# Final Presentation

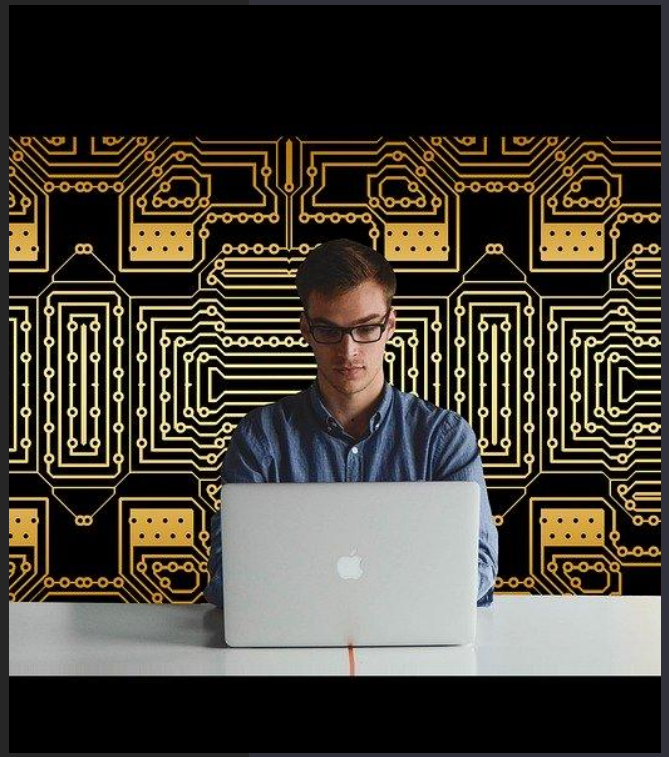
# Security Correlation Checking System

## for Stock & ETF

Group 10
Ela Yao, Shanrong Zhou, Sunny Deng,
Yilin Wang, Yuxin Liao

# Table of Content






# Background & Business Use Case

# Background

## Why is Correlation Important?

The correlation between the activity of two securities can be a very important factor in developing a prudent investing strategy.

- Sector analysis
- Diversification & Risk Management
- Trading strategies
- Portfolio optimization

