# Stock Viewer Shiny App

Ignacio Almodóvar & Alejandra Estrada

3/21/2022

## Introduction

For this Shiny-App project we want to develop an interactive web app that allows to visualize and analyze the historical stock market price for all the companies listed on any stock exchange.

We want to focus more on the use of different shiny functions and web development design rather than in developing algorithms in R. We believed that the aim to this project was to play around with shiny and explore all its potential. Therefore, the web won't be based on very complex functions that we have created to do statistical analysis. Instead, we will try to build a good looking web application, that feels smart and reliable. It will do very few things but also well implemented.

As we said we are basically going to work with stock markets datasets and try to plot/analyze them. To work with stock ralated time series data frames, there exists an very useful R package. The package is called "quantmod" and will be the one used to directly obtain data from the web yahoo finance and to do different calculations like calculating returns and moving averages.

## Design Requieremnts

Before starting to develop the application we want to have a some defined functionalities that we want to achieve with this web. Basically we want the app to have two main functionalities:

### Price visualization

As like any stock market web, the main functionality must be the possibility to see the historical price for the company that we want in a selected period. Therefore, one the main functionality to achieve is to be able to plot a graph with the price of a the company chosen and being able to select the time period that we want to visualize.

Once we have this main functionality we will like to be able to choose between any companies listed on the stock exchange and being able to customize a little bit the plot, so we can understand it better, like changing the scale and obtaining more information like the moving average, which sometimes is more useful to analyze stocks than the adjusted closed price. Also we would love to be able to show a candle chart, as it gives much more information about the day performance of the stock price.

### Returns visualization

When analyzing a company listed in the stock market, it is someteimes more important to check the returns on a period that the stock has given rather than the price. Therefore we will like to implement this option on our web.

# Development

Once we had clear what were our main goals started building the application. We decided first of all that we were going to split each functionality in a different window inside the app. This way, each of the window will be clear and easy to use, without many options in the same page. Before starting to code we also decided that everything in our app was going to be reactive, so any changes selected or chosen will be directly shown. Once we got the scheme defined, we started coding. We created 3 windows:

### Home

This page will be the main page of the application. It will open by default on this page and it only includes relevant information about the shiny web so anyone that runs it can have a brief idea on what does this app do.

We also included a logo that we made to give a formal look to the page.

### Price visualization

This would be the main functionality of the web page. The quantmod function that extracts data from the internet is getSymbols(). You have to specify the ticket of the stock that you want to visualize and the period. Therefore we built some inputs elements so the client can customize its search:

- TextInput: To specify the ticker symbol. It will return a string object that will be used as an argument of the function getSymbols().
- Two DateInputs: Both for selecting starting and end point of the period that we want to analyze. They will both return a character indicating the date.

Once we got this part working we started creating different functionalities so we set up a checkboxInput to show the moving average of that period. The moving average also requires a data window from which to start computing the average, so we included a number input. Once the flag of the Moving average is activated, it will add a line to the plot.

We believed that it is also very important to show the candle chart when analyzing company's stock prices. Therefore we also included a check-box that shows the candle plot for the period and company given. Also, as the moving average check point was already implemented we have to also make it available when the candle chart was plotted.

### Returns

For this functionality we created another window inside the application. We wanted to calculate the returns given by the stock in the period selected when evaluating the prices. However, for studying returns you have to set the period for which you want to see the returns. Therefore we created a select one choice input with the defined options (Daily, Weekly, Monthly, Quarterly and Yearly).

For analyzing this plots it is also important to clearly see the dates of the break points in the plot, therefore believed that being able to adapt them to the returned period selected was very important. Therefore we also added that option to the plot.