

Indian Stock Market Simulator

Minor project report submitted in partial fulfilment of the requirement for the degree of
Bachelor of Technology

in

Computer Science and Engineering

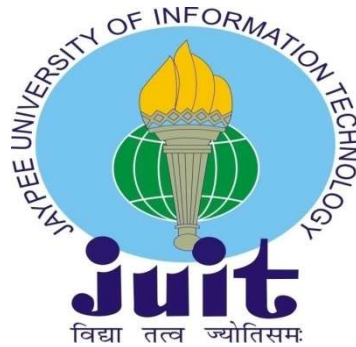
By

Shubham Sharma (181434)

Apoorav Chaudhary (181371)

UNDER THE SUPERVISION OF

Mr. Prateek Thakral



Department of Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Wagnaghat, 173234, Himachal Pradesh, INDIA

TABLE OF CONTENT

Content	Page No.
Declaration by Candidate	I
Certificate by Supervisor	II
AKCNOLEDGEMENT	III
Abstract	IV
1. Introduction	1-7
1.1 Introduction	1
1.2 Objective	2
1.3 Motivation	3
1.4 Languages Used	4,5
1.5 Technical Requirements (Hardware)	6
1.6 Deliverables	7
2. Minor Project SDLC.....	8-17
2.1 Feasibility Study	8
2.2 Functional Requirement.....	9
2.3 Non- Functional Requirement.....	10,11
2.4 Use-Case Diagram.....	12,13
2.5 DFD Diagram.....	14,15,16
2.6 State Transition Diagram.....	17
3. Implementation of Minor Project	18-38
3.1 Data Set Used.....	18,19
3.2 Types of Data Set.....	20
3.3 No. of Attributes, fields, description of data set.....	21,22
3.4 Design of problem Statement.....	23
3.5 Algorithm Used.....	24,25,26,27
3.6 Flowchart.....	28,29,30,31
3.7 Screenshots.....	32-38

4. Results.....	39-42
4.1 Discussions on Result Achieved.....	39
4.2 Applications of the Minor Project.....	40
4.3 Limitations of the project.....	41
4.4 Future Work.....	42
5. References.....	43

DECLARATION

I hereby declare that, this project has been done by me under the supervision of **Mr. Prateek Thakral**, Jaypee University of Information Technology. I also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:

Mr. Prateek Thakral

Assistant Professor

Department of Computer Science & Engineering and Information Technology

Jaypee University of Information Technology

Submitted by:

Apoorav Chaudhary (181371)

Shubham Sharma (181434)

Computer Science & Engineering Department

Jaypee University of Information Technology

CERTIFICATE

This is to certify that the work which is being presented in the project report titled “**Indian Stock Market Simulator**” in partial fulfilment of the requirements for the award of the degree of B.Tech in Computer Science And Engineering and submitted to the Department of Computer Science And Engineering, Jaypee University of Information Technology, Waknaghat, is an authentic record of work carried out by “Apoorav Chaudhary (181371), Shubham Sharma (181434)” during the period from January 2021 to May 2021 under the supervision of **Mr. Prateek Thakral**, Department of Computer Science and Engineering, Jaypee University of Information Technology, Waknaghat.

Apoorav Chaudhary (181371)
Shubham Sharma (181434)

The above statement made is correct to the best of my knowledge.

Mr. Prateek Thakral
Assistant Professor
Computer Science & Engineering and Information Technology
Jaypee University of Information Technology, Waknaghat,

ACKNOWLEDGEMENT

Firstly, we express our heartiest thanks and gratefulness to almighty God for His divine blessing that made it possible to complete the project work successfully.

We really are grateful and wish our profound indebtedness to Supervisor **Mr Prateek Thakral, Assistant Professor**, Department of CSE Jaypee University of Information Technology, Wakhnaghat. Deep Knowledge & keen interest of my supervisor in the field of “**Web Development and Computer Networks**” to carry out this project. His endless patience, scholarly guidance, continual encouragement, constant and energetic supervision, constructive criticism, valuable advice, reading many inferior drafts and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Mr Prateek Thakral**, Department of CSE, for his kind help to finish our project.

We would also generously welcome each one of those individuals who have helped us straight forwardly or in a roundabout way in making this project a win. In this unique situation, we might want to thank the various staff individuals, both educating and non-instructing, which have developed their convenient help and facilitated our undertaking.

Finally, we must acknowledge with due respect the constant support and patience of my parents.

Apoorav Chaudhary (181371)
Shubham Sharma (181434)

ABSTRACT

Our project “**Indian Stock Market Simulator**” is based upon **Web Development and Computer Networking**. In the web development part, we have created a Website/ Web Interface using web technologies, **php, HTML, CSS, Bootstrap, JavaScript**, for storing data we have used **MySQL Database** and to **Apache Server** for hosting the website locally.

Indian Stock Market Simulator is a website in which, users can create their account free of cost and doesn't need to add their real money. Whenever a user creates an account, he/she will be allotted ₹20,00,000 worth of virtual money which could only be used to practice trading, to be precise, this amount isn't real money. Our interface isn't connected to any stock exchange, so user won't be trading actual stock, just practicing as if they are, although stock prices are fetched using python in real time and will replicate the original price of the stock in real time. After creating an account, user can log in, after logging in, user can put order to buy a stock at certain price among NIFTY 50 companies, when the cost of the stock become less than the order price, stock's buy operation will be executed, after the stock is successfully bought, user can put an order to sell the stock at a certain price, LTPs of the Nifty 50 companies can be seen in watchlist, and their charts using the chart button, Profit and Loss can also be seen in Positions section.

As for the **Computer Networks** part we have created a Local DNS, a local Domain Name Server which will be able to resolve the Domain Names into the IP addresses, in this local network. Using Network Switch configuration, we configured our nodes to use our local DNS as the primary DNS of each node. So, when any node enters a domain name in their browser the request will be resolved by our local DNS. We have used **Bind9 DNS**.

INTRODUCTION

Our project **“Indian Stock Market Simulator”** is based upon **Web Development and Computer Networking**. Stock market is a place where people buy/sell shares of publicly listed companies. It offers a platform to facilitate seamless exchange of shares. In simple terms, if A wants to sell shares of Reliance Industries, the stock market will help him to meet the seller who is willing to buy Reliance Industries. However, it is important to note that a person can trade in the stock market only through a registered intermediary known as a stock broker. The buying and selling of shares take place through electronic medium. There are a lot of individuals and corporate houses who trade in a stock market. Anyone who buys/sells shares in a stock market are termed as a market participant.

The main goal of our minor project is to provide an interface for individuals to utilize in order to teach themselves in the stock market without spending real money. So that even if they join a loss transaction, they do not lose any actual money. Because the stock market is unpredictable, the likelihood of entering a loss transaction is relatively high for newcomers. When people lose their real money, they lose drive and morale, and they stop trading. This website is intended to allow users to exercise stock market trading without using real money. Practice makes a man perfect, when people will be able to practice the Stock Market, they will learn more and more about the Stock Market.

This project was designed for people who wish to learn about the stock market, but it may also be used by businesses or franchises to teach their personnel. Employees at franchises or stock brokers construct trades depending on numerous factors, then recommend those transactions to their customers; sometimes those trades are successful, and sometimes they are not. However, educating those people is essential. We have presented the notion of computer networking in this section. Organizations frequently have their own network within the firm, and they like to keep their network secure and private from the outside world, therefore we established the notion of Local Networking by utilizing technologies such as DHCP and DNS.

