

Database Requirements Document:

Objective:

- I've recently been paying a lot of attention to the market and have been passionate about my investments. I want to propose this idea: to make a dataset to help investors track high portfolio personals like politicians etc...
- Although the info is public, it could be very hard to see daily transactions and a data platform for that would be critical for new investors who want to get the experience and to be able to analyze the flow of the market and why investors make certain moves. A lot of people believe insider info plays a huge role, which it certainly does, so this would help the public

Target:

- Make a database that showcases all the current transactions in real time, similar to open insider but now its not restricted to employees
- Essentially, this info is public so that investors don't feel like they are missing out on anything
- Offer insights into market movements and potential reasons behind specific investment decisions.

Rules:

- Investor Information: The database must store detailed information about investors, including their name, affiliation, contact information, and net worth.
- Stock Information: Details about stocks, such as ticker, company name, sector, current price, and market cap
- Transaction Records: Every transaction made by the tracked individuals should be recorded, detailing the investor, stock, transaction date, type (buy/sell), quantity, and price
- Political Figure Tracking: Political figures are to be tracked separately, with their name, position, political affiliation, and other info stored
- Market Insights: The system should capture and store insights related to the market, specifying the source of the insight (investor or politician) and a description
- Portfolio Management: The database should enable tracking and management of investor portfolios, including the stocks owned, quantities, and average purchase prices

Nouns and Verbs

Nouns:

- Investor
- Stock
- Transaction
- Political Figure
- Market Insight
- Portfolio
- Net worth
- Transaction
- Quantity
- Position
- Source
- Market Cap
- Insight Date
- Insight
- Current Price

Verbs:

- Investor Actions:
 - Create, Read, Update, Delete (CRUD) for investors
- Stock Actions:
 - CRUD for stocks
- Transaction Actions:
 - Record transactions in real-time
 - Retrieve transaction details
- Political Figure Actions:
 - CRUD for political figures
- Market Insight Actions:
 - Capture and store market insights
- Portfolio Actions:
 - Track and manage investor portfolios

In-Memory Key-Value Storage Use Cases

Most Viewed Stocks:

- Cache the most frequently accessed stock details (like Ticker, Company Name, Sector, Current Price, Market Cap) to reduce load on the database and speed up query response times.
- Data from Documents: Use the _id, StockID, and Ticker as keys, and cache details for quick lookup.

Current Logged-In Users:

- Store session data for logged-in users to quickly access user-specific data and preferences.
- Data from Documents: Use InvestorID as a key to store and retrieve user sessions, along with quick access to associated portfolios and transactions.

Recent Transactions Cache:

- Temporarily hold the latest transactions in memory for immediate retrieval before they are batch-processed into the main database.
- Data from Documents: Use TransactionID as a key, storing details like InvestorID, StockID, Transaction Date, Type, Quantity, and Price.

Insight and Analysis:

- Cache recent market insights linked to specific stocks and political figures to provide rapid access to these insights for users.
- Data from Documents: Use MarketInsightID as a key, associating insights with StockID, SourceID (which can be linked to PoliticalFigureID), and other details for quick retrieval.

Political Figures' Recent Activities:

- Cache recent activities or news related to political figures to show how their actions might influence market movements.
- Data from Documents: Use PoliticalFigureID as a key, caching recent transaction data and market insights related to these figures.