

# CIS 550 Project Proposal

[https://github.com/ChengChen-Steven/CIS550\\_Final\\_Project/blob/main/document/Project\\_Proposal.md](https://github.com/ChengChen-Steven/CIS550_Final_Project/blob/main/document/Project_Proposal.md)

## Table of Contents

- 1. Motivation for the idea/description of the problem the application solves
- 2. List of features you will definitely implement in the application
- 3. List of features you might implement in the application, given enough time
- 4. List of pages the application will have and a 1-2 sentence description of each page
- 5. ER diagram
- 6. SQL DDL for creating the database
- 7. Explanation of how you will clean and pre-process the data
- 8. List of technologies you will use
- 9. Description of what each group member will be responsible for

## 1. Motivation for the idea/description of the problem the application solves

Nowadays we have seen plenty of stock trading apps and stock information sites in the market but they are getting more and more complex and heavyweight as we use them. We are trying to build a simple but useful site for people to look up and track the stock performance of the companies of their choice. We will have a clean and easy-to-read dashboard for people to customize their own portfolio/watchlist and obtain useful information for the stocks on the website.

## 2. List of features you will definitely implement in the application

- **Feature #1:** An overview of SP500 market performance by industry
- **Feature #2:** A side-to-side comparison for people to view a key metrics comparison between two (or more) stocks
- **Feature #3:** A stock filtering page to filter candidate stocks based on specified criterias

- **Feature #4:** A detailed page includes specific information for any specific stock in SP500

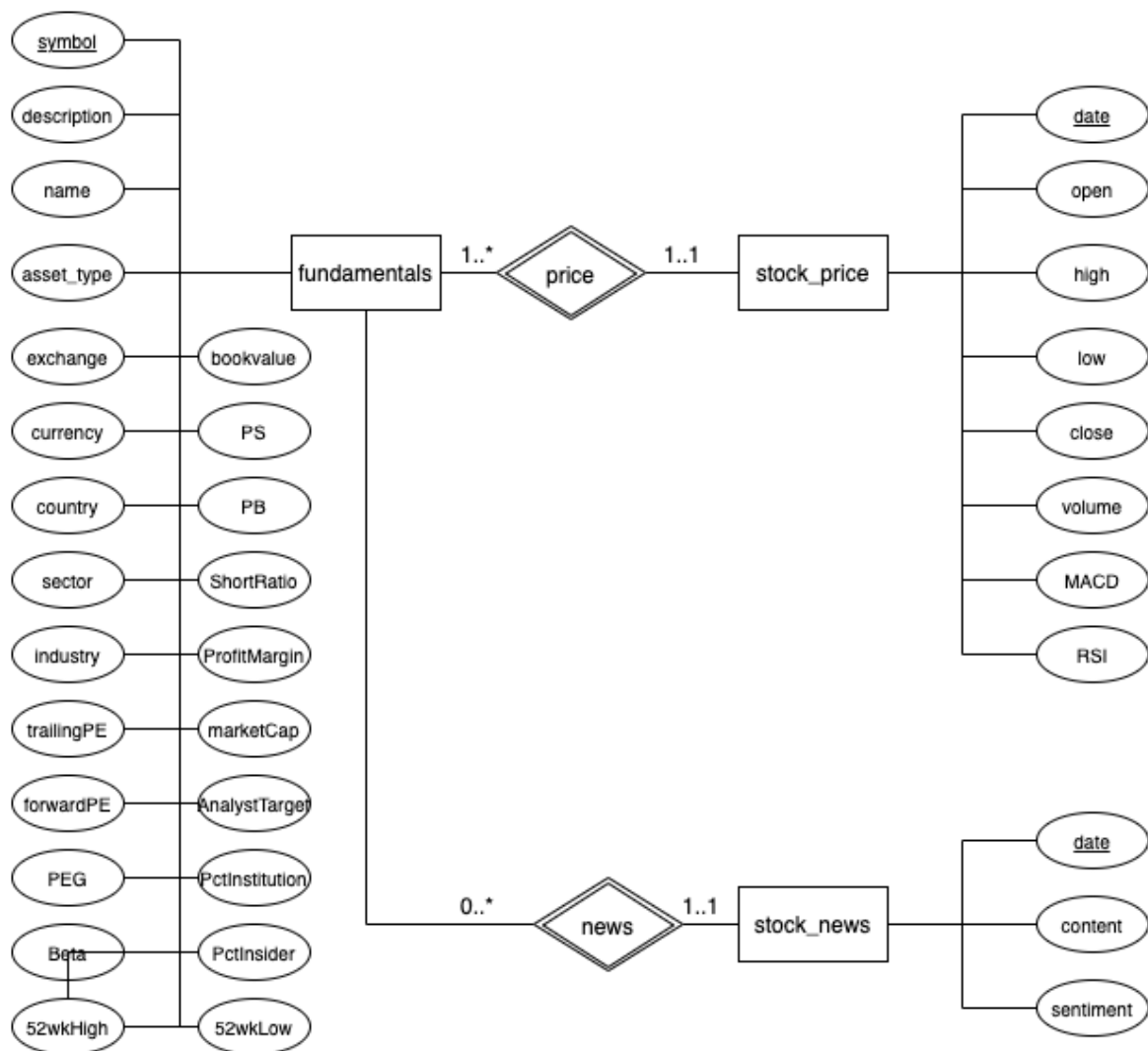
### 3. List of features you might implement in the application, given enough time