# Project Scope

**Securities Trading Data:**

- Time Frame: Past 20 years
- Trading Interval: Daily
- Data Size: 2.5GB

**11,000+**

STOCKS

**US Stock Market**

**(NASDAQ, NYSE, AMEX, and OTC )**

**500**

TRADED COMPANIES

**S&P 500 ETF**

**30**

TRADED COMPANIES

**Dow Jones ETF**

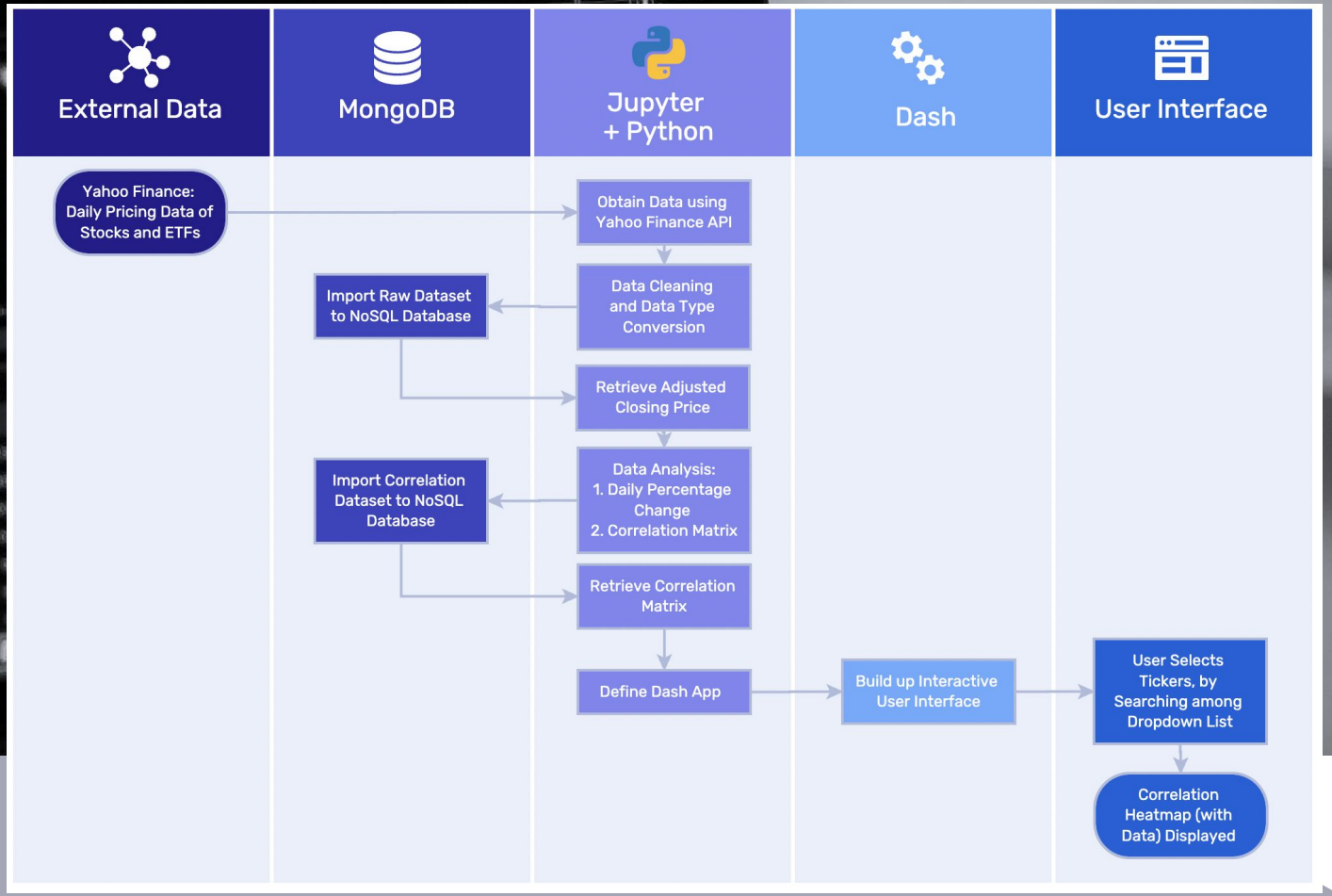
# Business Use Case & Purpose

This Project aims to check **correlations among ETF and individual stock prices** is to help investors gain insights into the relationships among these securities. By using Python to analyze historical price data, we can identify patterns and trends in the correlations among ETFs and individual stocks.

Final analysis result can help investors to better understand the risks and potential returns of their investment portfolio, and to make informed decisions about how to make better asset allocation and risk diversification


# Implementation Design & Technologies

# Implementation Design Flow



# Technical Viability and Applicability

Jupyter Notebook & Python

## API

We connect the Yahoo Finance API with Jupyter using the Python yfinance package, allowing for easy retrieval and integration of historical financial data, ensuring a security correlation checking system. APIs offer standardized data access and real-time data availability while being compatible with various programming languages and tools. Leveraging these advantages enables streamlined and effective data sourcing.

## MongoDB

MongoDB can be used to store and retrieve large volumes of historical price data for ETFs and individual stocks. Its flexible data model and native support for JSON data make it easy to integrate data from different sources and perform complex queries. MongoDB's scalability and high availability features make it well-suited for handling large volumes of financial data.

## Dash

Dash can be used to create custom dashboards and visualizations that allow investors to gain insights into the relationships among ETFs and individual stocks. Its support for real-time data updates and interactivity make it a powerful tool for monitoring financial markets and identifying trends

# Front End Technology

*Dash: Python Framework for UI*

- *Use **Dash** to create an interactive User Interface with a search engine allowing for fuzzy search*

