**2020**

Connor McHugh

n10522662

***Super Stocks***

CAB230 Stocks API – Client Side

Table of Contents

[Introduction 2](#_Toc40299416)

[Purpose & Description 2](#_Toc40299417)

[Completeness & Limitations 2](#_Toc40299418)

[Use of End Points 3](#_Toc40299419)

[/stocks/symbols 3](#_Toc40299420)

[/stocks/{symbol} 3](#_Toc40299421)

[/stocks/authed/{symbol} 4](#_Toc40299422)

[/user/register 4](#_Toc40299423)

[/user/login 5](#_Toc40299424)

[Modules Used 5](#_Toc40299425)

[ag-Grid-react 5](#_Toc40299426)

[reactstrap 5](#_Toc40299427)

[Application Design 5](#_Toc40299428)

[Navigation & Layout 5](#_Toc40299429)

[Technical Description 6](#_Toc40299430)

[Architecture 6](#_Toc40299431)

[Test Plan 7](#_Toc40299432)

[Difficulties / Exclusions / Unresolved & Persistent Errors 8](#_Toc40299433)

[User Guide 8](#_Toc40299434)

[References 10](#_Toc40299435)

[Appendix A 10](#_Toc40299436)

[Appendix A - 1 10](#_Toc40299437)

[Appendix A – 2 10](#_Toc40299438)

[Appendix A - 3 11](#_Toc40299439)

[Appendix A – 4 11](#_Toc40299440)

**Re: The video demonstration:**

Unfortunately, due to the fact that I do not own a microphone, I was not able to do a voiceover for the video, but all points that I would have discussed will be mentioned somewhere in the report. Also, in the video, when I demonstrate filtering by date, the selector pop-ups do not show up in the OBS window capture, please be assured that the selectors do in-fact work.

# Introduction

## Purpose & Description

The *Super Stocks* web application is designed to allow users to view and analyse stock market statistics taken from the REST API. The application provides a number of ways of making this easier and more accessible to the user, such as the ability for filter stocks by industry, and view statistics in a certain date range. The latter of these techniques requires the user to have registered an account and to be logged in.

In the making of this web application, it was decided that a number of modules would be used in order to enhance its functionality/aesthetics. The first of these is ag-Grid-react which was used to create nicely styled tables with many customisable features. The second of these modules is reactstrap, which is a react implementation of the ever-popular Bootstrap framework. This was used for most of the components and controls in the application in order to create an attractive and uniform look and feel.

## Completeness & Limitations

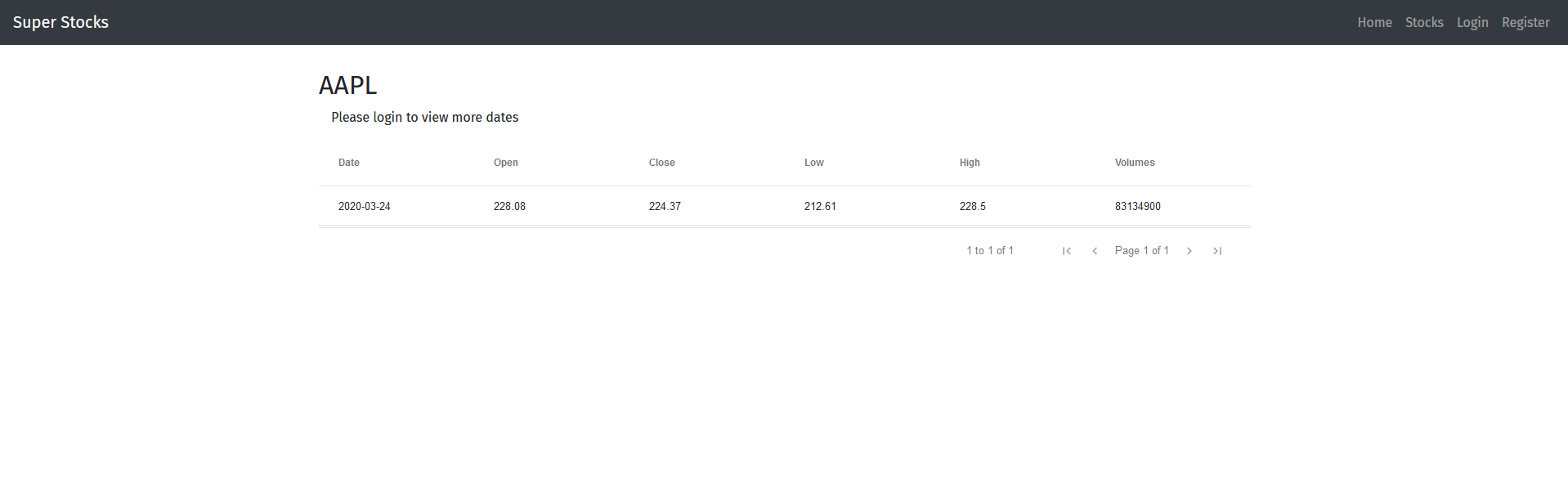
The *Super Stocks* web application uses react routing for navigation, and is capable of handling both the unauthenticated and authenticated query routes, as well as the user endpoints. The data fetched from the Query endpoints is presented in an ag-Grid table, user input is handled by the reactstrap form components.