- A recommendation engine that provides recommendation to users for similar stocks and popular companies
- User authentication and user-level information
- News feed for each specific stock

### 4. List of pages the application will have and a 1-2 sentence description of each page

- **Home Page:** market overview of index such as S&P500 and performance chart
- **Stock Screener:** a stock screening page to select stock based on different criterias
- **Detail Page:** a display of full details of each stock and their financial performance
- **Comparison Page:** a side-to-side comparison to show people of any two stocks they like to compare
- **Watchlist Page:** a dynamic customized watchlist for user to add/delete stocks they follow

### 5. ER diagram



## 6. SQL DDL for creating the database

- **Fundamentals** (symbol, description, name, asset\_type, exchange, currency, country, sector, industry, trailingPE, forwardPE, PEG, beta, 52wkhigh, 52wklow, pct\_insider, pct\_institution, analyst\_target, marketcap, profit\_margin, short\_ratio, PS, PB, book\_value)
- **Stock\_Price** (symbol, date, open, high, low, close, volume, macd, rsi)
- **Stock\_News** (symbol, date, content, sentiment)

## 7. Explanation of how you will clean and pre-process the data

- We will implement a python program to perform the data processing.
- **Step 1:** Put together a list of all the tickers that we want to input into our tables in the server and the time interval that we would like to use to extract our data.
- **Step 2:** Establish a database instance on AWS and then make connection to the server.
- **Step 3:** Perform a fetch request to the Alpha Vantage API to obtain the all the related stock data and export as a csv file in local drive (we will have a 12 second interval between each api call because the API allows only 5 calls per minute for free account)
- **Step 4:** Create a table in our database that contains columns with fields like ticker, date, price, high, low, etc.
- **Step 5:** Insert data from csv file for each ticker in the list into the table and remove any rows with empty fields or illegal format of data.
- **Step 6:** We think our data should be naturally straightforward with a unique ticker for each stock and all basic required data field in correct format.
- **Step 7:** Perform step 1,3,4,5 for a list of all industries of SP500 to generate a data table for industries and companies in each industry.
- **Step 8:** Make sure that the name of data fields in two tables are consisten(such as ticker/company name,etc.)

## 8. List of technologies you will use

- **Front-end:** React
- **Back-end:** NodeJs Express, Python for data processing
- **Database:** MySQL & MongoDB hosted on AWS

## 9. Description of what each group member will be responsible for

- Wenting Zhao, backend, front-end
- Ruikang Liu, data processing, front-end
- Puran Zhang, backend, front-end
- Cheng Chen, data processing, front-end