01

02

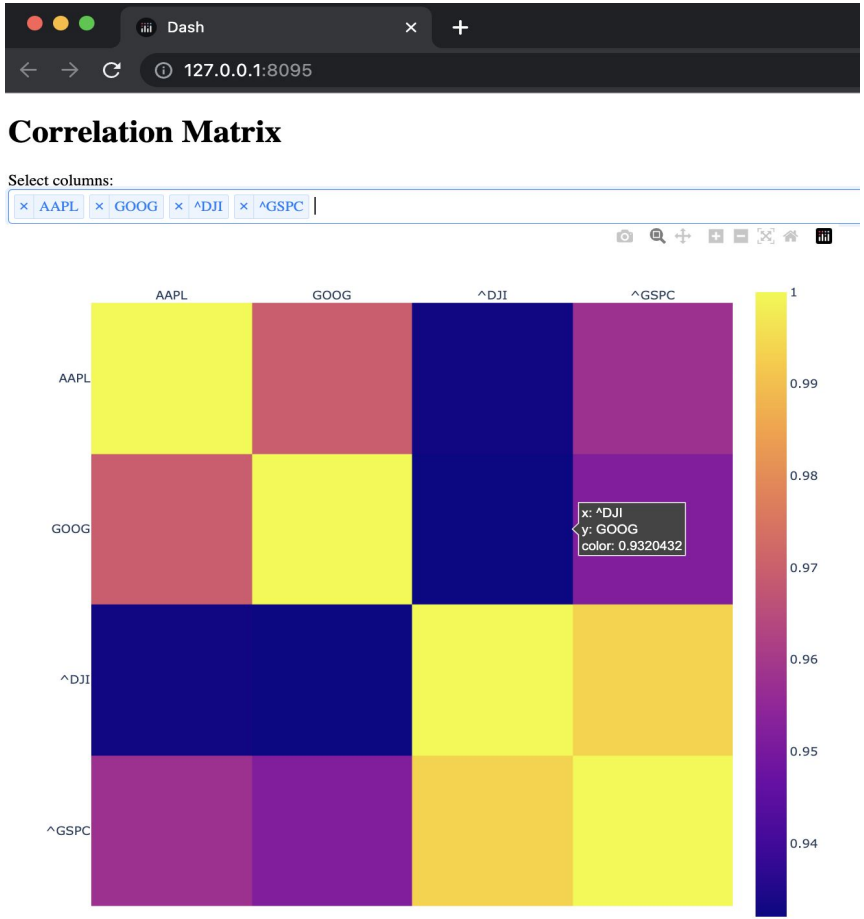
- *Generate results in **seconds**, no waiting time*

03

- *Showcase beautiful heatmap, readable and **artistic***

04

- ***Allow for downloading, zooming in and out, checking correlation number by moving cursor onto a specific area***



# Backend Technology

*MongoDB: High-Performance Database*

**Evolved Data Schemas**

Document database

**Replications**

Tolerance for Node Failure


**Sharding**

Support faster and more scalable applications

**Optimized Performance**

Map-Reduce Support

Therefore, it can handle **high volumes** of data at **high velocity** needs


# Data Quality, Cost Implications & Scalability

# Licensing & Cost Implications

***All of the resources used are OPEN SOURCES without any compliance issue***

### MONGODB

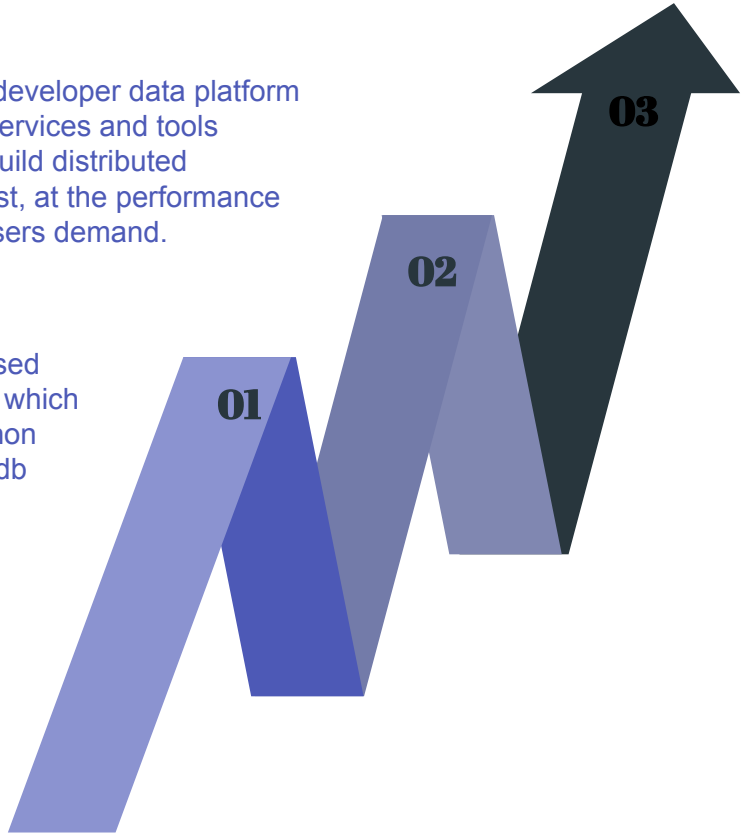
Mongodb is a developer data platform that provides services and tools necessary to build distributed applications fast, at the performance and at scale users demand.

