CSCI 4610 Project Report

Abinash Narendiran 100588875
Saijeeshan Ketheeswaran 100589716
Rohil Arya 100585195

Introduction

For our project, we did a stock prediction app on android with python implementation. The objective of this app is to be able to check the European stock market and recommend what stock to buy based on the predicted stock value, when to buy the stock based on the date and when to sell the stock to ensure a maximum profit, so it will easier and more convenient for the user instead of doing it manually. We want to help people by making it easier for them to check their stocks and alleviate the pressure's have having to keep up with the constantly . We didn't get the exact way and the best way that we visisioned for this project, but it is works the way we want. We want to get it working and make sure we get all the requirements to make this project good. It is not the outcome we were hoping for, but the logic is correct.

System Requirements

Our system is separated into two parts; the frontend and application portion is done in Android Studio, while the backend is implemented in Python. The system requires Java for Android Studio in order to implement the login function to make sure our users are logged in order to view their list of stocks. This requires the implementation of firebase to store user info (email and country) into a realtime database. If users do not exist in the database they must sign up. Our application uses a list view to display the list of stocks. With the use of an onclicklistener the user can select a stock from the list view. They will then be redirected to an activity with the URL link to the stock's data in the form of a CSV file.

The major components of the system is the the parts of each stock from the data depending on what parts the user wants, Stock API where the data comes from, and it's get the quarterly reports ,menu where the user can select that parts of the stock they want to choose, programming language that we will use for the project and more components that come up which we will thinking of as the project goes on. For the parts of each stock from the data, the user would want the specific parts from the stocks to see that they can validation for the stocks for the next day or next week etc.

We need to the Stock API to get all stock data and the data will help the user with their stocks for the prediction for a seven day price movement. The data allows us get the quarterly reports. The menu allows the user to select on what parts of the stock they want , so it's easier get not all the data because of the memory. They can be independently developed by use researching and finding way figure how to code and learn new things that we never done or learned in previous courses. It can indentely tested by letting people that we know how to use stocks test it out for us or we can also test if we know what we are doing.

Project Description

Agents

In terms of Agents, our application is a software agents because there is no physical presence doing all the work only , only the application itself. It is a software agent because all our data comes from the internet , it is software based and runs in the back end also. Our application interacts with the environment , but not the physical world because everything that is is accessing comes from the internet , which are the stocks. Since physical world means it has to be something physical there, so like you can taste, see etc, which is not possible for our app. That is why our app is a software agent.

Type of Environment

In terms of type of environment, the abilities that our application can do it getting data from a stock website called Quandl and also get the prediction based on the data and neural networks that we coded using layers etc. Our goals for this application is to make sure that the user using the application can check if they want to buy this stock or not based on the predication , to make sure they won't lose money and gain money. Our observations are all based on the data , where depending on the high , low for each date in the data. The prior knowledge is the stock data that we have, which the agents needs to the highs,low, etc for each date and make sure it doesn't mess up.

Performance Measure

In terms of performance measure, it all depends on what you are using like a virtual machine, or terminal on OS like Mac, Windows etc. It does not working on a virtual machine as we tested it out and it's gives segmentation fault , and the application works in terminal on an OS. It would take a long time because we are teaching our AI in our application. It may take upwards of six hours to run because we are feeding the AI 8000 inputs of data and expect a specific outcome. We realized that the more we iterated through the learning process the better result we got so we increased it to 250 and it took about seven hours to give an output. The accuracy of the output was 70% (133/186).

Actuators

In terms of the actuators, considering that our program is a software agent and not a physical presence as mentioned before, it does not have any physical components that control

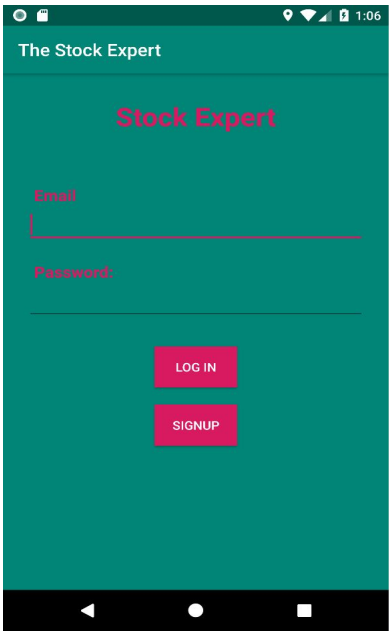
the functions of our system. Since our program is an application, the actuators in our case would be the android device that displays the list of stocks that the user chooses from. In the case of the backend, we have a visual plot that displays the stock value prediction in the form of a line graph and a terminal that displays the prediction value of the stock. There's also the firebase database which stores the url of the selected stock, which is the retrieved by the backend to make the predictions.

