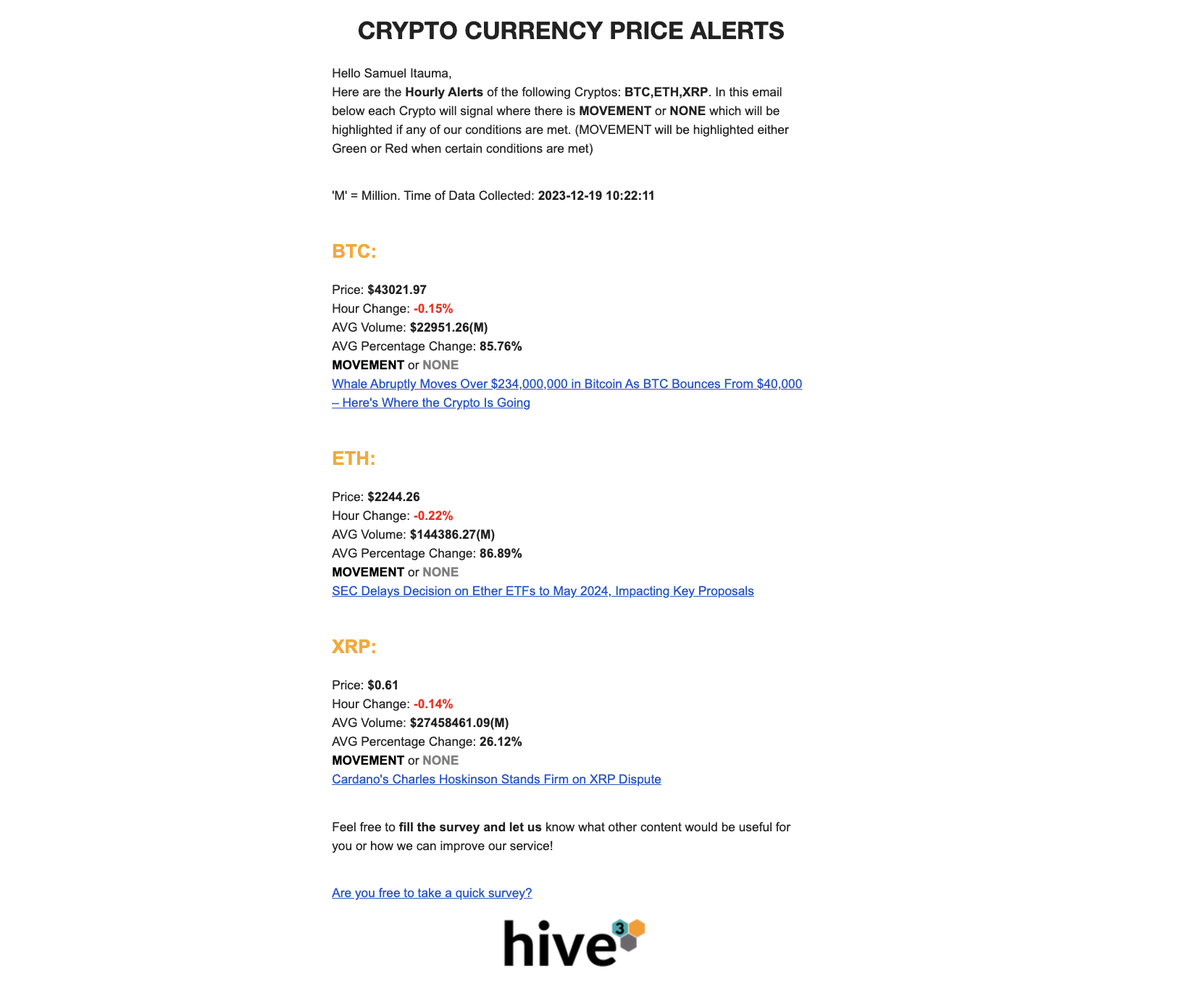
# Blockchain Alerting

## Email Alerting

Regarding the Email Alerting system; I have created two different variations:

1. 3 Cryptos of any Particular choice
2. Top 3 Gainers within the Crypto industry.

Beginning with No.1 within this email the recipient is told details about 3 specific crypto currencies. In this email the recipient is told about the: Price, Hour Change Percentage, Average Volume, Average Volume Percentage Change, Movement and Latest News. The key in this email is the **“MOVEMENT or NONE”** this is where the alerting takes place as I constructed certain targets where if they are all met then **“MOVEMENT”** with light up green signalling there is “MOVEMENT” happening which is unusual that you should have a look.