OBJECTIVE

Our project's major purpose is to create an interface for anyone to use in order to teach themselves about the stock market without spending actual money. As a result, even if they participate in a losing trade, they will not lose any actual money. Because the stock market is unpredictable, beginners have a relatively high risk of joining a loss transaction. People lose motivation and morale when they lose actual money, and they quit trading. This website is designed to allow users to practice stock market trading without having to use real money. People will not lose money since they will not be utilizing actual money.

This project was created for those who want to learn about the stock market, but it may also be used to train employees at firms or franchises. Employees at franchises or stock brokers build deals based on a variety of parameters and then propose them to their consumers; sometimes such trades are profitable, and sometimes they are not. However, teaching such individuals is critical. In this part, we introduced the concept of computer networking.

To recap, the goal here is to teach individuals in the stock market so that they do not lose money and cease trading as a result. If more and more people engage in stock market trading, the nation's GDP will benefit as the stock market rises and foreign investors make more investments in the Indian market.

MOTIVATION

In the past year we have seen that during these unfortunate time, people have been looking for some alternate methods to earn money, and as a result more and more people started Stock market trading, as the condition of the country wasn't good, stock market was also crashing, people were indulging in the stock market when the market was crashing and a result, people lost money, and quit trading, but some people also made money as the volatility in the market was high, but in the near future, things are bound to get better, stock market will also become less volatile. If that happens, people are more likely to start trading, keeping that in mind we are motivated to do something which can help people.

Motivation behind this project is also somewhat personal, as I also started trading last year, I lost money in the beginning, if only I had trained first before putting my actual money, I wouldn't have lost so much money. And that's how I came up with the idea of stock market simulator. The idea that it could be used up by franchises was realized later which led to introduction of concepts of computer networking into the project, to make this project LAN oriented.

And as any other project, main motivation behind this project was to learn and implement something which could be real world application.

LANGUAGES USED

Implementation of this project is achieved with the help of various technologies. Web Development technologies like, **HTML, CSS, Bootstrap, JavaScript, php, MySQL, python.**

- **HTML:** The Hypertext Markup Language, or HTML is the standard markup language for documents designed to be displayed in a web browser. Web browsers receive HTML documents from a web server or from local storage and render the documents into multimedia web pages. HTML describes the structure of a web page semantically and originally included cues for the appearance of the document.
- **CSS:** CSS is the language we use to style an HTML document. CSS describes how HTML elements should be displayed. CSS is the language for describing the presentation of Web pages, including colors, layout, and fonts. It allows one to adapt the presentation to different types of devices, such as large screens, small screens, or printers. CSS is independent of HTML and can be **used with** any XML-based markup language.
- **Bootstrap:** Bootstrap is a free and open-source CSS framework directed at responsive, mobile-first front-end web development. It contains CSS- and JavaScript-based design templates for typography, forms, buttons, navigation, and other interface components.
- **JavaScript:** JavaScript is *an object-based scripting language* which is lightweight and cross-platform. JavaScript is not a compiled language, but it is a translated language. The JavaScript Translator (embedded in the browser) is responsible for translating the JavaScript code for the web browser. JavaScript (js) is a light-weight object-oriented programming language which is used by several websites for scripting the webpages.

- **php:** The **PHP Hypertext Preprocessor (PHP)** is a programming language that allows web developers to create dynamic content that interacts with databases. PHP is basically used for developing web-based software applications. This tutorial helps you to build your base with PHP.
- **MySQL:** MySQL is the most popular Open-Source Relational SQL Database Management System. MySQL is one of the best RDBMS being used for developing various web-based software applications. MySQL is developed, marketed and supported by MySQL AB, which is a Swedish company.
- **Python:** Python is a high-level, interpreted, interactive and object-oriented scripting language. Python is designed to be highly readable. It uses English keywords frequently where as other languages use punctuation, and it has fewer syntactical constructions than other languages.

Other technologies involved are:

- **BIND9:** Domain Name Service (DNS) is an Internet service that maps IP addresses and fully qualified domain names (FQDN) to one another. In this way, DNS alleviates the need to remember IP addresses. Computers that run DNS are called name servers. Ubuntu ships with BIND (Berkley Internet Naming Daemon), the most common program used for maintaining a name server on Linux.

TECHNICAL REQUIREMENTS (Hardware)

Completion of any project without hardware requirements is impossible, following are the hardware we have used for completion of the project:

- **Network switch:** A network switch is a device that operates at the Data Link layer of the OSI model—Layer 2. It takes in packets being sent by devices that are connected to its physical ports and sends them out again, but only through the ports that lead to the devices the packets are intended to reach. Once a device is connected to a switch, the switch notes its media access control (MAC) address, a code that's baked into the device's network-interface card (NIC) that attaches to an ethernet cable that attaches to the switch. The switch uses the MAC address to identify which attached device outgoing packets are being sent from and where to deliver incoming packets. So, the MAC address identifies the physical device as opposed to the network layer (Layer 3) IP address, which can be assigned dynamically to a device and change over time.
- **Category 6 cable (Cat 6):** **Category 6 cable (Cat 6)**, is a standardized twisted pair cable for Ethernet and other network physical layers that is backward compatible with the Category 5/5e and Category 3 cable standards. A **Cat6 cable** is used mainly for computer networks reaching a GB, 1000 Mbps or one Gbps of data transfer speed (DTR) or higher.
- Other hardware requirement includes Client and Server computers.

Deliverables of the Minor Project

Final outcome of our project is as following:

- We were successfully able to create a Domain Name Server, which can resolve the Domain Name into the IP Address whenever it receives a request from the client.
- DHCP is successfully configured, whenever a new node is connected to the network, it is assigned an IP address from the defined range.
- Apache web server is configured successfully, website is being hosted successfully.
- Website/Web Interface has been created successfully.
- LTP (Latest Trade Price) are being fetched from the NSE (National Stock Exchange).
- User accounts can be created successfully.
- Users can log in to their account.
- Users can successfully take trades; both Buy and Sell.
- Users can see the status of their order in the orders section.
- When a buy trade is executed, Profit and Loss can be seen in the positions section.
- When the web server is running, client computer can access this website using Domain Name.

Feasibility Study on Minor Project

- **Technical feasibility:** Technical feasibility states if the project is achievable with the current technology available. In our case, yes, the project is achievable with the current technology available. Be it, the hardware availability, software availability or the personnel required, this project is completely feasible.
- **Economic feasibility:** Economically, this project is completely feasible, as, all this project require is time and effort, alternates to hardware requirements are also easily available.
- **Operational feasibility:** This project does require hardware system, and hardware requirements also need regular maintenance. But at industrial level this project is completely feasible.
- **Social feasibility:** We have used Open-Source resources to create this project, so this project is socially feasible as well. The project stands above all the ethical requirements. The legal standards resolving around this type of project have already been covered. The project does not violate any copyright policy.