### JUPYTER NOTEBOOK

Jupyter Notebook is a web-based interactive computing platform which we used to implement our python code and connect with Mongodb database.

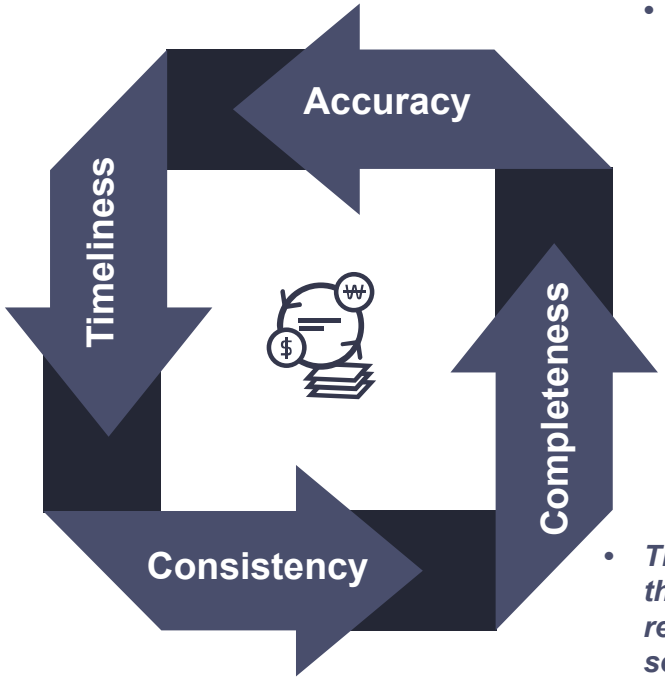
### YAHOO FINANCE

Yahoo Finance is a free Finance API Package containing large real-time stock databases which we used as the databases of our project



# Data Quality

- *The system must have the ability to collect financial data in real-time from the Yahoo Finance API, enabling it to identify any correlations and anomalies immediately.*




- *The system must ensure that the financial data obtained from Yahoo Finance API using Python yfinance packages is accurate, reliable and up-to-date.*

- *The financial data must be consistent across different sources and time periods, ensuring that the system can provide reliable insights.*

- *The system aims to gather all the relevant financial data required to perform effective security correlation checks, leveraging the full capabilities of the Yahoo Finance API.*

# Scalability

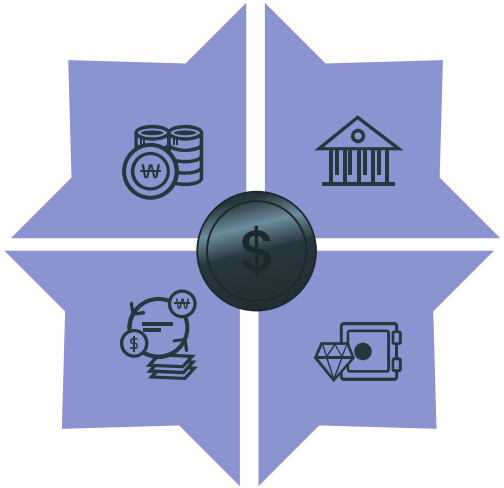
## Maintenance Plan

For the maintenance of the website, we plan to assign a website administrator to refresh our backend. Since the new data does not affect correlation number dramatically, a reasonable refreshing frequency will be weekly. Also, the administrator only needs to load newly added data from API, which largely shortens the maintenance time.

## Large Volumes of Data & Concurrent Requests

· Horizontal scaling: adding more servers to system to handle increased traffic or data volumes.

· Asynchronous processing: Async/await is a feature of Python that allows multiple tasks to be executed concurrently, without blocking the main thread of execution.



## Caching

Client-side caching involves storing data in the user's browser or device, such as in the browser's cache or in local storage.

