FINANCIAL PLANNER:

MANAGE YOUR PORTFOLIO EASILY

BY

HARDIK GROVER

UTKARSH NIGAM

SUMMARY

This project was taken up as a course project under Engineering Analysis and Design course. Through this project we intend to make our contribution towards reducing the prevalent ignorance about the field of Finance and Personal Investment. As a result of this project, we have designed a web-app called **Financial Planner** as a small step towards this great cause.

Financial Planner is all an investor needs to manage and organise his financial portfolio. This web-app keeps track of all your stock holdings and informs about your net profit/loss. The client can search for any number of stocks available on the major stock exchanges and our web-app will give a detailed summary and customisable graph based on the historical data. The client can view all the stocks held under My Portfolio. The value of the portfolio keeps on updating in real time which lets the user know about their current standing. There is also the tedious and mind numbing task of stock selection and optimisation which exerts the investor since there is an infinite array of options, Financial Planner has a unique feature of optimising among various stocks in order to maximise profit while minimising risk and help the investor reach his/her financial goals. This feature is based on Markowitz Portfolio Selection Model based on maximisation of Sharpe Ratio.

Financial planner can also be used as a stock simulator by students and amateur investors in order to learn about the stock market without really investing real money in it. This would really help them in making informed and educated decisions before putting in their hard earned money.