Functional Requirements

The primary functionality of this project is to fetch the data of the stock in real time and present it to each user. The project should be able to handle the user requests of creating an account, letting them log in and then see the stock prices in real time, users should also be able to put their orders, buy orders and sell orders, when any order is executed, they should also be able to see their order status and their Profit and Loss in real time. Stock price is fetched using python script using nsetools package. As for the networking part, user should be able to use domain name instead of the IP address of the web server. The requirement to run this project are: Computers with working internet connection, which could be made into Servers and Clients, Network Switch, Ethernet cables and as for the software requirements, HTML, CSS, JavaScript, Bootstrap, php, MySQL, Python, apache web server, Bind9 are required.

Non-Functional Requirements

Nonfunctional Requirements (NFRs) define system attributes such as security, reliability, performance, maintainability, scalability, and usability. They serve as constraints or restrictions on the design of the system across the different backlogs. They ensure the usability and effectiveness of the entire system.

- **Security:** Security in this context can be seen in two ways, can this website cause threat to the users? or, can the users be using the website harm the server in any way? The answer to these questions depends upon how the project is being used, if the project is being hosted on online server without the implementation of the DNS and the local network, the chances of any security breach is very low, whereas if the project is being implemented locally, the security might be breached from within the local network.
- **Reliability:** The reliability of this project highly depends upon various factors. Those factors are as follows: (1): Internet connection should be reliable as the stock prices are being fetched online. (2). Connectivity using Network Switch, CAT cables should be reliable as well. (3) Use of this project must be done as proposed. If the above parameters are met, this project is reliable.
- **Availability:** Availability of this project highly depends upon Internet availability which will ensure the availability of the Stock Prices. Other factors are also responsible for availability, like, electricity, availability of hardware.
- **Capacity:** The capacity here means storage capacity, the project should be able to store information about the users, as the user will register, the information will be stored in the database, the database should be enough to store a large amount of data.

- **Maintainability:** This project is highly maintainable, but not necessarily relied upon maintenance, as the technology is growing day by day, new features can be added into the project but it's not necessary to do so.
- **Scalability:** Scalability here means if there is a scope for the project to grow, well there is. With the growth in technology day by day, the project can grow immensely.
- **Data Integrity:** Integrity of the data in the database should be maintained at all cost, as the details like, Phone number and email address are sensitive and should be protected, in our project, we have ensured the data integrity.
- **Usability:** A project should always be user friendly, as in, users should be able to use the project without any complications, Interface of our project is designed in such a way, that the project is easy to use by even a non-technical person.
- **Platform Independence:** A project should be free of any platform restrictions. Our project is highly Platform Independence, it can be accessed on any platform, be it, Windows, Linux, Android.

Use Case Diagram

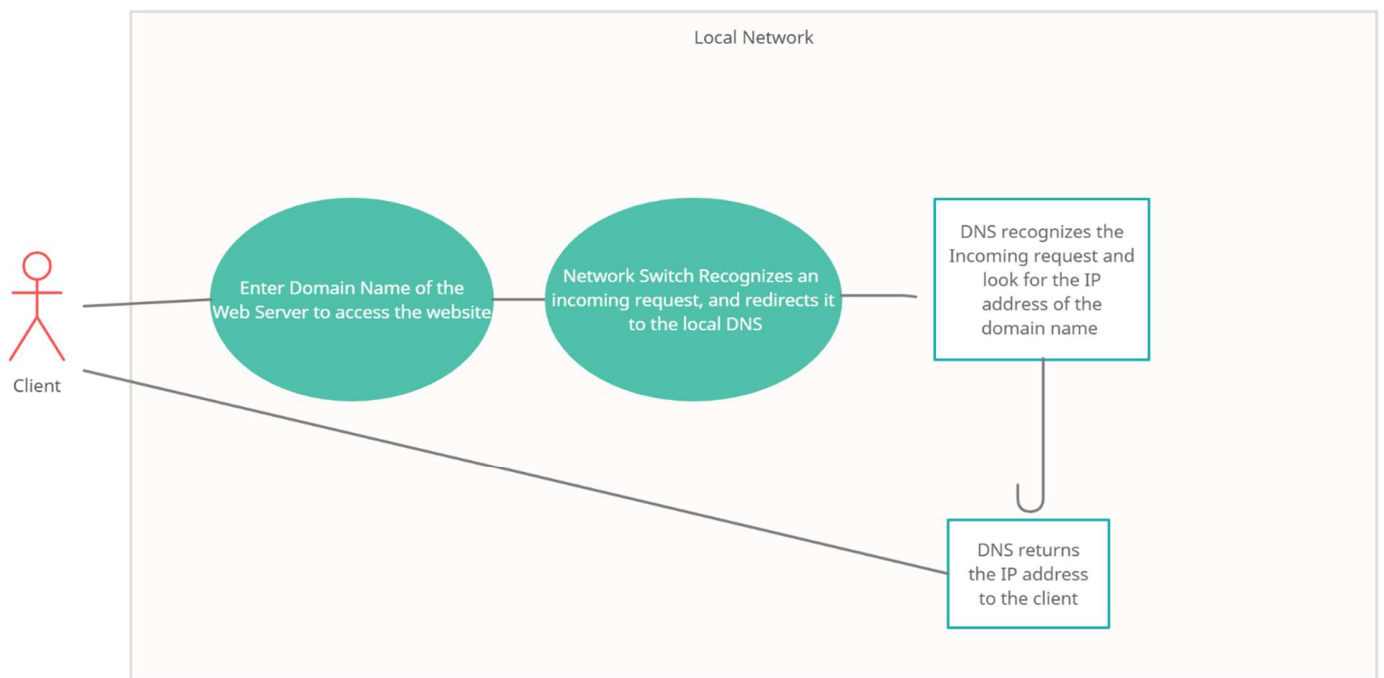


Fig 2.4.1- The above figure is a Use case diagram of how a DNS works.

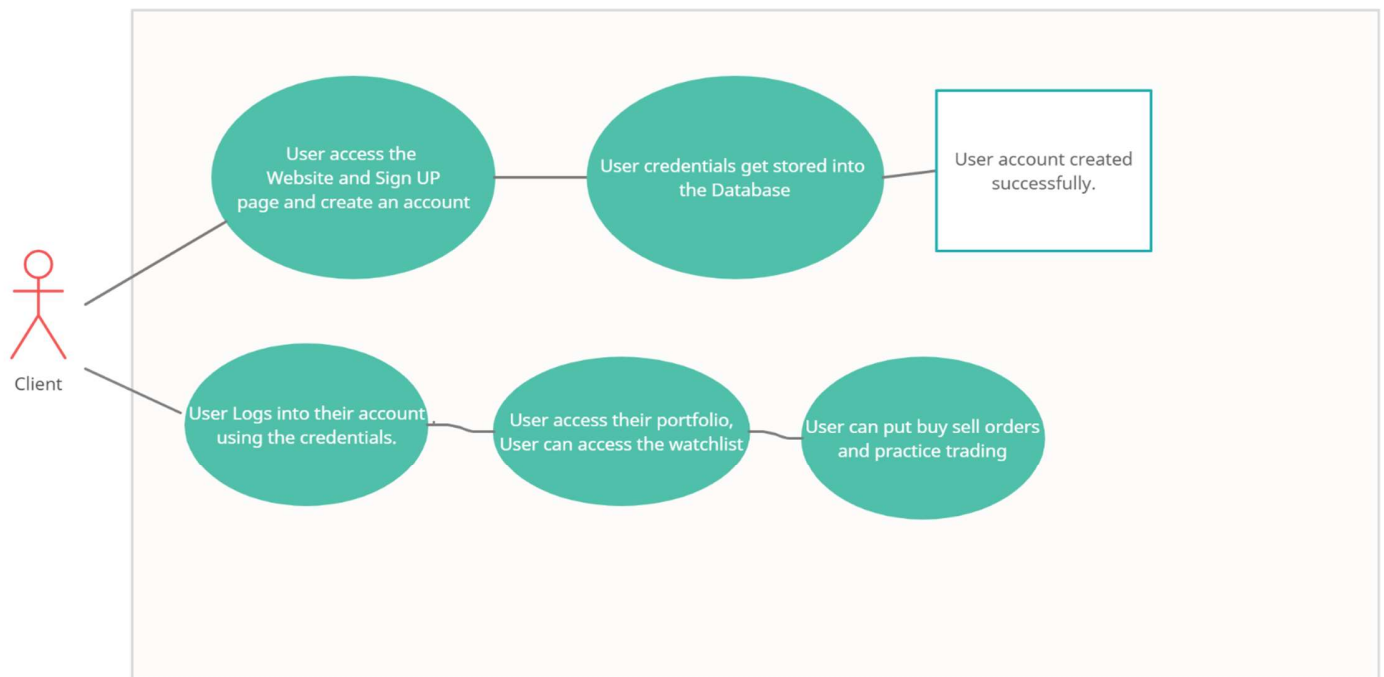


Fig 2.4.2- The above Use Case diagram show how the website works

DFD Diagram

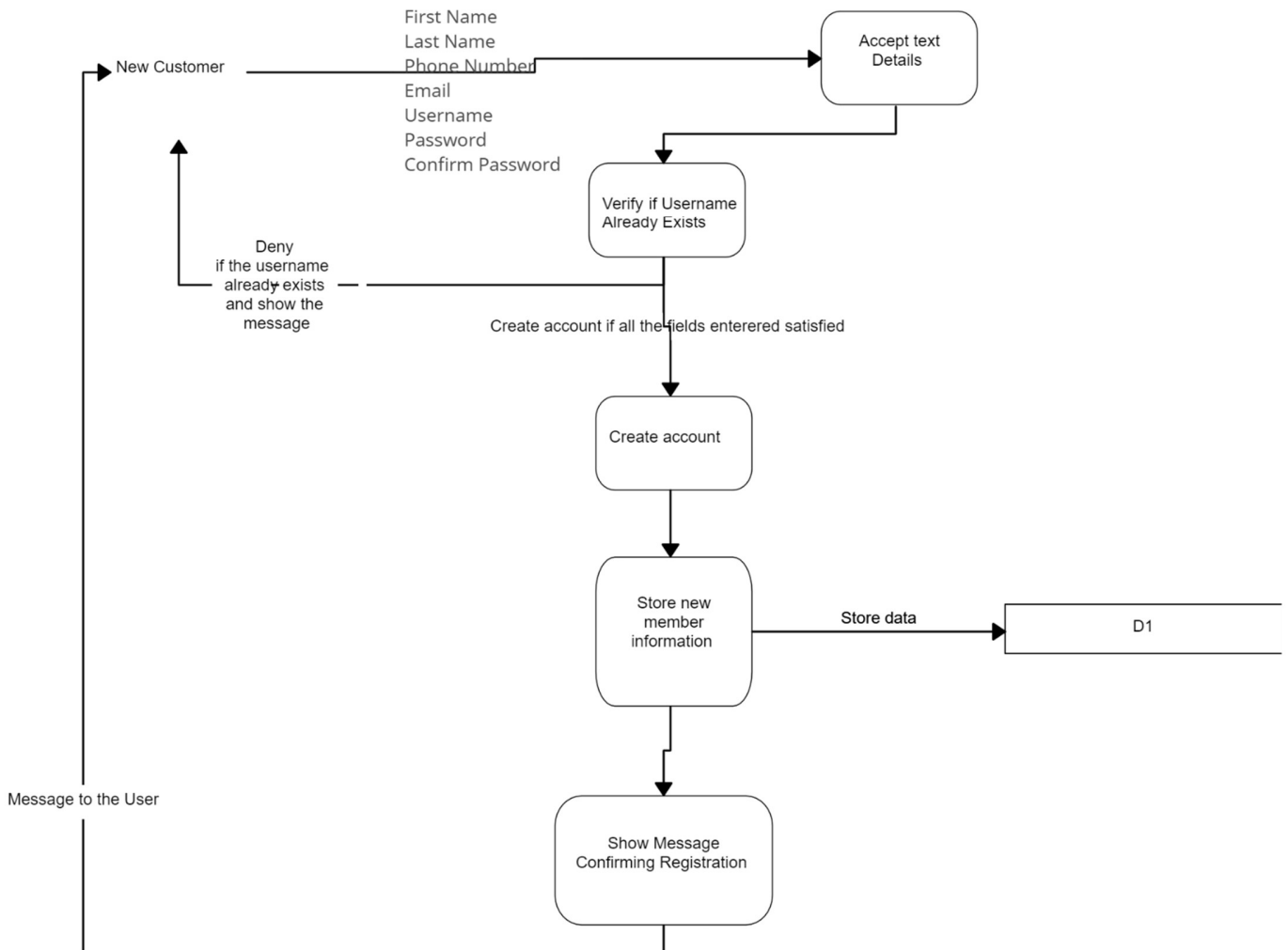


Fig 2.5.1- The above DFD diagram is of User registration

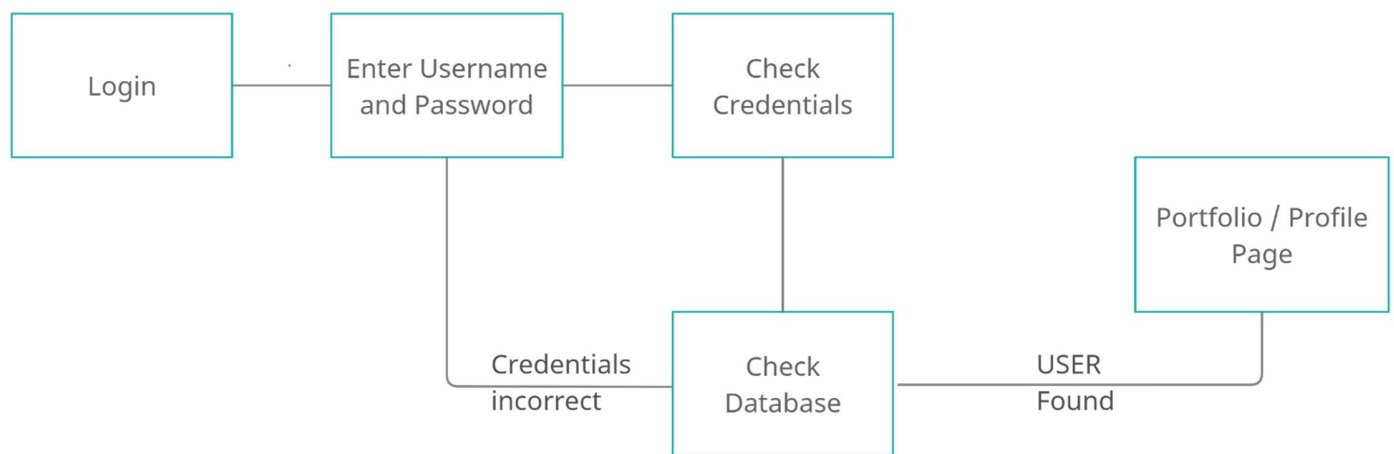


Fig 2.5.2- The above DFD diagram is of user login

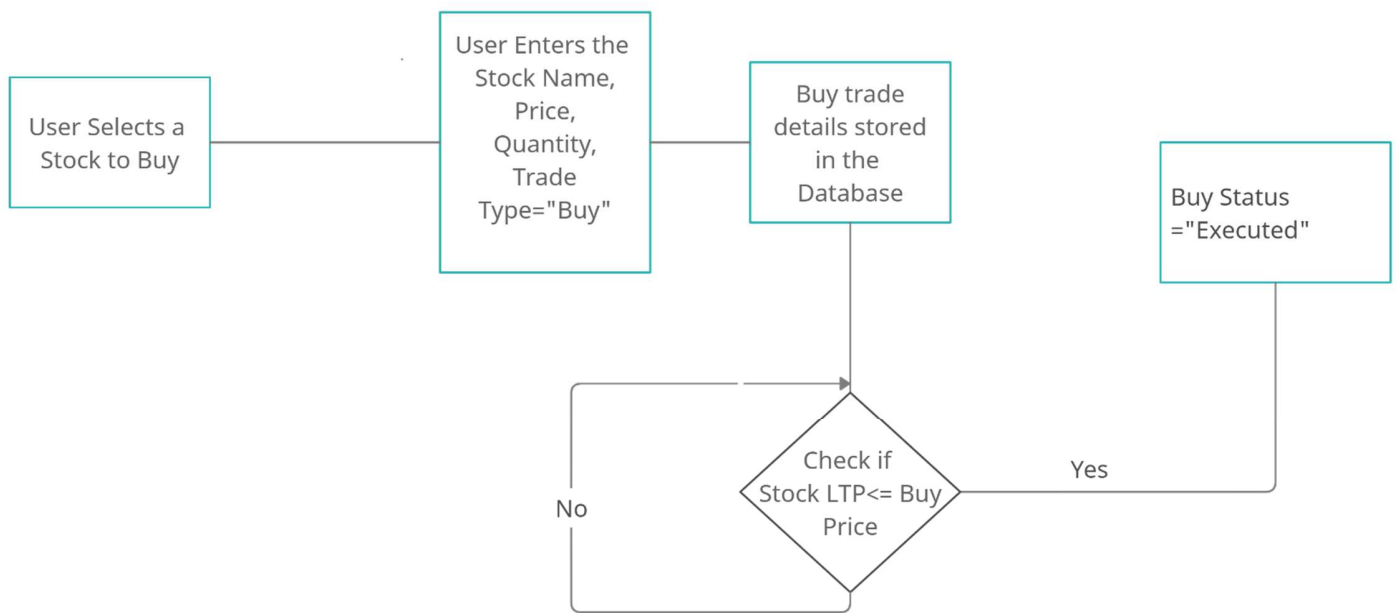


Fig 2.5.3- The above DFD is of Buy Trade

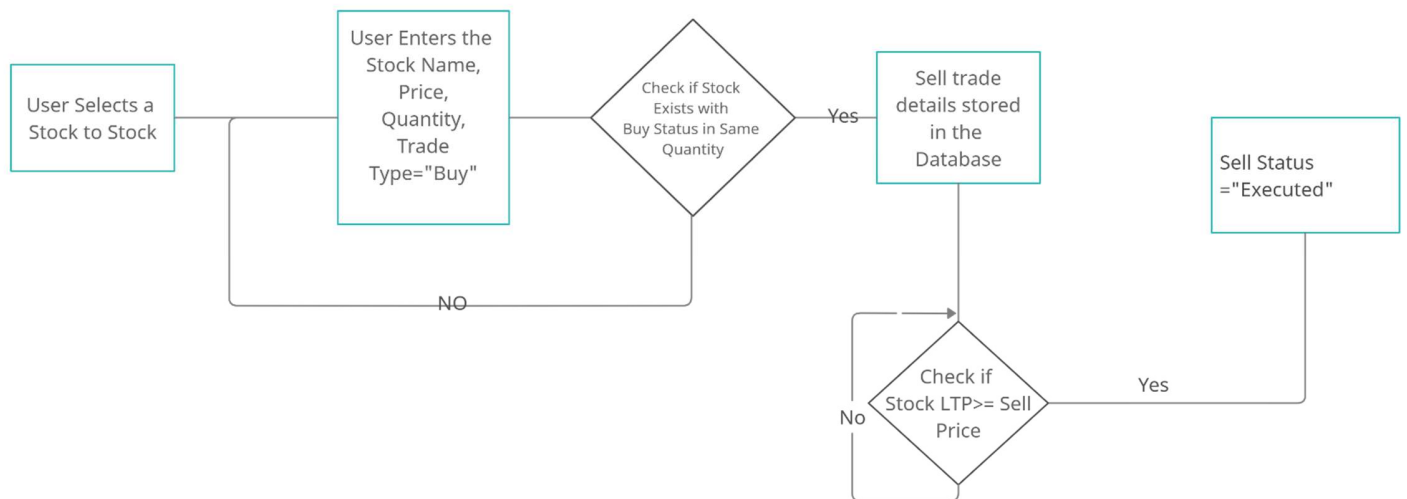


Fig 2.5.4- The above DFD is of Sell Trade

State Transition Diagram

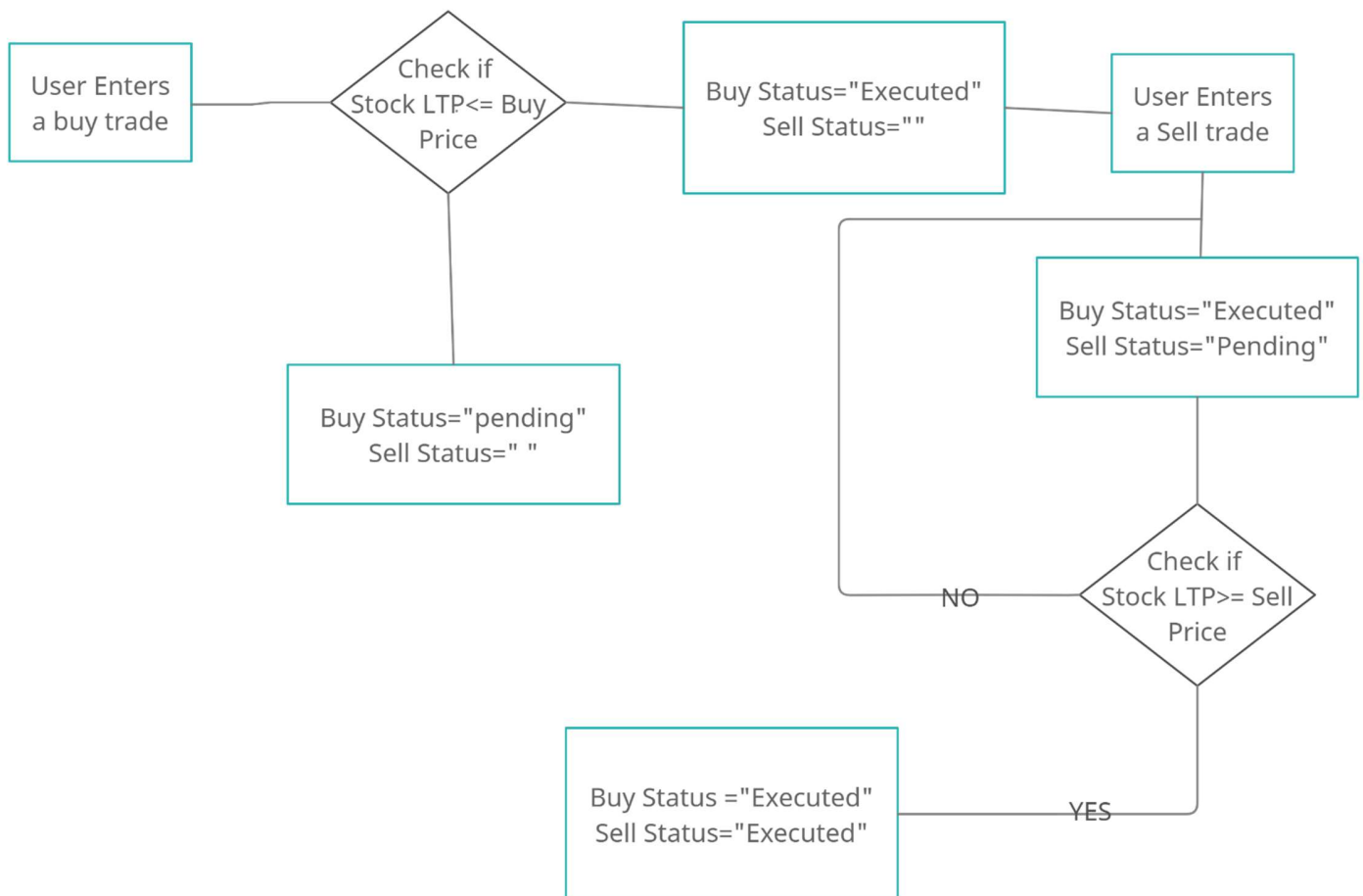


Fig 2.6.1- The above figure is a State Transition Diagram

Data Sets Used

#	Name	Type	Collation	Attributes	Null
1	Name	varchar(200)	utf8mb4_general_ci		No
2	Price	varchar(200)	utf8mb4_general_ci		No
3	Open	varchar(200)	utf8mb4_general_ci		No
4	High	varchar(200)	utf8mb4_general_ci		No
5	Low	varchar(200)	utf8mb4_general_ci		No

Fig 3.1.1- Above figure is Structure view of the data of each stock being fetched

#	Name	Type	Collation
1	Symbol	varchar(20)	utf8mb4_general_ci
2	BuyPrice	varchar(20)	utf8mb4_general_ci
3	SellPrice	varchar(20)	utf8mb4_general_ci
4	Quantity	int(20)	
5	Balance	varchar(30)	utf8mb4_general_ci
6	BuyStatus	varchar(20)	utf8mb4_general_ci
7	SellStatus	varchar(20)	utf8mb4_general_ci
8	TradeType	varchar(5)	utf8mb4_general_ci

Fig 3.1.2- Above figure is Structure view of the data of each user

#	Name	Type	Collation
1	FName	varchar(200)	utf8mb4_general_ci
2	LName	varchar(200)	utf8mb4_general_ci
3	EmailID	varchar(200)	utf8mb4_general_ci
4	Username	varchar(200)	utf8mb4_general_ci
5	Password	varchar(200)	utf8mb4_general_ci
6	MobileNo	varchar(200)	utf8mb4_general_ci
7	Balance	varchar(200)	utf8mb4_general_ci

Fig 3.1.3- Above figure is the Structure view of Registered Users

Name	Date modified	Type	Size
Python Scripts	10-05-2021 14:22	File folder	
SignUp	10-05-2021 14:25	File folder	
SQL	11-05-2021 21:51	File folder	
vector	10-05-2021 14:19	File folder	
HowItWorks.html	05-05-2021 22:25	Microsoft Edge Be...	3 KB
index.html	13-05-2021 12:43	Microsoft Edge Be...	4 KB
logo.png	03-05-2021 21:23	PNG File	3 KB
—Pngtree—2 5d financial stock market_4...	26-04-2021 21:12	PNG File	1,147 KB
—Pngtree—stock market fluctuation gra...	26-04-2021 22:17	PNG File	89 KB

Fig 3.1.4- Above figure is the Web Dataset of our Website

Types of Data Set

The type of data set we have used in our project is Database Dataset and Web Datasets.

A database dataset, as the name suggests, is a collection of data kept in a database. In general, each database is a separate dataset (however, to be precise, each user/schema within a database is a separate dataset). Each table in the database is handled as a feature type in a database dataset. An online dataset is a collection of data that is hosted on a website. In this scenario, the dataset's name is the same as the URL's name (Universal Resource Locator).

Number of Attributes, fields, description of the data set

➤ In **Stock** dataset, the number of attributes/fields are 5, namely:

1. Name: Stores the name of the Stock.
2. Price: Stores the current price i.e., LTP of the stock.
3. Open: Stores the opening price of the stock i.e., Price at 9:15 AM
4. High: Store a single day's highest price of the stock.
5. Low: Stores a single day's lowest price of the stock.

All in all, this dataset stores the information about the stock

➤ In **each user** dataset, the number of attributes/fields are 8, namely:

1. Symbol: Stores the name of the Symbol/Company, user has put order of.
2. BuyPrice: Stores the buy price user has set for each Symbol or trade.
3. SellPrice: Stores the sell price user has set for each 'buy executed' trade.
4. Quantity: Stores the quantity of the stock, user has bought.
5. Balance: Stores the Balance money user has.
6. BuyStatus: Stores the status of buy trade, possibly, pending or executed.
7. SellStatus: Stores the status of sell trade, possibly, "", pending or executed.
8. TradeType: Stores the current type of the trade a symbol is in.

➤ In **Registered_Users** dataset, the number of attributes/fields are 7, namely:

1. FName: Stores the first name of the users.
2. LName: Stores the last name of the users.
3. EmailID: Stores email ID of the users.
4. Username: Stores the username of the users.
5. Password: Stores the password of the users.
6. MobileNo: Stores the Mobile number of the users.
7. Balance: Stores the real time balance of each user, updated upon each trade.

Problem Statement

What Exactly Is a "Design Problem"? We've all experienced them, solved them, and very certainly caused them. However, putting it simply is a difficulty in and of itself. According to the Oxford Dictionary, a problem is “a matter or circumstance considered as unpleasant or unpleasant and necessitating dealing with and overcoming.” True, this implies that the desired consequence is known. An issue is more than just an unfavorable condition or an issue that deviates from the norm—though these are nonetheless legitimate definitions of an issue. An issue, in the context of designers and creative problem solving, is an unmet need that, if fulfilled, can fulfil the user's purpose.

The problem statement to our project can be defined as following; As we already know, due to the current condition of the country, economy isn't doing so well, and as a result people were and are being laid off from their jobs and as a result, we have seen a huge rise in unemployment. Many tried to earn their living through stock market without any prior knowledge of it. Some people actually made money but the number of people who lost money was way higher. And **the possible reason or the problem here was lack of knowledge or practice.** Which can be stated as the problem statement of our project, when we look more into this problem and tried to find a solution, there weren't many, as they required money or there were some other problems. So, we defined this as the main problem and sought out for a solution and tried to make one of our own, through this project.

Algorithms Used

➤ User Sign Up:

- Initiate apache and MySQL Server
- Connect MySQL to SignUp.php
- If (Sign up page connected to database) {
 - Enter User Details
 - Check validation of the entered credentials
 - If (validationCheck==" Valid") {
 - ✓ Upload data into the database
 - ✓ Show status "Account Created Successfully"
 - }else{
 - ✓ Show validation error
 - }
- }else{
 - Show connectivity error
- }

➤ User Sign In:

- Initiate apache and MySQL Server
- Connect MySQL to SignIn.php
- If (Sign in page connected to database) {
 - Enter User Credentials
 - Check validation of the entered credentials with stored value
 - If (validationCheck==" Valid") {
 - ✓ Show profile/portfolio page of said user
 - } else{
 - ✓ Show validation error
 - }
- } else{
 - Show connectivity error
- }

➤ Buy Trade Execution:

- Fetch watchlist data from Database in real time with 3ms Delay.
 - User selects a Stock , its buy price and its target price
 - Navigate to “Put an order Section”.
 - Enter the Stock name, Buy Price, Quantity and Trade type==”Buy”
 - Set BuyStatus==”Pending”
 - Set Balance= Balance-(Quantity*BuyPrice)
 - If(BuyStauts==”Pending”){ check If (BuyPrice>=LTP) {
 - Set BuyStatus==”Executed”
 - Navigate to Positions section
 - If (BuyStatus==” Executed”) {
 - ✓ Show profit and loss in real time}
 - } else{
 - Show BuyStatus==”Pending”}
- }else{
 - Show BuyStatus==” Executed”}

➤ **Sell Trade Execution:**

- User selects a Stock from Portfolio and its Sell price.
 - Navigate to “Put an order Section”.
 - Enter the Stock name, Sell Price, Quantity and Trade type==”Sell”
 - If (BuyStatus==”Executed”) Set SellStatus==”Pending”
 - If(SellStauts==”Pending ”){ check If (SellPrice<=LTP) {
 - Set SellStatus==”Executed”
 - Set Balance=Balance+(Quantity*SellPrice)} else{
 - Show SellStatus==”Pending”}
- }else{
- Show SellStatus==” Executed”
- }