# Use of End Points

## /stocks/symbols

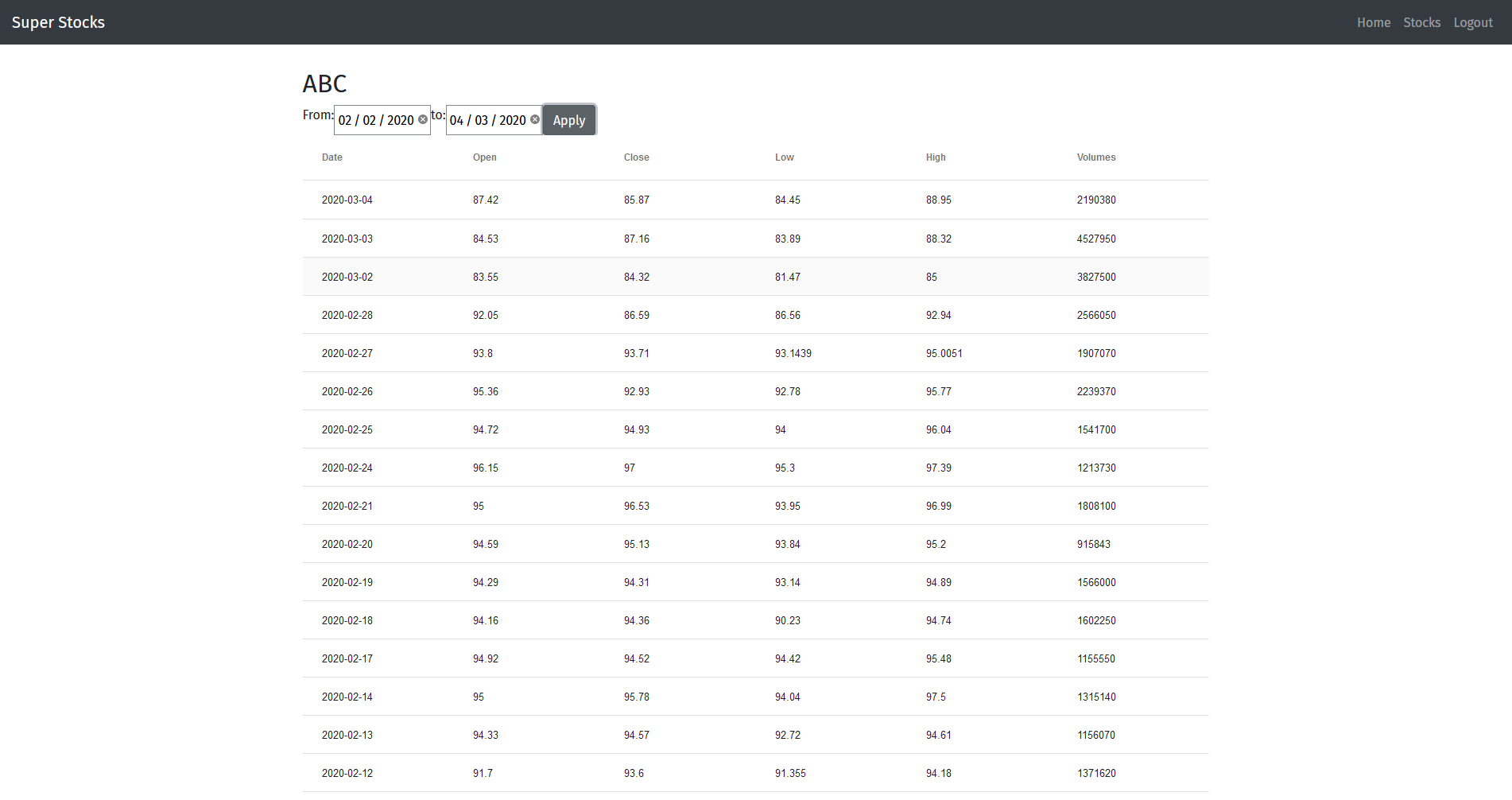
This end point is used to fetch all of the stocks when first going to the *stocks* page. It also allows for the results to be filtered by industry.

## /stocks/{symbol}



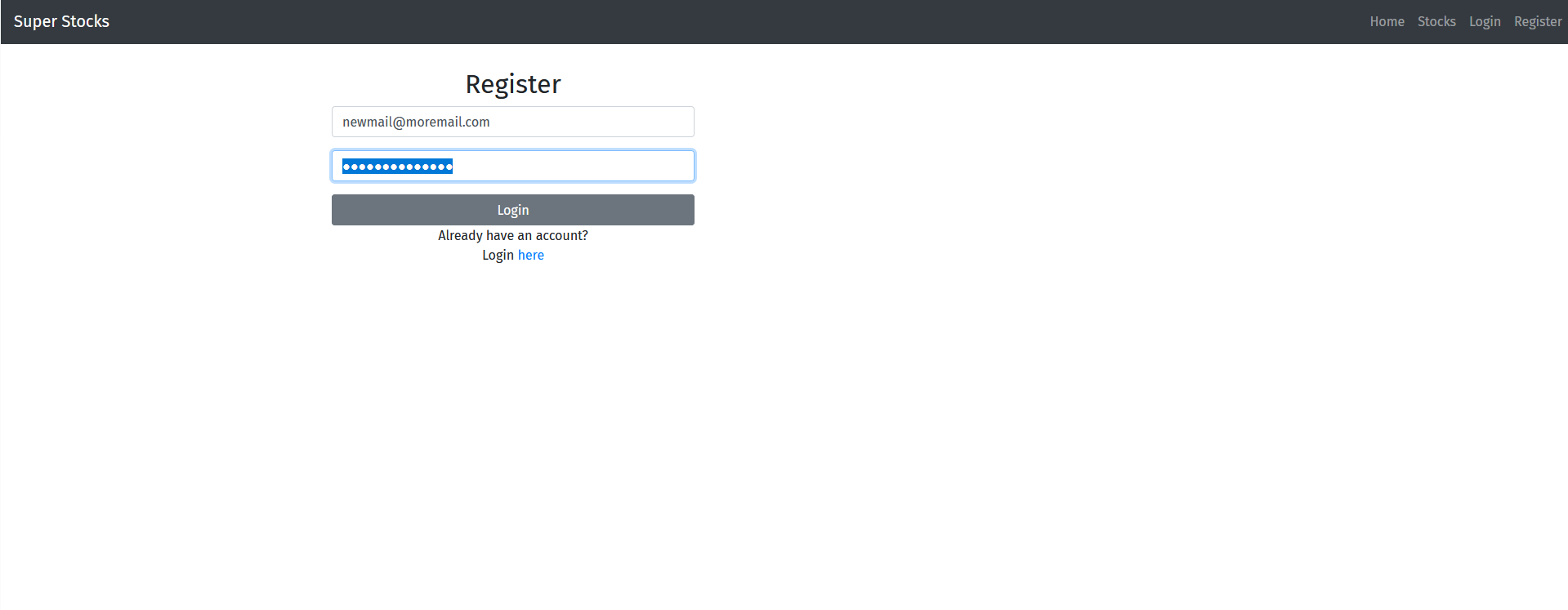
This end point is used when an unauthenticated user clicks on a stock in the main stock list. It shows the latest information from the selected stock.