CONTENTS

Summary 2

Contents 3

Introduction 4

Features 6

Portfolio Management 6

Stock Search And Analysis 7

Portfolio Optimisation 9

Implementation Details 10

Low level Design 10

Database Design 11

Django Backend Implementation 12

Data Fetching 13

User Interface 14

Portfolio Optimisation 14

Conclusion 17

References 18

Learning Resources 18

Data Sources 18

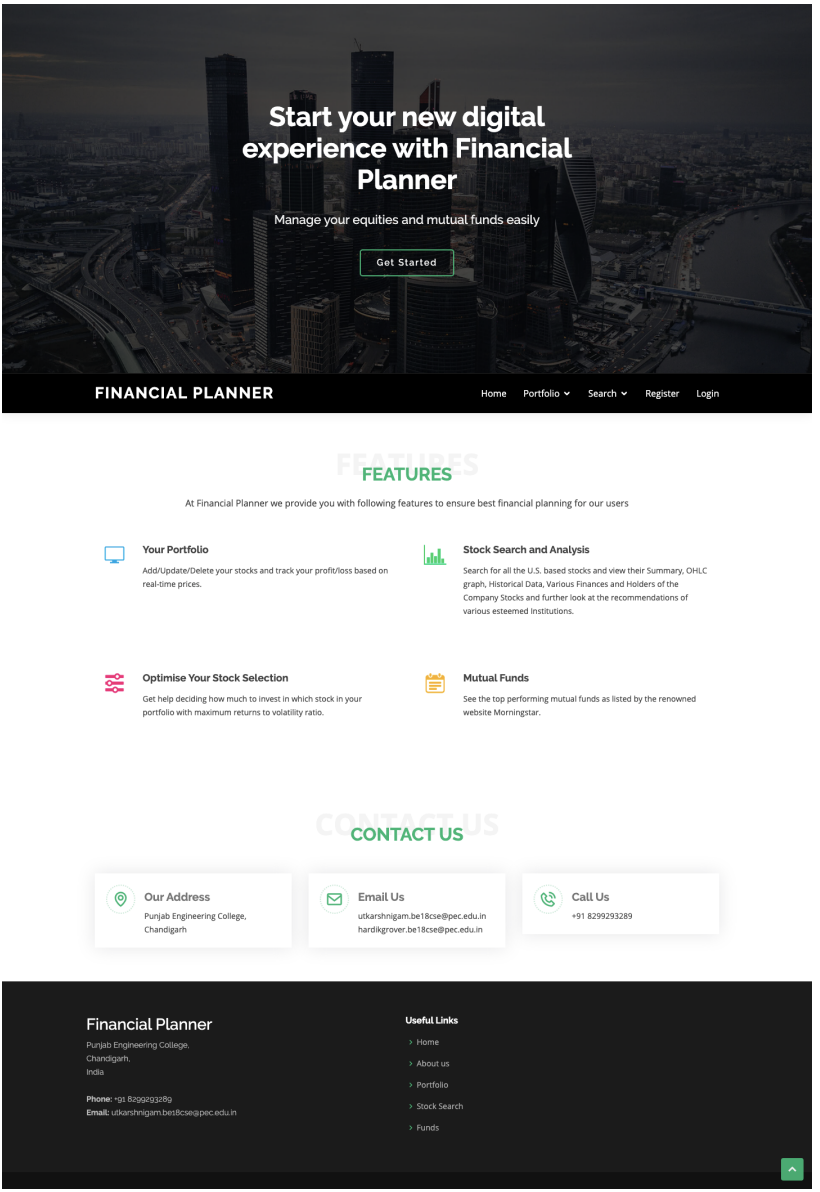
INTRODUCTION

According to a global survey, about a staggering 76% of Indian adults do not understand basic financial concepts and are unfortunately financially illiterate even today. The survey confirms the financial literacy rate in India has been consistently poor as compared to the rest of the world.

Investing in equities requires time, knowledge and constant monitoring of the market.For those who need help to manage their investments, a financial planner comes as an answer. The business of portfolio management has never been an easy one. Juggling the limited choices at hand with the twin requirements of adequate safety and sizeable returns is a task fraught with complexities. Given the unpredictable nature of the market it requires solid experience and strong research to make the right decision. In the end it boils down to make the right move in the right direction at the right time. The term portfolio management in common practice refers to selection of securities and their continuous shifting in a way that the holder gets maximum returns at minimum possible risk.

Stock exchange operations are peculiar in nature and most of the Investors feel insecure in managing their investment on the stock market because it is difficult for an individual to identify companies which have growth prospects for investment and thus arises the need of a stock search which can present real time analysis and buzz news related to the searched stocks.

The **main aim of this project** is to address these issues and develop a web-app that simplifies the process of managing portfolios and provides a basic idea of investment management to beginners by using one of the popular portfolio selection models.



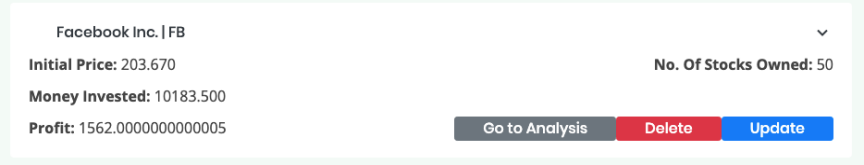
Financial Planner

FEATURES

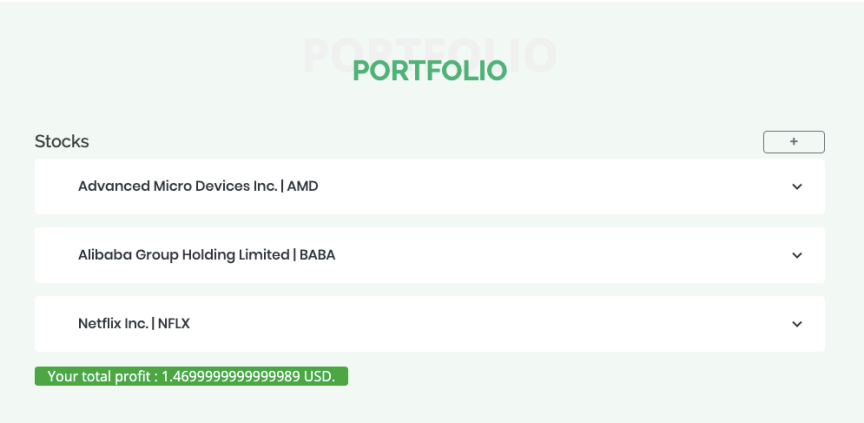
PORTFOLIO MANAGEMENT

Add/Remove/Update quantity of stocks owned from your portfolio, track your current profit/loss based on real-time prices for each stock owned and the entire portfolio. Each portfolio entry(owned stock) will show the following:

1. Stock Name and Symbol.
2. Initial Price at which the stock was brought.
3. Quantity Of Stock Owned.
4. Initial Investment.
5. Current Profit/Loss (Current Value of Investment - Initial Investment).
6. Link to view analysis(historical data, etc.) of that particular stock.

