4. Work Done

# 4.1 Time Series Analysis

We used **ARIMA (Auto Regressive Integrated Moving Average)** model for our Time Series Analysis. We created a python notebook using **Jupyter Notebook** for this purpose. We used **NSEpy** library to collect historical stock data of **RELIANCE** stock from 1st January 2019 till 31st August 2020. Our goal is to train an ARIMA model that will predict the closing price of the given stock for the next 30 days based on the test data given.

After importing the dataset from **NSEpy**, we remove the unnecessary variables from the dataset so that only the closing prices remain in the dataset. After that, we run the **auto\_arima** function from **pmdarima** library to generate an optimum ARIMA model. Using the generated model, we predict the closing prices of the given stock for the next 30 days.

To check for the accuracy of our predictions, we split our dataset into **train** and **test** dataset. We train our ARIMA model on the train dataset and then predict the closing prices for the length of the test dataset. We then compare the predicted dataset and the test dataset with the help of **sklearn.metrics** library to find out the **Root Mean Square Error** and the **Mean Absolute Percentage Error** of our model.

# 4.2 Wireframe Design

To popularise the use of machine learning for stock market trading, we decide to create an **app** which will display the predicted stock prices to the end user, which will help the user in making informed choices regarding the trading of stocks. Before we started developing the app, we decided to create a **wireframe** for our app using **Figma**. Figma is a vector graphics editor and prototyping tool which has dedicated tools for wireframing and prototyping. Using a wireframe helped us save valuable time while coding the app as it provided us of a clear vision of what our will finally look like.

# 4.3 API Deployment

We used **Flask Microframework** to convert our data analysis into an API and deploy it on **Heroku.** Flask is a micro web framework written in **Python**. It is classified as a microframework because it does not require particular tools or libraries. **Heroku** is a cloud platform as a service supporting several programming languages

# 4.4 App Development

We used **Flutter** Framework by Google for the development of our application. Flutter is an open-source UI software development kit created by **Google**. It is used to develop applications for Android, iOS, Linux, Mac, Windows, Google Fuchsia and the web from a single codebase. Flutter uses **Dart** programming language**.** Dart is a client-optimized programming language for apps on multiple platforms. It is developed by Google and is used to build mobile, desktop, server, and web applications. Dart is an object-oriented, class-based, garbage-collected language with C-style syntax. Dart can compile to either native code or JavaScript. It supports interfaces, mixins, abstract classes, reified generics, and type inference. We have used **Firebase** to host the database of our app and perform authentication.

Our app consists of an interface where new users can **signup** and existing users can **login** into their accounts. After login, users will be greeted by a list of stocks which they have added to their account on their **dashboard**. The users can add more stocks in the **add stocks** page. To get the stock price prediction graph, users will have to click on the stock of their choice and they will be directed to the **analysis** page. On the analysis page, users will find a **floating** **button** which will direct users to a page which will show **more information** to the user. The page will show the **Root Mean Square Error** and the **Mean Absolute Percentage Error** to the users. The users will find the **Profile** page, **Terms and Conditio**ns page and the **FAQs** page in the menu button.