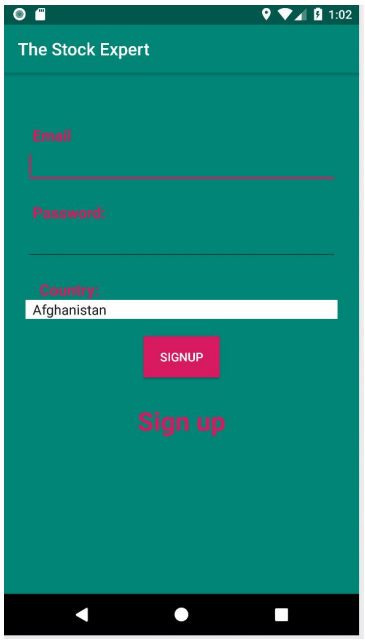
Sensors

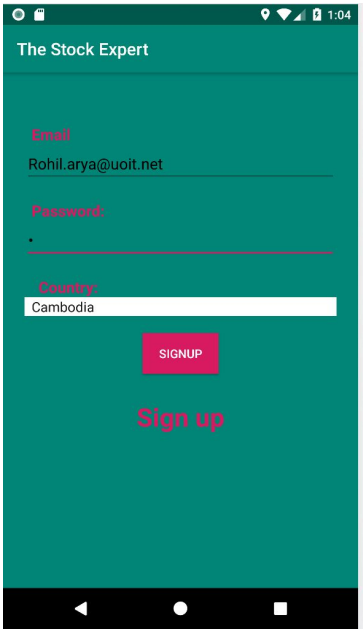
In terms of the sensors, the android device is able to sense the touch of a user's input when they select what kind of stock they want. This is then translated to an activity which displays the url of the stock. As for the backend portion, the sensors would be our neural networks. We are basically teaching our AI, which means that it must learn the data and use it to make predictions on the stock value.

Screenshots of Important Components

Login and Sign Up:

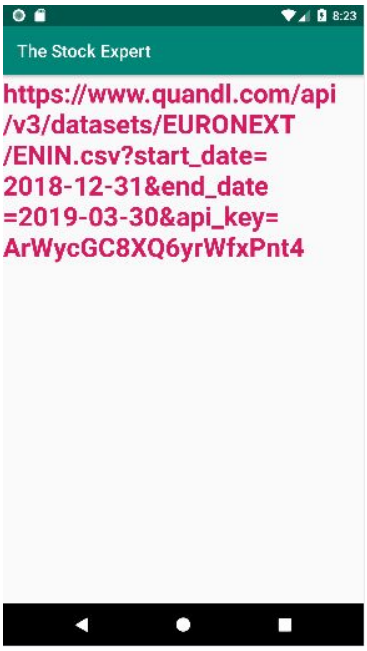




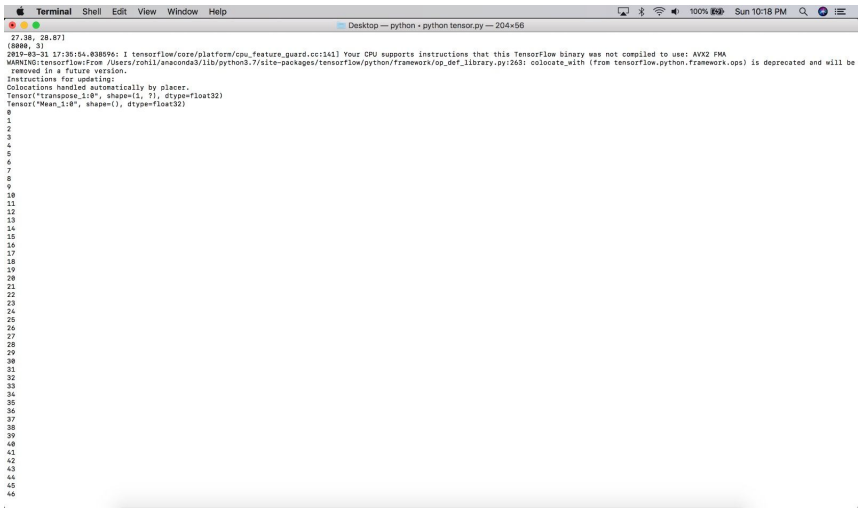


Stocks List Page:

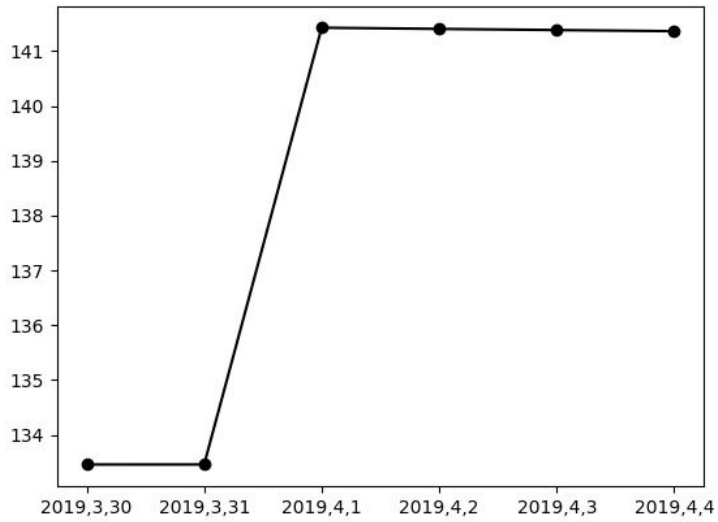
Stock URL Page



Terminal/Output



Graph



Link to github :
https://github.com/saijeeshanketheeswaran/The-Stock-Expert