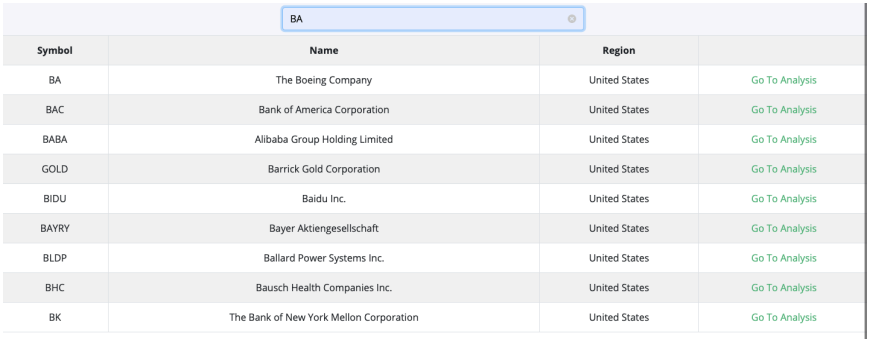


A Portfolio Entry



Portfolio Display

STOCK SEARCH AND ANALYSIS



BA

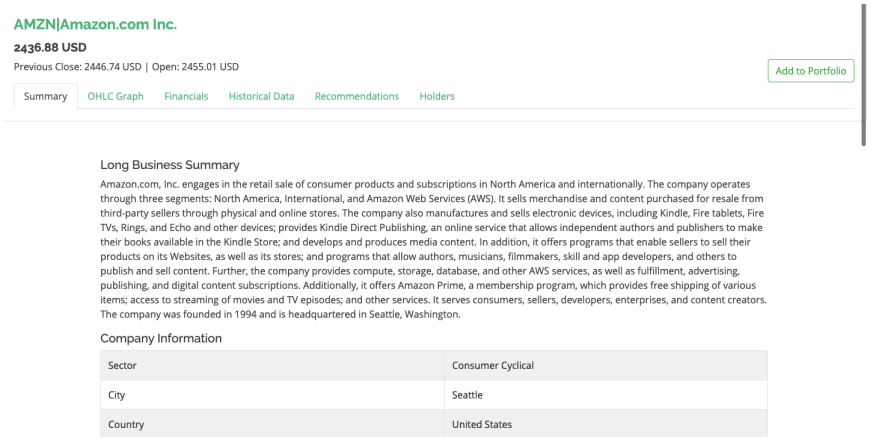
Symbol	Name	Region	
BA	The Boeing Company	United States	Go To Analysis
BAC	Bank of America Corporation	United States	Go To Analysis
BABA	Alibaba Group Holding Limited	United States	Go To Analysis
GOLD	Barrick Gold Corporation	United States	Go To Analysis
BIDU	Baidu Inc.	United States	Go To Analysis
BAYRY	Bayer Aktiengesellschaft	United States	Go To Analysis
BLDP	Ballard Power Systems Inc.	United States	Go To Analysis
BHC	Bausch Health Companies Inc.	United States	Go To Analysis
BK	The Bank of New York Mellon Corporation	United States	Go To Analysis

Stock Search

Search US stocks by keywords and get the top matches of the same. Each match displays a link to view the analysis of that particular stock. Analysis Page for a stock displays following fields:

1. **Summary**

It includes long business summary and basic company information.



AMZN|Amazon.com Inc.

2436.88 USD

Previous Close: 2446.74 USD | Open: 2455.01 USD

Add to Portfolio

Summary | OHLC Graph | Financials | Historical Data | Recommendations | Holders

Long Business Summary

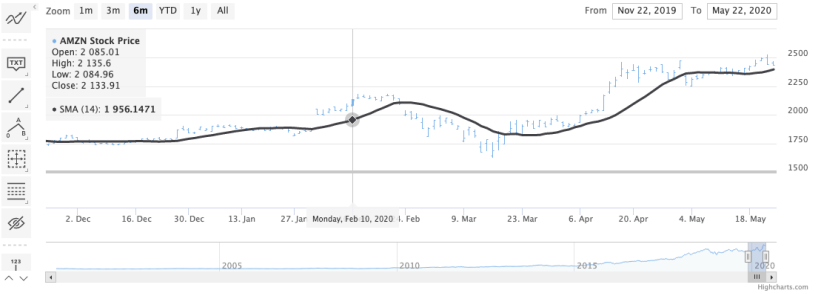
Amazon.com, Inc. engages in the retail sale of consumer products and subscriptions in North America and internationally. The company operates through three segments: North America, International, and Amazon Web Services (AWS). It sells merchandise and content purchased for resale from third-party sellers through physical and online stores. The company also manufactures and sells electronic devices, including Kindle, Fire tablets, Fire TVs, Rings, and Echo and other devices; provides Kindle Direct Publishing, an online service that allows independent authors and publishers to make their books available in the Kindle Store; and develops and produces media content. In addition, it offers programs that enable sellers to sell their products on its Websites, as well as its stores; and programs that allow authors, musicians, filmmakers, skill and app developers, and others to publish and sell content. Further, the company provides compute, storage, database, and other AWS services, as well as fulfillment, advertising, publishing, and digital content subscriptions. Additionally, it offers Amazon Prime, a membership program, which provides free shipping of various items; access to streaming of movies and TV episodes; and other services. It serves consumers, sellers, developers, enterprises, and content creators. The company was founded in 1994 and is headquartered in Seattle, Washington.

Company Information

Sector	Consumer Cyclical
City	Seattle
Country	United States

2. OHLC Graph

Open, High, Low and Close price graph of that stock with graph data ranging from 1 month to maximum time period. Indicators like SMA, Aroon Oscillator can be added in the graph by clicking on topmost right button, among many other features of the graph.



3. Financials

This section displays information of how the company has allocated its assets across various domains.

4. Historical Data

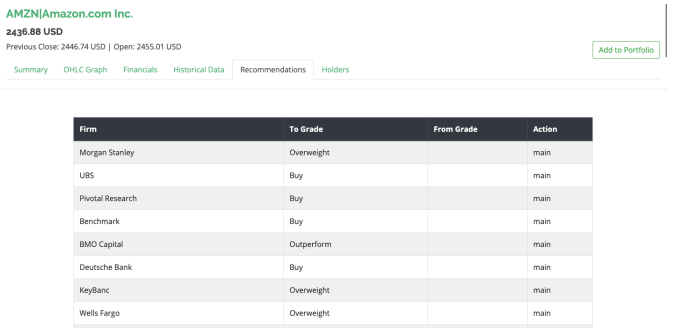
In this section, user can view historical data of the stock ranging from 1 day to maximum time period and can download that data with just a simple click on ***Download Data in CSV Format.***

1 Month data

Date	Open	High	Low	Close	Volume
2020-05-22	2455.010010	2469.850098	2430.129883	2436.879883	2860900
2020-05-21	2500.000000	2525.449951	2442.540039	2446.739990	5114400
2020-05-20	2477.870117	2500.010010	2467.270020	2497.939941	3998100
2020-05-19	2429.830078	2485.000000	2428.969971	2449.330078	4320500
2020-05-18	2404.350098	2433.000000	2384.010010	2426.260010	4366600

5. Recommendations

This section lists the recommendations of top financial institutions regarding the actions about the stock.



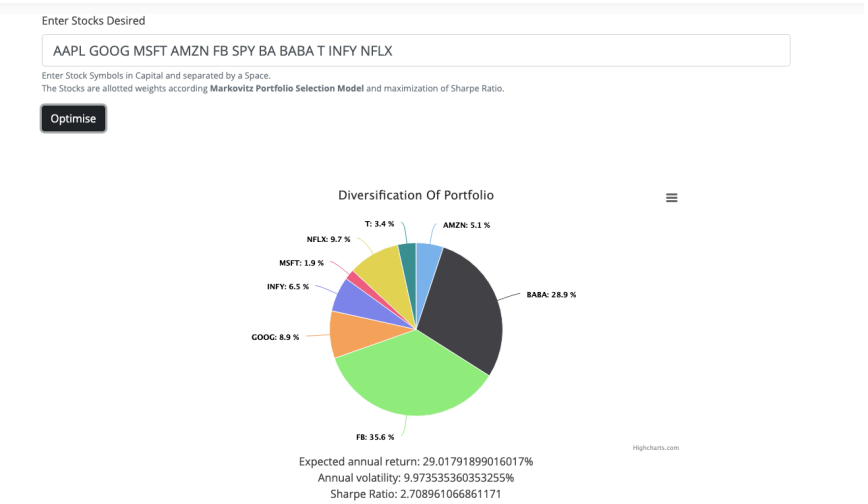
Firm	To Grade	From Grade	Action
Morgan Stanley	Overweight		main
UBS	Buy		main
Pivotal Research	Buy		main
Benchmark	Buy		main
BMO Capital	Outperform		main
Deutsche Bank	Buy		main
KeyBanc	Overweight		main
Wells Fargo	Overweight		main

6. Holders

This section lists the major and institutional holders of the company stocks.

PORTFOLIO OPTIMISATION

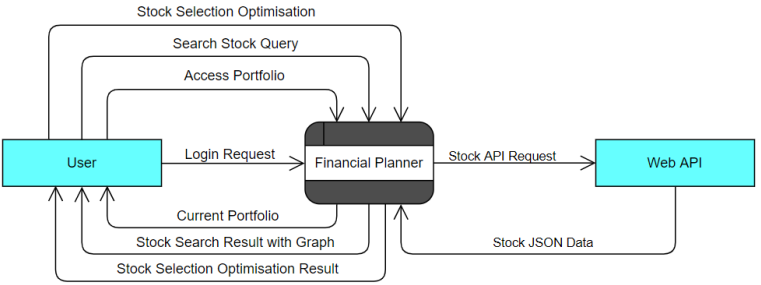
This feature allows user to use Markowitz Portfolio Selection Model based on mean-covariance optimisation to maximise the Sharpe-ratio of their respective portfolios. For further details refer to the Implementation Details Section.



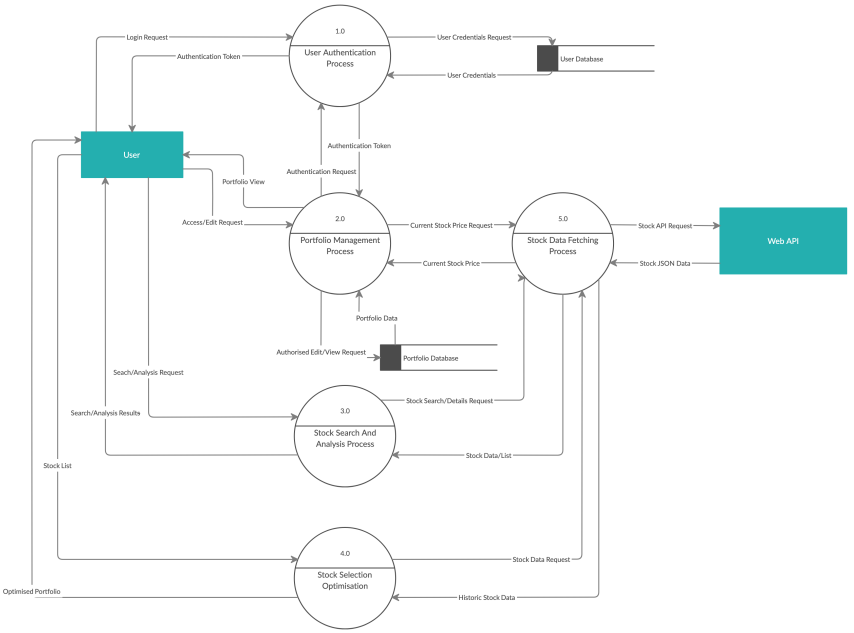
IMPLEMENTATION DETAILS

LOW LEVEL DESIGN

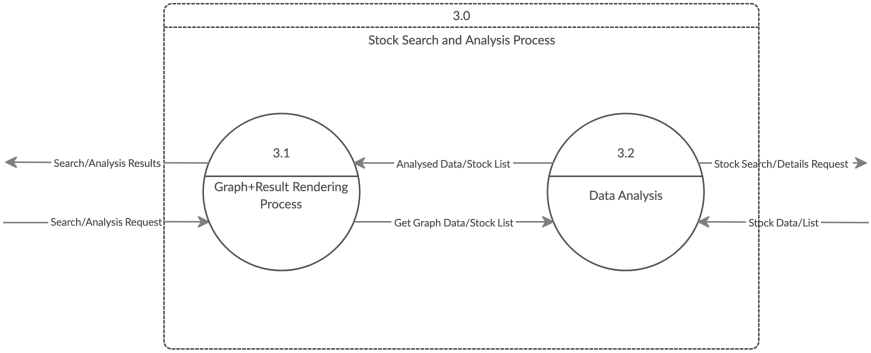
1. Level-0 DFD:



2. Level-1 DFD:



3. Level-2 DFD:



DATABASE DESIGN

1. Stock Table Schema

Stocks

ID	Stock Database
NAME	Stock Details
ALIAS	Securities/ Shares Details
FILE TYPE	Computerised
FILE FORMAT	Database Table
MAXIMUM RECORDS	10,000
TABLE NAME	Stocks
PRIMARY KEY	Ticker Symbol, User
SECONDARY KEYS	Name, Quantity Owned, Purchasing Price, Comments.
DESCRIPTION	Stores the Stock Information for each stock in the portfolio.

2. User Table Schema:

Users

ID	Users Database
NAME	User Details
ALIAS	-
FILE TYPE	Computerised
FILE FORMAT	Database Table
MAXIMUM RECORDS	10,000
TABLE NAME	Users
PRIMARY KEY	UserName
SECONDARY KEYS	Password, Email_Id
DESCRIPTION	Stores site users login data.

DJANGO BACKEND IMPLEMENTATION

Django is a high-level Python Web framework that encourages rapid development and clean, pragmatic design. Built by experienced developers, it takes care of much of the hassle of Web development, so you can focus on writing your app without needing to reinvent the wheel. It's free and open source and ridiculously fast.

The entire web-app Financial Planner was implemented using Django.

The Django web framework includes a default object-relational mapping layer (ORM) that can be used to interact with application data from various relational databases such as SQLite, PostgreSQL and MySQL. The Django ORM is an implementation of the object-relational mapping (ORM) concept. For database queries and management Django ORM was used in integration to PostgreSQL to handle the web-app database. Each Table in Database is called a model in Django. Our Stock Model implementation is shown below:

```python
from django.db import models
from django.utils import timezone
from django.contrib.auth.models import User
from django.urls import reverse
from .alpha_vantage_interaction import curr_p
class Stock(models.Model):
    WATCHLIST = 'W'
    MAIN_PORTFOLIO = 'P'
    ENTITY_CHOICE = [("WATCHLIST", WATCHLIST),("MAIN_PORTFOLIO",MAIN_PORTFOLIO)]
    symbol = models.CharField(max_length=10)
    name = models.TextField()
    date_posted = models.DateTimeField(default=timezone.now)
    quantity_owned = models.IntegerField(default = 0)
    initial_price = models.DecimalField(max_digits=10,decimal_places=3,default = 0.0)
    choice = models.CharField(max_length = 20, choices = ENTITY_CHOICE)
    user = models.ForeignKey(User, on_delete=models.CASCADE)
    def get_absolute_url(self):
        return reverse('financial_planner-portfolio-list')
    @property
    def curr_price(self):
        f = curr_p(self.symbol)
        return float(f)
```

Different urls of websites were mapped by Django backend to their respective views and the web-pages were rendered using Django Template rendering system.

```python
from django.urls import path
from . import views
from .views import (
    PortfolioListView,
    StockDetailView,
    StockUpdateView,
    StockDeleteView
)
urlpatterns = [
    path('', views.home, name = 'financial_planner-home'),
    path('analysis/<str:symbol>/<str:name>', views.stock_analysis_page, name = 'financial_planner-home'),
    path('portfolio/', PortfolioListView.as_view(), name = 'financial_planner-portfolio-list'),
    path('fsearch/', views.fund_search, name = 'financial_planner-fund-search'),
    path('port_opt/', views.recommend, name = 'portfolio_opt'),
    path('optimise/', views.port_opt, name = 'ajax'),
    path('stock/<int:pk>/', StockDetailView.as_view(), name = 'stock-detail'),
    path('about/', views.about, name = 'financial_planner-about'),
    path('search/', views.stock_search, name = 'financial_planner-stock-search'),
    path('gethistory/<str:symbol>/<str:period>/<str:interval>',views.get_history,name = 'history'),
    path('stock/<str:symbol>/<str:name>/<str:choice>',views.stock_create,name = 'stock-create'),
    path('stock/<int:pk>/update/', StockUpdateView.as_view(), name = 'stock-update'),
    path('stock/<int:pk>/delete/', StockDeleteView.as_view(), name = 'stock-delete'),

]
```

URLs for the Web-App as used in Django

DATA FETCHING

The stock data is fetched from various open-source sites like **Alpha Vantage**, **Finnhub**, **Yahoo Finance**, etc. Python requests library was used to send get requests to the respective servers to fetch the stock data. The data returned

from these servers was in JSON format. One such API Request response is depicted below:


```
{
    "Global Quote": {
        "01. symbol": "IBM",
        "02. open": "121.3000",
        "03. high": "122.3500",
        "04. low": "121.0101",
        "05. price": "122.2900",
        "06. volume": "1222602",
        "07. latest trading day": "2020-05-26",
        "08. previous close": "118.3900",
        "09. change": "3.9000",
        "10. change percent": "3.2942%"
    }
}
```


USER INTERFACE

The user interface was designed using HTML5, CSS, JavaScript and Bootstrap. Various javascript events like keyup for stock search, were used to make the website more responsive. The graphs in the website are rendered using another JavaScript library Highcharts.

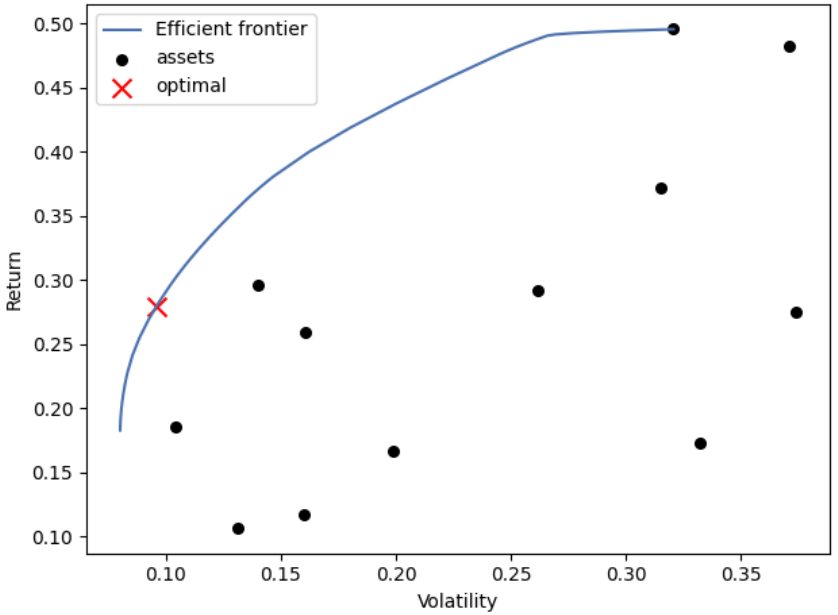
PORTFOLIO OPTIMISATION

As mentioned earlier, portfolio, that is, the group/ compilation of assets(stocks in this case) owned by a person, can be optimised using Markowitz Portfolio Selection Model. This model is based on **Modern Portfolio Theory,** which states,

"A good portfolio is more than a long list of good stocks and bonds. It is a balanced whole, providing the investor with protections and opportunities with respect to a wide range of contingencies." – Harry Markowitz

MPT shows that an investor can construct a portfolio of multiple assets that will maximise returns for a given level of risk. Likewise, given a desired level of expected return, an investor can construct a portfolio with the lowest possible risk. Based on statistical measures such as variance and correlation, an individual investment's return is less important than how the investment behaves in the context of the entire portfolio.