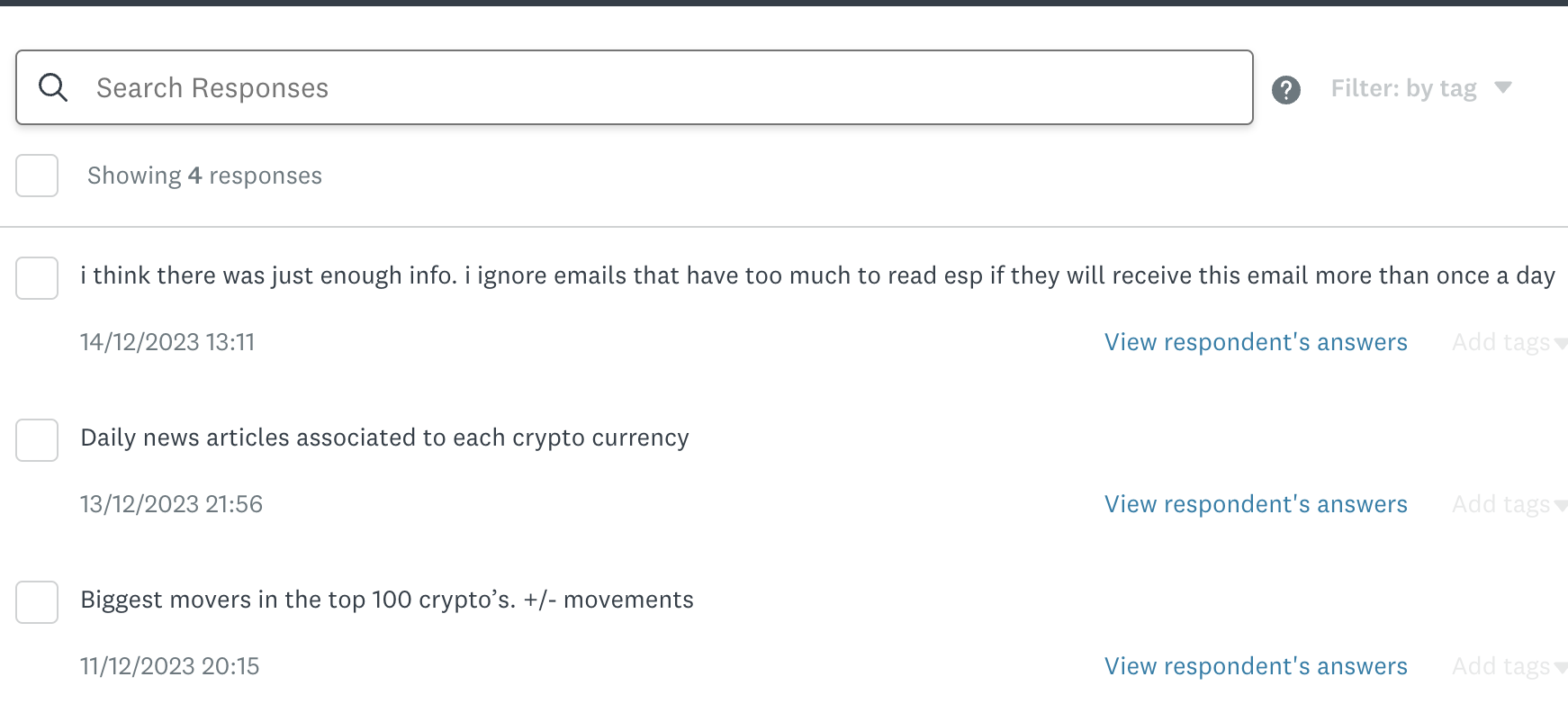
Moving on to No.2 within this email the recipient is alerted everyday about the **TOP 3 GAINERS** across the whole cryptocurrency market. Recipient can understand which Crypto coins are leading in the **“GAINERS”** aspect and with the help of a News Alert object in the email they will not just be alerted **which** coins are rising the most but also **why** if there are any associated news>



## Testing and User-Feedback

To test the effectiveness of the email I used “Survey Monkey’ (uk.surveymonkey.com) to compose me a survey which I attached to the end of every email I sent. Furthermore, within this survey I have 7 questions ranging from their rating to their “ways to improve”. One of the most common types of feedback was to create the email more Personalized and Eye-Appealing which I considered when creating the final piece.

Response 1:



Response2:

