EZ Investment

Portfolio Management

Priya Sundararaj | Insiya Kanjee | Andriy Khalik | Scott Stimpson

INTRODUCTION

It has become more difficult for the average person to save their money. With rising inflation and low interest rates, investors are looking for alternatives.

Smart individuals increasingly want to take control of their investments and make decisions that best tailor to their goals and objectives.

With this project we want to enable users to take control of their investments and make informed financial decisions.

PROJECT SCOPE

1. **AUTOMATED PORTFOLIO OPTIMIZATION:**
   A PAGE WHERE USERS CAN INPUT THEIR PREFERENCES AND BE PRESENTED WITH AN OPTIMIZED PORTFOLIO

2. **MANUAL PORTFOLIO OPTIMIZATION:**
   A PAGE TO ENABLE USERS TO EXPLORE, MODIFY, AND CREATE THEIR OWN PORTFOLIO

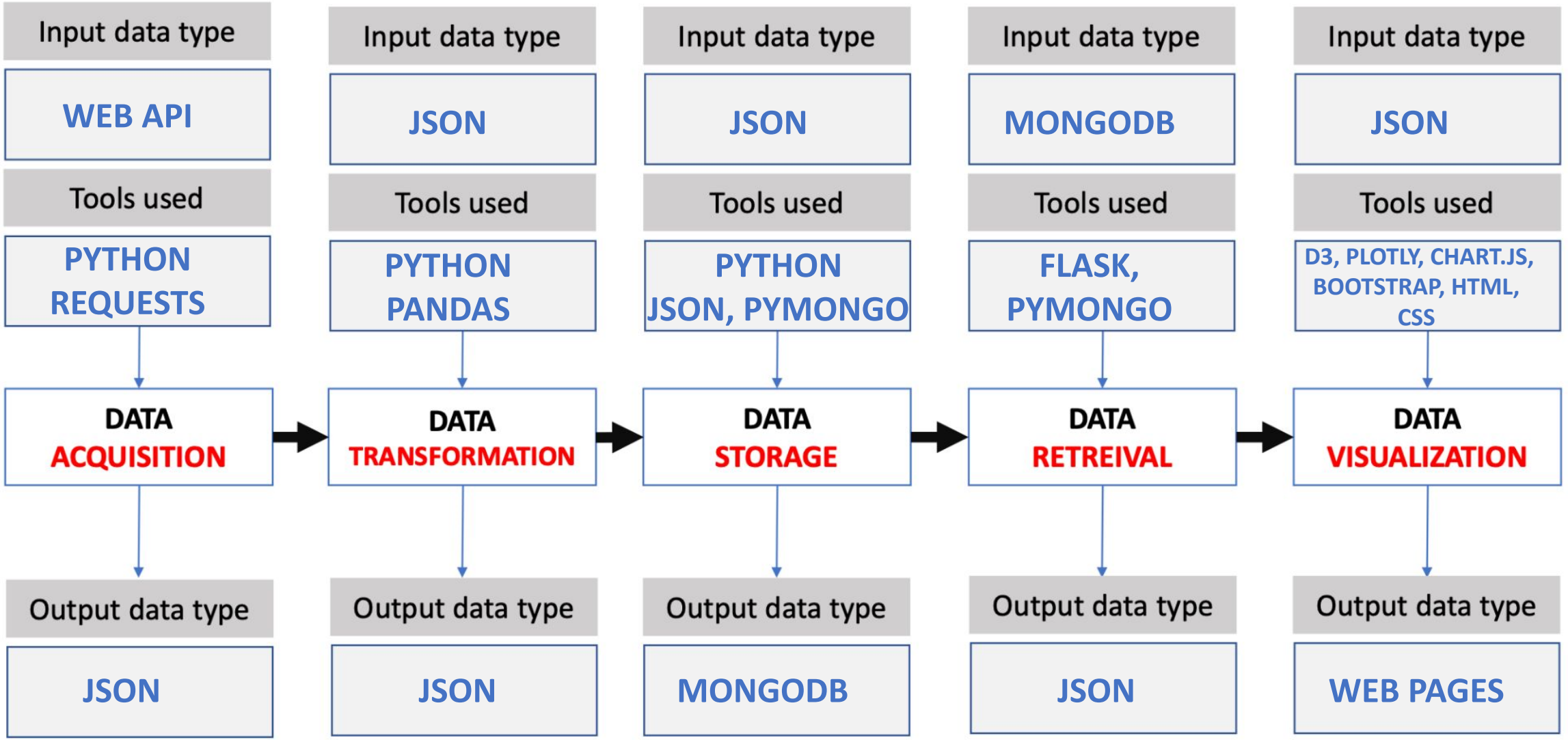
3. **LEARN MORE ABOUT EZ INVESTMENTS:**
   SERIES OF PAGES WHERE USERS CAN EXPLORE THE INDIVIDUAL STOCKS THAT THE EZ INVESTMENT PORTFOLIO OFFERS

AGENDA

1. Project Cycle Overview
2. Acquisition & Transformation
3. Storage & Retrieval
4. Visualizations
5. Live Demo
6. Limitations and Further Opportunities
7. Appendices

1. PROJECT CYCLE OVERVIEW

PROJECT CYCLE



2. ACQUISITION & TRANSFORMATION

DATA ACQUISITION

All of the stock information was acquired from Rapid API.

https://rapidapi.com/apidojo/api/yahoo-finance1/

DATA TRANSFORMATION

Data Cleaning

- Acquired JSON data was transformed using Python and Pandas
- Removed irrelevant fields
- Added date field with specific format converted from epoch time
- Rounded the closing stock prices to 2 decimal places
- Cleaned data was saved to a file

Calculations

- Various statistical functions were used to calculate Annual Volatility, Annual Return, Daily Volatility and Last 3 month Return for each of the stocks
- Daily returns and asset covariance was calculated and then portfolio return and volatility was computed

3. STORAGE & RETRIEVAL

STORAGE AND RETRIEVAL

- MongoDB and PyMongo library were used to store the clean json files.
- Each investment was saved as a separate collection. The stock computations were saved in respective collections.
- Flask and PyMongo were used to retrieve the data from MongoDB and a local web server was created with support for different API calls
- D3 javascript library was used to get data from Flask and populate the visual elements in HTML

4. VISUALIZATIONS

VISUALIZATIONS

The Website was created using these technologies;

We used a base template and built the pages based on the theme https://startbootstrap.com/theme/sb-admin-2

Chart.js is the new library we explored in this project.
https://www.chartjs.org/

Easy animation could be added to the charts to make them attractive.

5. LIVE DEMO

LIVE DEMONSTRATION

Let’s take a look!

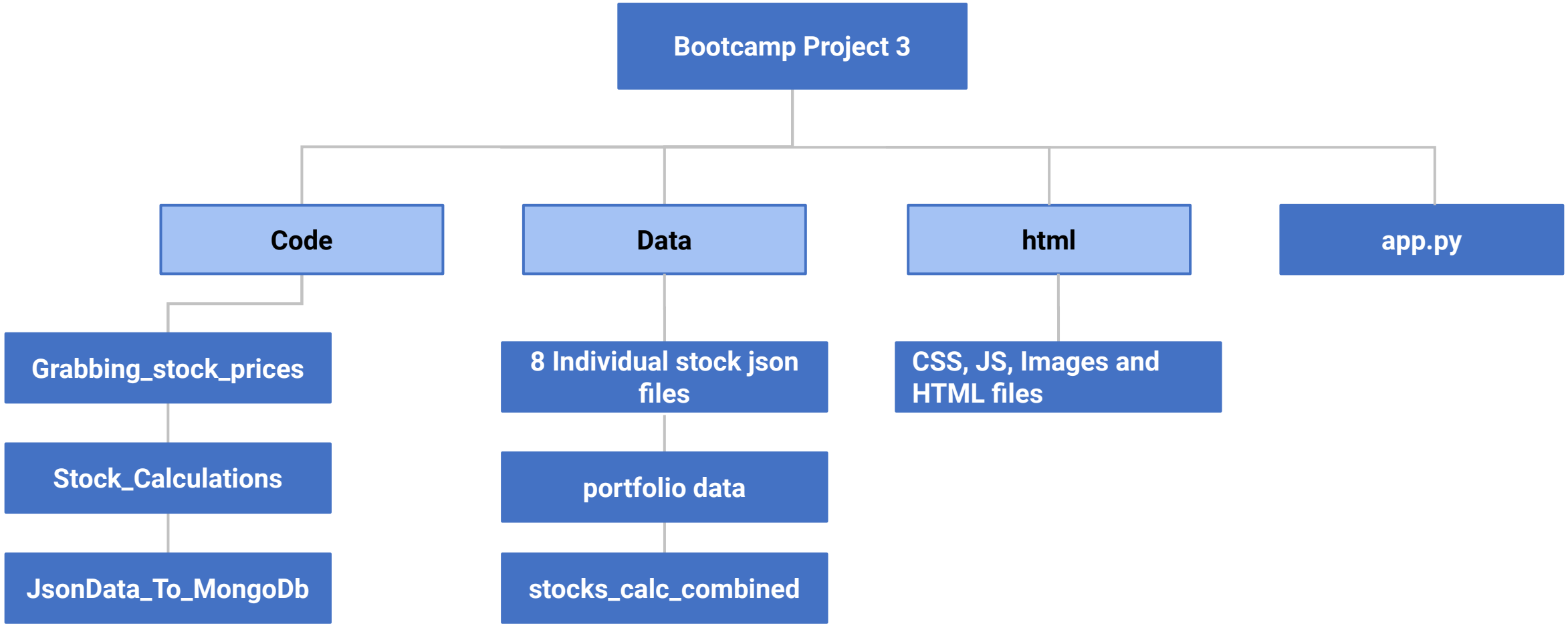
6. LIMITATIONS & FURTHER OPPORTUNITIES

LIMITATIONS & FURTHER OPPORTUNITIES

- There were only 8 investments chosen to study; Add other investments, stocks or exchanges.
- Historical Data was limited to the beginning of 2020; Additional data on investment performance could be added.
- Further mathematical improvement of model
- The subscription cost of Rapid API requests & Yahoo Finance API is no longer updated
- Measure the results from EZ Investments against historical trends

7. APPENDICES

GITHUB WORKFLOW



QUESTIONS?