## /stocks/authed/{symbol}



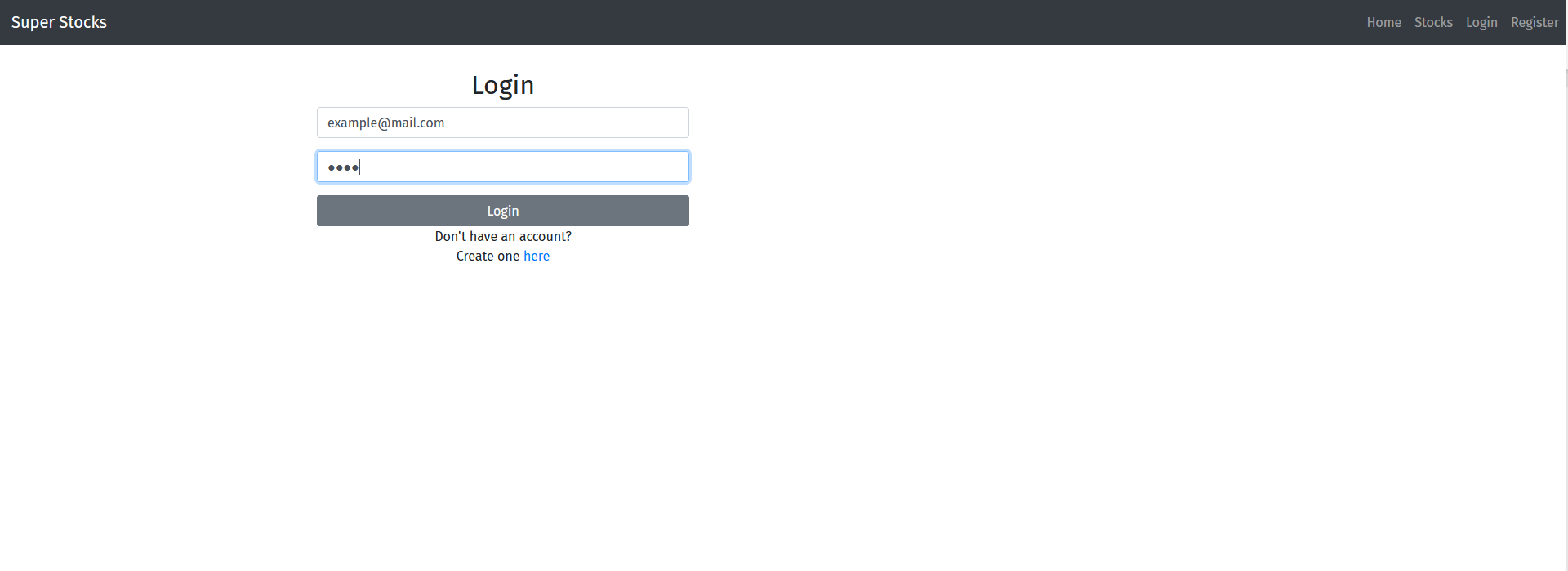
This end point is used to display all information about the given stock within the date range provided. It is only available to users who are logged in.

## /user/register



This endpoint is used in order for users to register a new account in order to view more detailed information about a stock.

## /user/login



This end point is used for users that have already created an account to login in order to view more detailed information about a stock.

# Modules Used

## ag-Grid-react

This module allows for fully featured tables, with built-in filtering, sorting, and features to handle cell/row selection.

