

A Stock Watchlist App

The goal of this app is to provide a way for a user to manage watchlists of quotes.

Framework Requirements

- You must use <https://svelte.dev/> to develop the front end.
- You may use a server side framework if you like, it's not required
 - If you choose to use a database, it must be an embedded one like Sqlite
 - Local browser storage is acceptable
- You should use a modern CSS layout framework such as flexbox or css-grid.
- You *cannot* use any jQuery

Github

- Your code submission must be committed into a single github project. It should be a private one.
- Your github project must include a "README.md" file that has clear instructions on how to setup and run your project.
- You will give @hcnnguyen5, @dec0y, and @dgore7 read access to your repository when you're ready for review.

Quote Data Source

You should use IEX Cloud's API to get the current stock prices: <https://www.iexcloud.io/docs/api>

You should be able to use a free trial plan.

Symbol Search Endpoint

<https://vast.tastyworks.com/symbols/search/><Search String>

Example:

<https://vast.tastyworks.com/symbols/search/GOO>

Application Requirements

1. When the app first starts up, the user should be prompted to enter their username.
 - a. A password is not required. User registration and proper authentication should not be done for this exercise.
2. After the user is established, the app should transition to a Watchlist screen.
 - a. If this is a new user, a default watchlist should be created for the user

- i. The default watchlist's name should be "{user's first name} first list"
 - ii. The default watchlist should have the following symbols: AAPL, MSFT, SPY
 - b. If this is a returning user, the user's previously created watchlists should be loaded
- 3. The Watchlist screen should consist of the following elements:
 - a. A way to switch watchlists
 - b. The watchlist table composed of the following columns:
 - i. Stock Symbol
 - ii. Description
 - iii. Bid Price
 - iv. Ask Price
 - v. Last Price
 - c. A way to modify a watchlist
 - i. Add a symbol to an existing watchlist
 - ii. Remove a symbol from an existing watchlist
 - iii. Create a new watchlist with a given name
 - iv. Delete a watchlist
- 4. When the user is adding a symbol to a watchlist, they should be presented with a symbol search screen to allow for auto-completion of a symbol.
 - a. Symbol auto-completion should be done using the symbol search endpoint specified above
- 5. The application should refresh the quotes on the active watchlist every 5 seconds.
- 6. When a user selects a symbol from a watch list it should bring up a new view that displays the Stock Symbol, Bid, Ask, and Last Price as well as a chart.
 - a. The chart should show the last 30 days of the symbol's last price history.
 - i. The stock price history chart should render a point for the average price per historic quote
 - ii. You can retrieve the last 30 days of data from IEX's chart api.
 - iii. Feel free to use an open source library to interact with IEX.
 - b. The quote prices should update every 5 seconds as well.