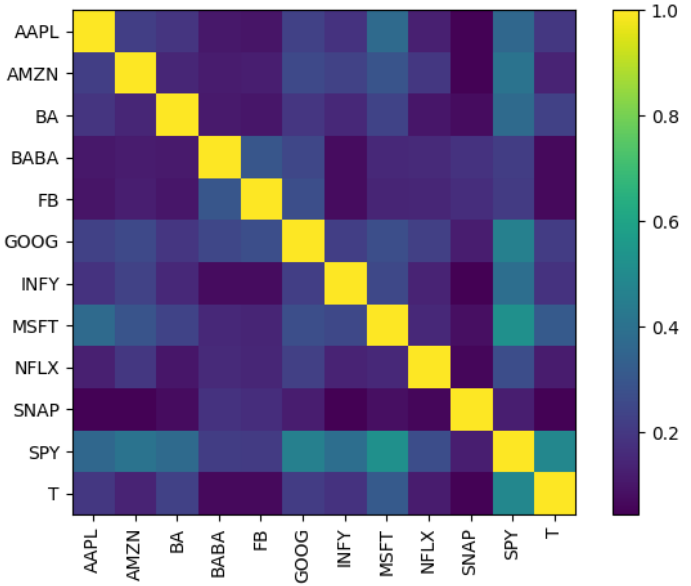
Every possible combination of assets that exists can be plotted on a graph, with the portfolio's risk on the X-axis and the expected return on the Y-axis. This plot reveals the most desirable portfolios. For example, assume Portfolio A has an expected return of 8.5% and a standard deviation of 8%, and that Portfolio B has an expected return of 8.5% and a standard deviation of 9.5%. Portfolio A would be deemed more "efficient" because it has the same expected return but lower risk. It is possible to draw an upward sloping hyperbola to connect all of the most efficient portfolios, and this is known as the **efficient frontier**. Investing in any portfolio not on this curve is not desirable.



Efficient Frontier
(AAPL MSFT AMZN INFY SPY NFLX T SNAP BA BABA FB GOOG)

The above graph shows the efficient frontier for stocks AAPL, MSFT, AMZN, INFY, SPY, NFLX, T, SNAP, BA, BABA, FB, GOOG. For obtaining this frontier mean-covariance optimisation was used through scikit-learn library of Python.

Efficient Frontier calculation requires two parameters, namely, expected returns and covariance matrix for the stocks. The expected returns for each stock is assumed to be the mean of the historical return to the latest day while instead of building normal covariance matrix for the stocks a Ledoit Wolf Covariance Shrinkage Matrix is used to provide faster and optimal convergence to the efficient frontier algorithm. The Covariance Shrinkage Matrix for the above efficient frontier is shown below:



When you analyse a set of assets using mean-variance analysis, the **tangency portfolio** is the **portfolio** with the highest **Sharpe ratio**. t's called the **tangency** because it's located at the **tangency** point of the Efficient Frontier The optimal point marked by a cross '**X**' is the portfolio with highest **Sharpe Ratio.** The Sharpe ratio is the average return earned in excess of the risk-free rate per unit of volatility or total risk. It is calculated as follows:

Formula and Calculation of Sharpe Ratio

$$Sharpe\ Ratio = \frac{R_p - R_f}{\sigma_p}$$

where:

R_p = return of portfolio

R_f = risk-free rate

σ_p = standard deviation of the portfolio's excess return

The higher a fund's Sharpe ratio, the better its returns have been relative to the amount of investment risk it has taken.

CONCLUSION

With this project we hope to fight the financial illiteracy in our country and ease out the apprehension among people against Personal Investments in the Stock Market. This project is our effort to demystify the stock market. This project can have wide scale applications in personal finance and will help investors make smart decisions about their money.

Please check out finplanapp.herokuapp.com and be one step closer towards financial independence.

REFERENCES

LEARNING RESOURCES

1. Django Tutorials————————-Corey Schafer, YouTube
2. HTML, CSS, JavaScript———— www.w3schools.com
3. Bootstrap4————————————-getbootstrap.com
4. Highcharts.js————————————www.highcharts.com/docs/index
5. Markowitz Model———————————www.math.ust.hk/~maykwok/courses/ma362/07F/markowitz_JF.pdf
6. Modern Portfolio Theory—-—--www.investopedia.com/terms/m/modernportfoliotheory.asp
7. Sharpe Ratio————————————www.investopedia.com/terms/s/sharperatio.asp

DATA SOURCES

1. Alpha Vantage API————————-www.alphavantage.co
2. FinnHub————————————www.finnhub.io
3. Yahoo Finance————————————in.finance.yahoo.com