<https://www.ag-grid.com/react-grid/>

## reactstrap

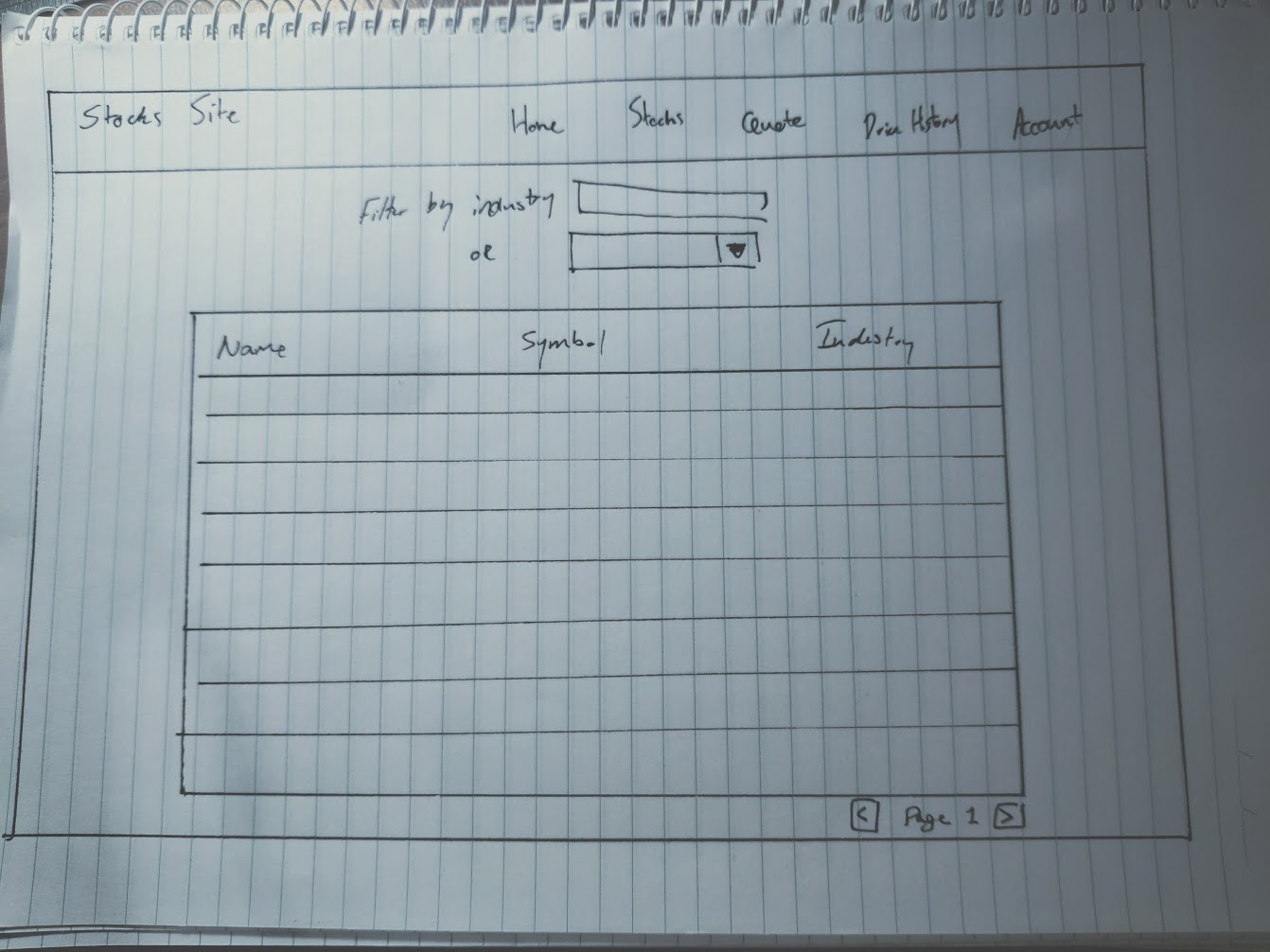
This module allows for the easy implementation of Bootstrap 4 components into a react webapp, without the need to use jQuery.

<https://reactstrap.github.io/>

# Application Design

## Navigation & Layout

The structure of the web app follows that of the layout of a fairly standard web page with a navigation bar spanning the width of the page at the very top with links to the various pages of the application. This structure was chosen as it will be most familiar to most users, so that they will not experience as learning curve when they first begin to use the application.



# Technical Description

## Architecture

The image below (Figure 1) shows the src directory of the web application. Upon running the web app, the file index.js is called, which then renders the file App.js. App.js handles all of the react routing for the application, so it renders the .jsx files in the pages directory depending on the url.

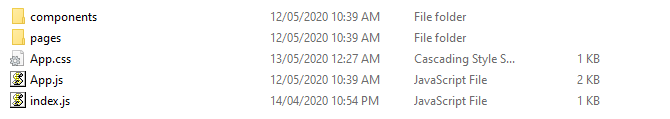


Figure 1: ./src directory

Inside the pages directory (Figure 2) are all of the .jsx files. Each of these is for an individual page of the web application.

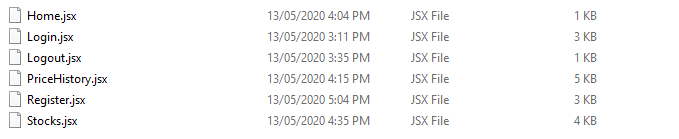


Figure 2: ./src/pages directory

Inside the components directory (Figure 3), are the .jsx files for any custom components used by the web application. The only component used at this the navigation bar at the top of the application (there are logged in and logged out variants of this component).



Figure 3: ./src/components directory

## Test Plan

|  |  |  |  |
| --- | --- | --- | --- |
| **Task** | **Expected Outcome** | **Actual Outcome** | **Screenshot**  **(Appendix A)** |
| View stocks | All stocks should be displayed | All stocks are displayed | A - 1 |
| Filter stocks by partial industry | Stocks of industries containing the entered string should be shown | Stocks of industries containing the entered string are displayed | A - 2 |
| Filter stocks by full industry name | Stocks of the entered industry should be displayed | Stocks of the entered industry are displayed | A - 3 |
| Login | User should be able to login to an existing account | User can login to their account | - |
| Register | User should be able to register a new account | User can create a new account | - |
| Logout | User should be able to logout of there account | User can logout of their account | - |
| Filter by date | Authed users should be able to filter stock data by date | Authed users can filter stocks by date | A - 4 |
| Display chart | Authed users should be able to see a chart of close values for stock data | Chart functionality not implemented | - |

## Difficulties / Exclusions / Unresolved & Persistent Errors

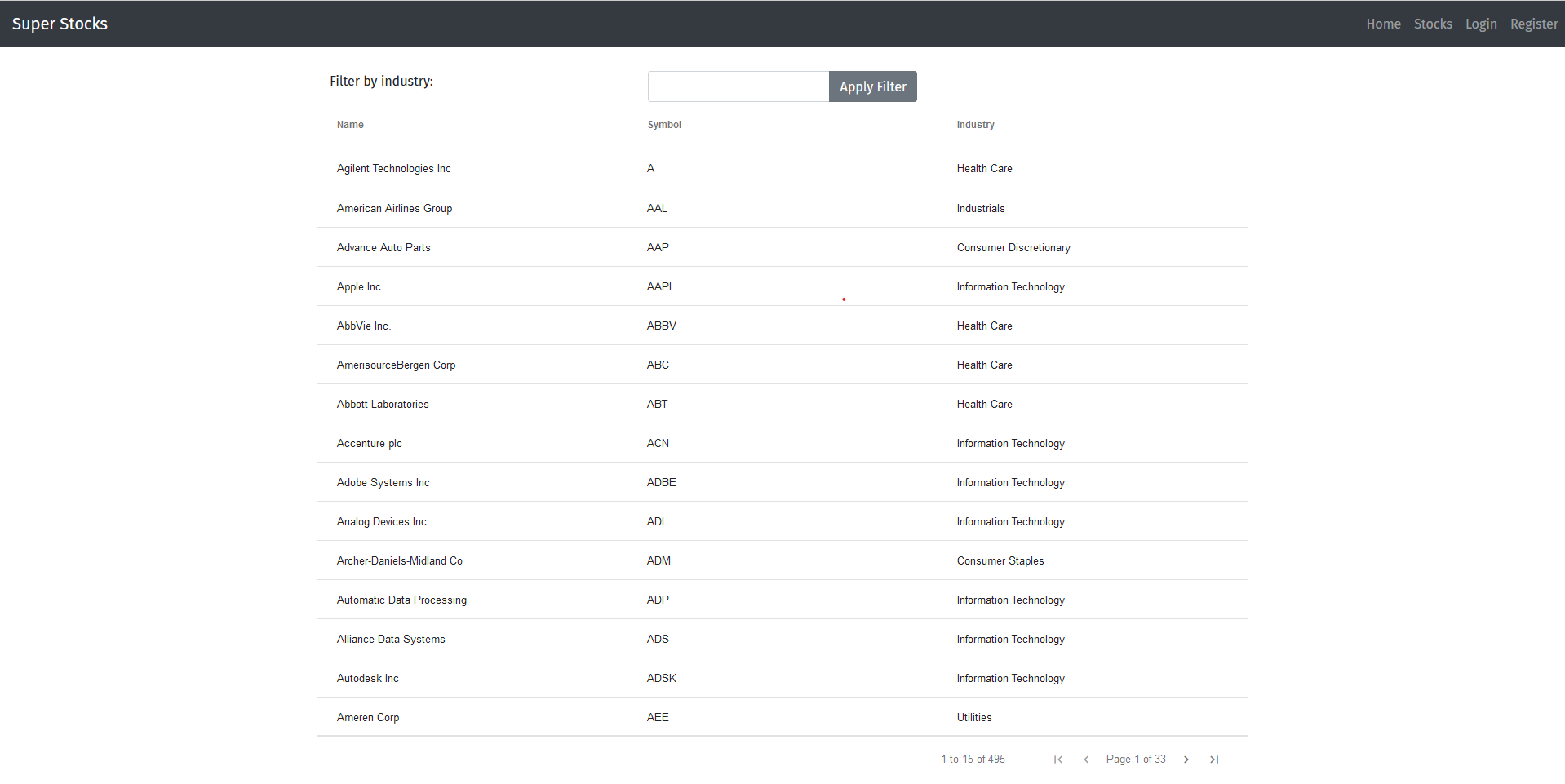
I faced a number of difficulties throughout the development of this web application. The first of these issues was that I would get errors when trying to populate the table with data from the API when the query to the API returned only one result. The issue was caused by the fact that I used the map function to populate the data, but it only works with an array, so single values had to be put into an array for map to work.

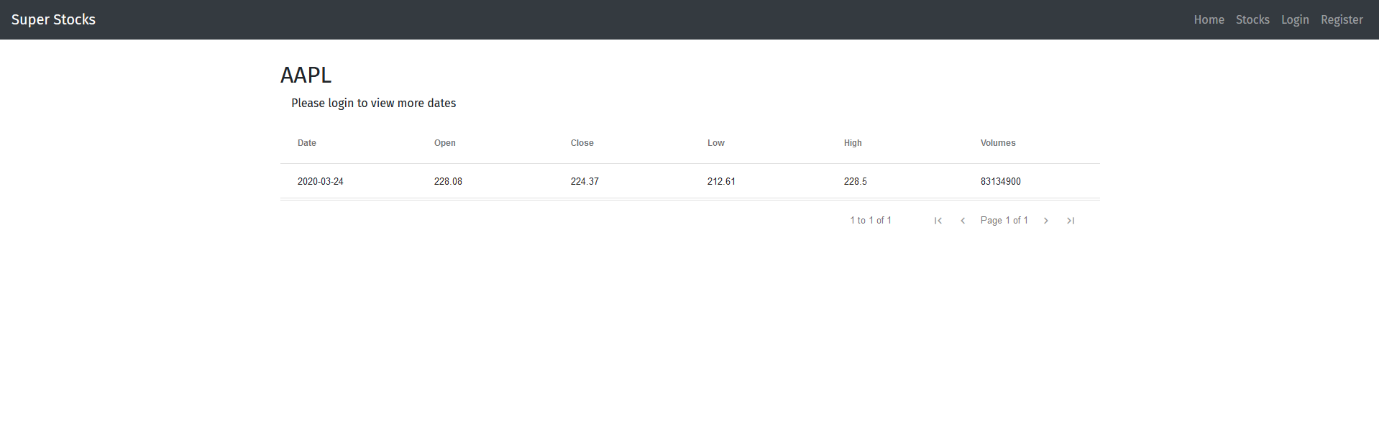
The second major issue that I encountered was with elements that would have to change depending on whether or not the user was logged in. The issue was caused because the components were not being re-rendered after logging in. This was fixed by forcing a page reload after logging in or out.

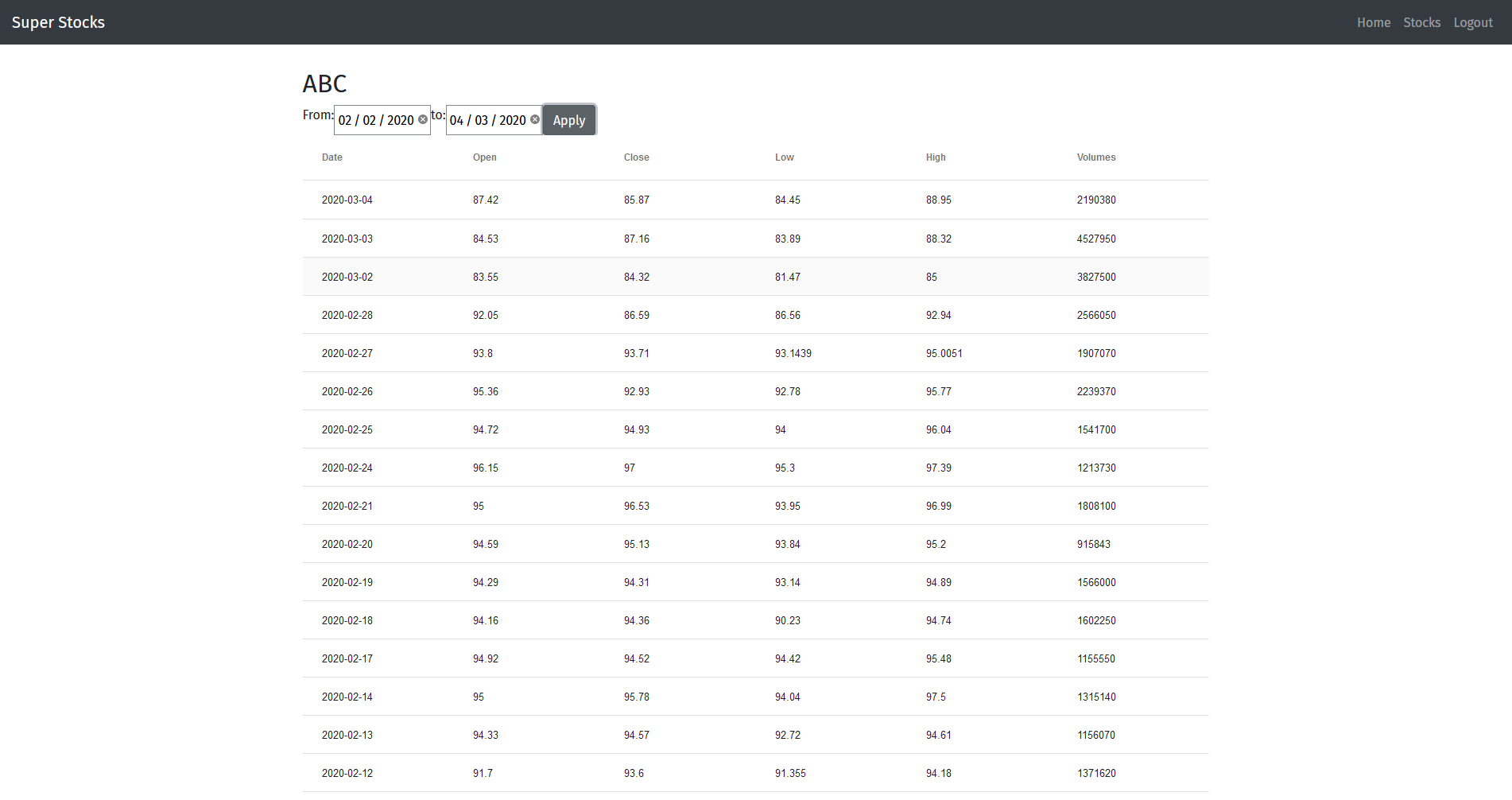
The only issue that I believe to remain in the web application at this time is related to error handling. In some cases, when there is an error, a single error message is displayed no matter what the actual error is.

The only feature that has not being implemented is charting for authenticated stock data, which I would like to have implemented using chartJS.

# User Guide

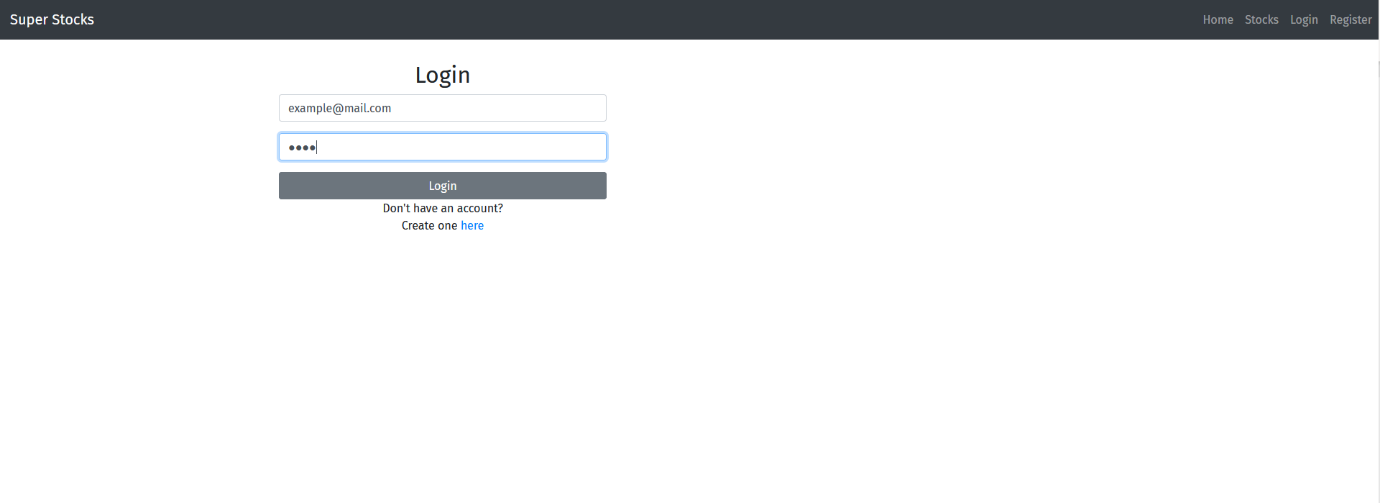
To filter stocks by industry: go to *Stocks* page, enter an industry name and click *Apply filter*

To see more information about a stock: click on it from the list of stocks from the *Stocks* page (you will see more information if you are logged in)

OR (if authenticated)

You can also filter by date from here by clicking the date selectors, choosing start and end dates from the popups, then clicking apply.

To login or register: go to the login / register screen and enter your credentials.

To logout, click the logout button in the navigation bar, then the logout button again on the confirmation screen.



# References

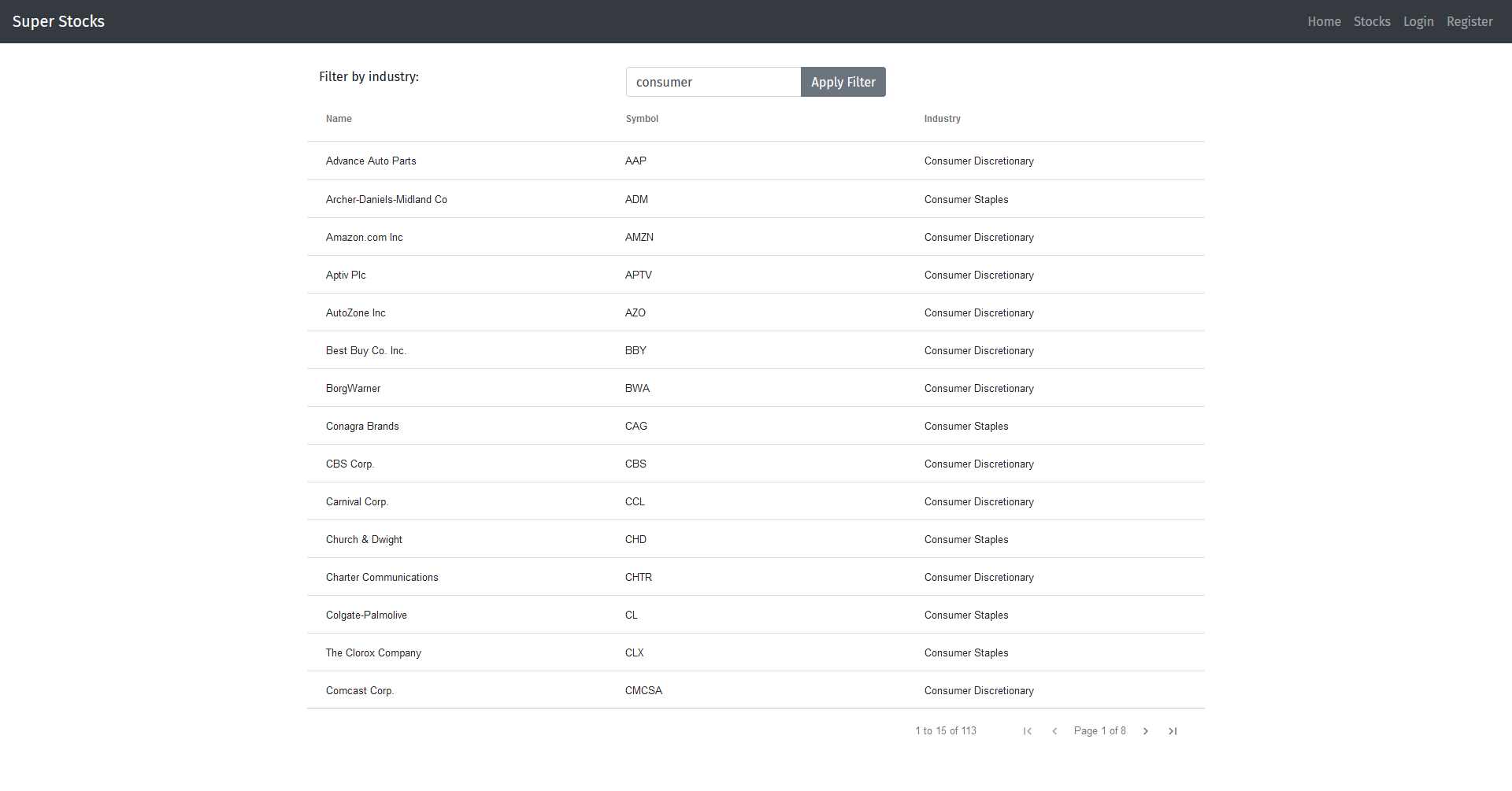
ag-Grid (n.d.). *ag-Grid Documentation.* [*https://www.ag-grid.com/documentation-main/documentation.php*](https://www.ag-grid.com/documentation-main/documentation.php)

Reactstrap (n.d.). *reactstrap components*. [https://reactstrap.github.io/components/](https://reactstrap.github.io/components/alerts/)

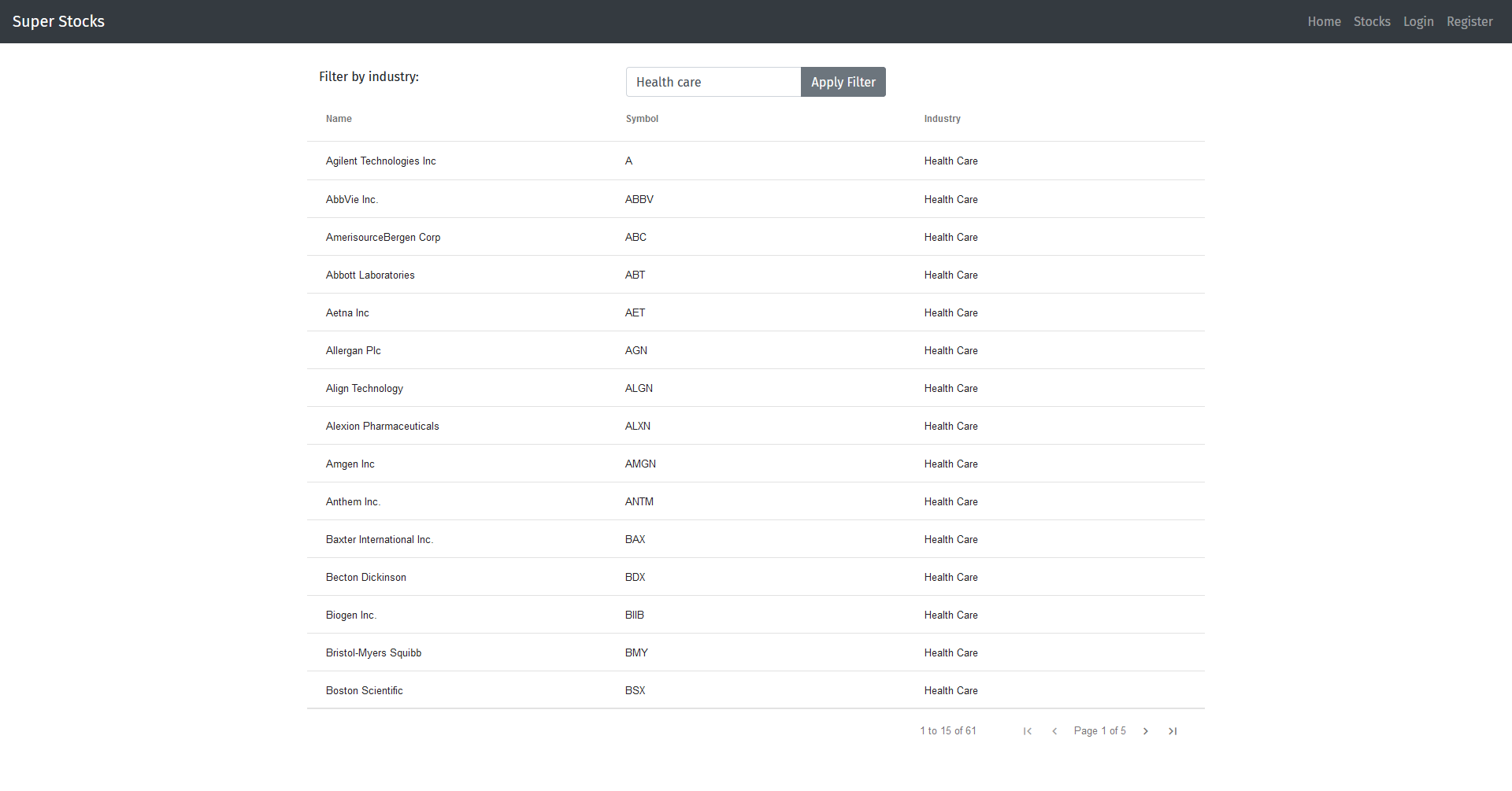
# Appendix A

### Appendix A - 1

### Appendix A – 2



### Appendix A - 3



### Appendix A – 4