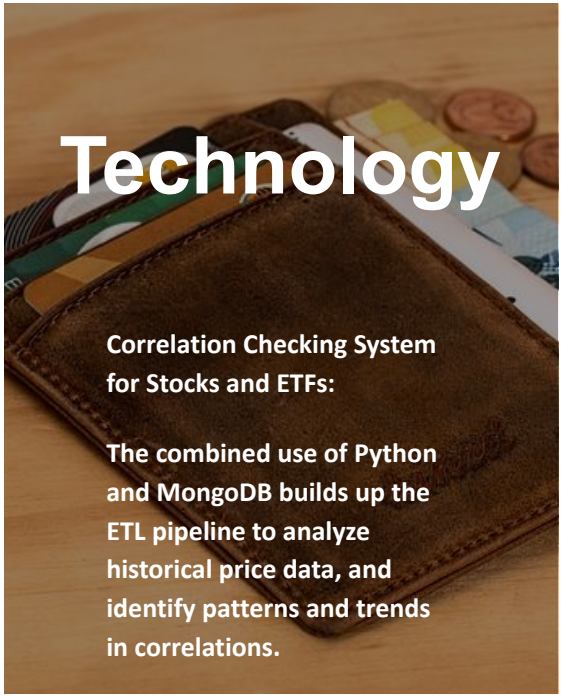
Server-side caching involves storing data in a cache server that sits between the application server and the database.

## Cloud Computing Services

The use of cloud computing services, such as Amazon Web Services (AWS) or Microsoft Azure, could provide extra scalability and elasticity to the system.
These services can automatically scale up or down based on demand, ensuring that the system can handle any workload.


# Conclusion & Recommendation

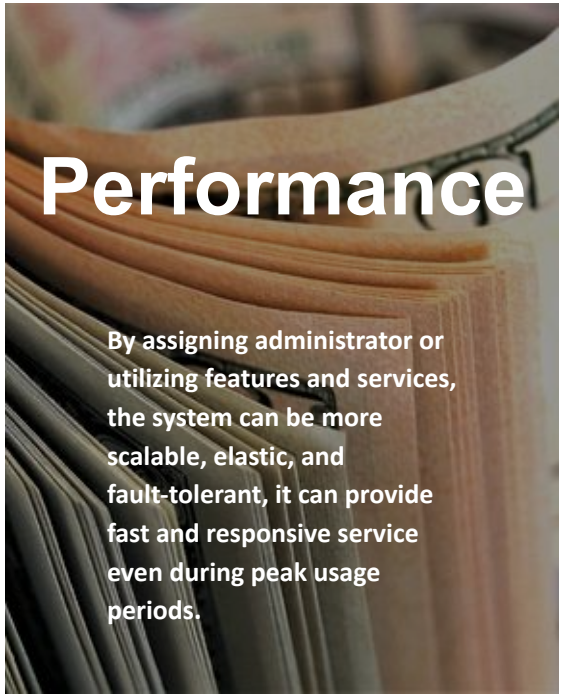
# Conclusion & Recommendation



## Technology

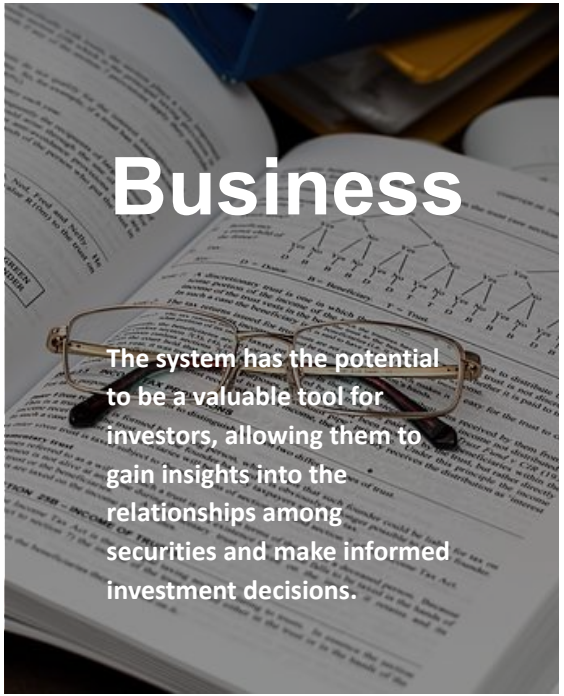
Correlation Checking System for Stocks and ETFs:

The combined use of Python and MongoDB builds up the ETL pipeline to analyze historical price data, and identify patterns and trends in correlations.



## Performance

By assigning administrator or utilizing features and services, the system can be more scalable, elastic, and fault-tolerant, it can provide fast and responsive service even during peak usage periods.



## Business

The system has the potential to be a valuable tool for investors, allowing them to gain insights into the relationships among securities and make informed investment decisions.


# User Interface Demonstration


Thank you