Flowchart

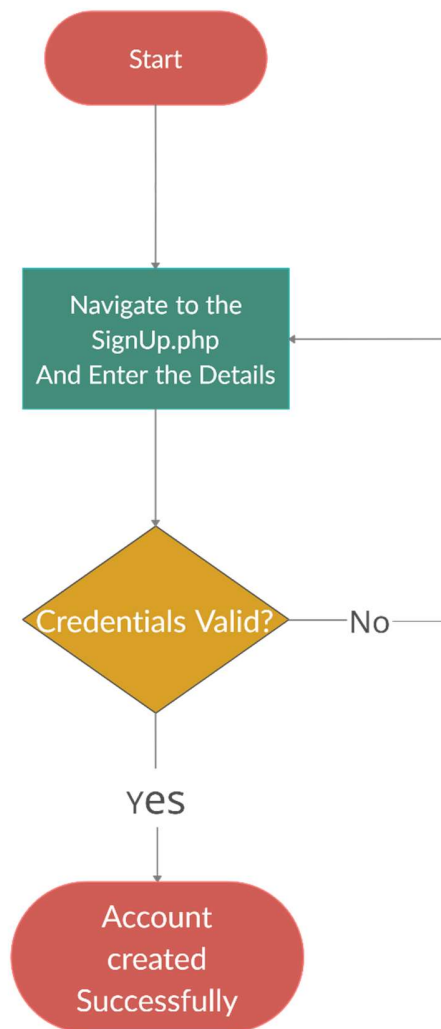


Fig 3.6.1- Sign Up Flowchart

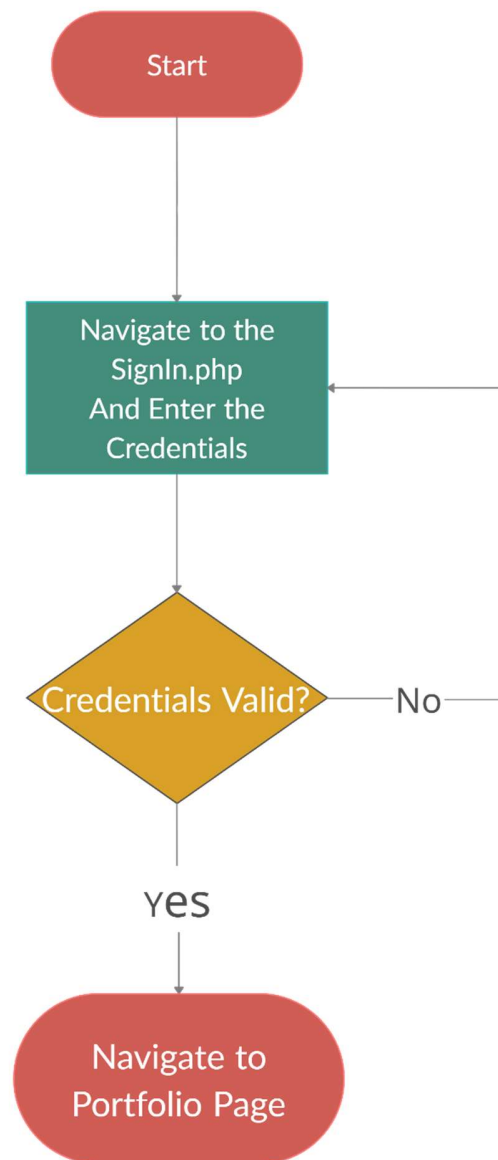


Fig 3.6.2- Sign in Flowchart

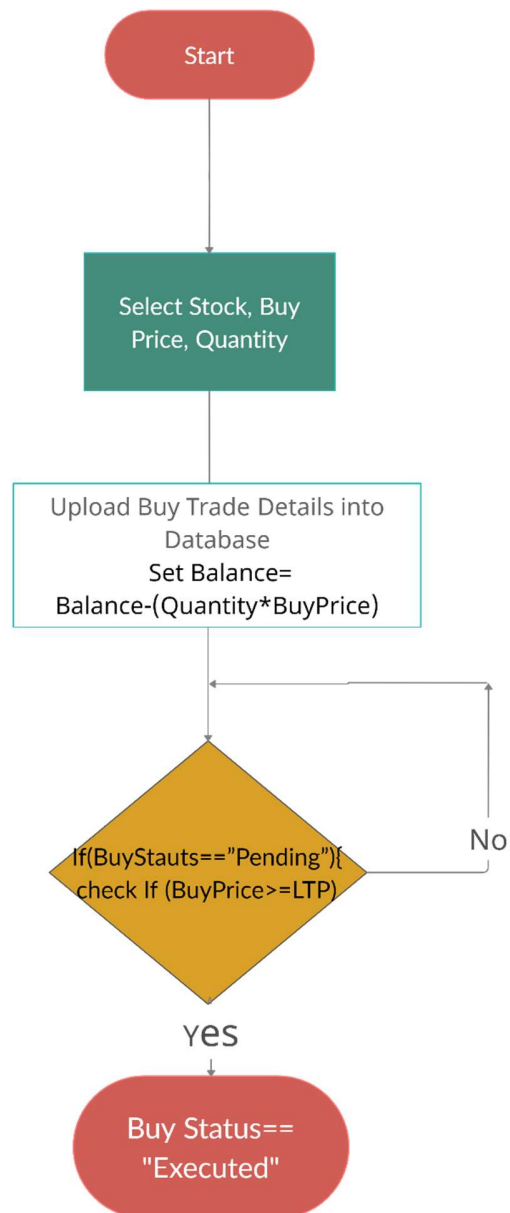


Fig 3.6.3- Buy Trade Flowchart

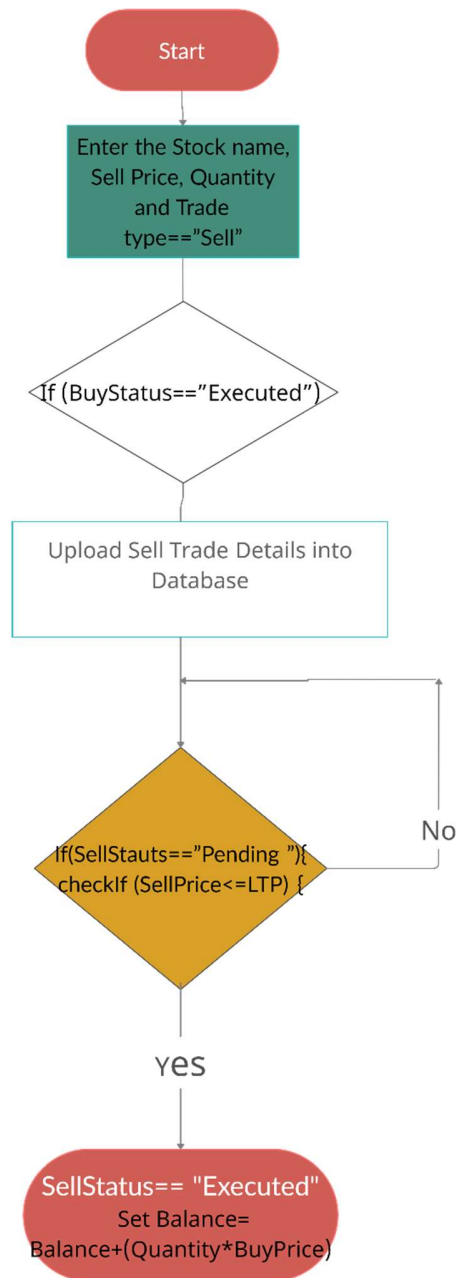


Fig 3.6.4- Sell Trade Flowchart

Screen shots

Website Screenshots:

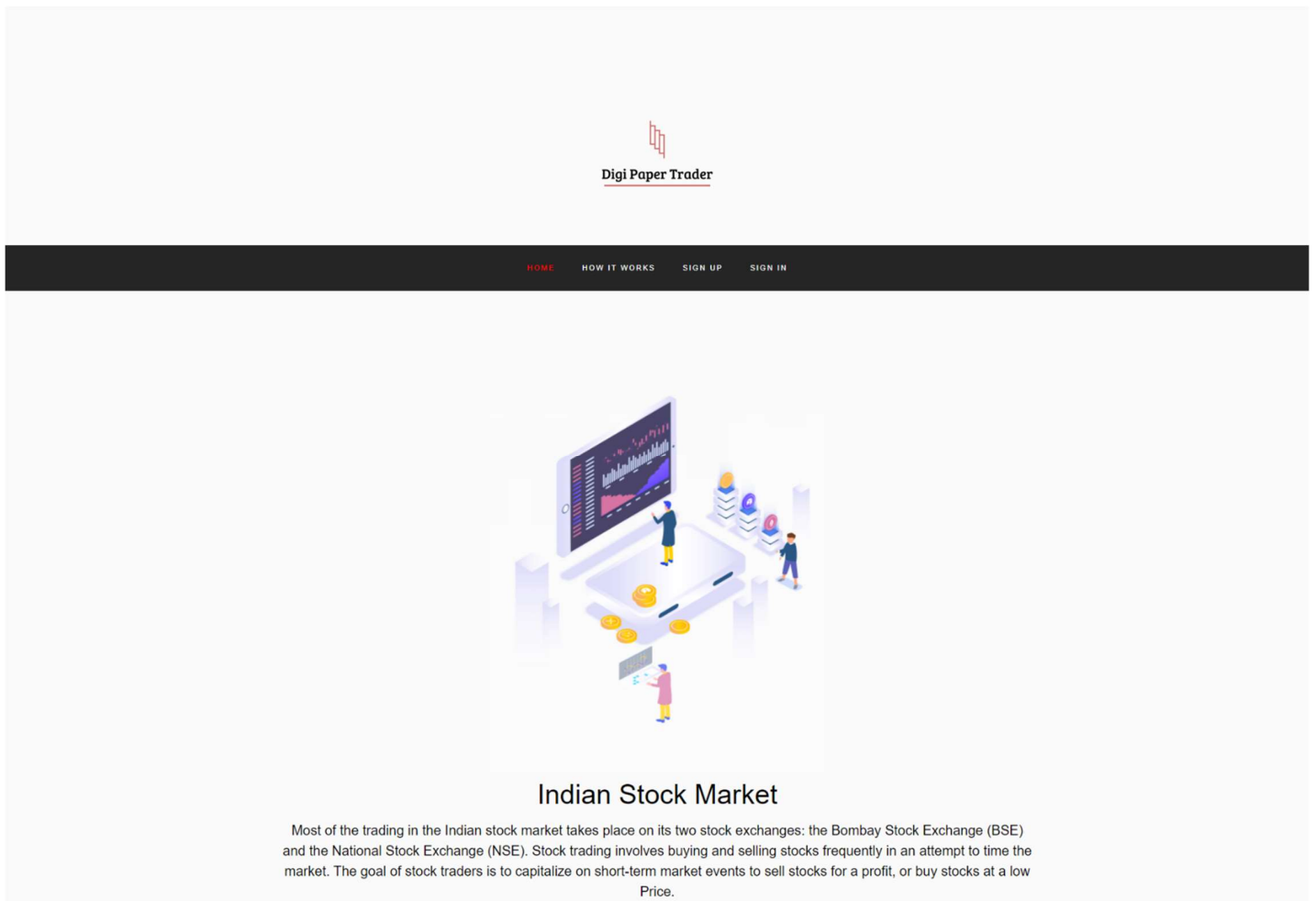


Fig 3.7.1- Homepage



How It Works

This website lets you practise stock market trading with virtual money. You don't have to put in your real money. This way, you can practise stock market trading without having to worry about losing your real money.

Fig 3.7.2- How It Works

This page describes how this website works

The image shows a 'CREATE ACCOUNT' sign-up form centered on a background of financial charts and documents. The form is white with rounded corners and contains the following elements:

- CREATE ACCOUNT**: Title of the form in bold black text.
- First Name: Text input field.
- Last Name: Text input field.
- Phone Number: Text input field.
- Your Email: Text input field.
- Username: Text input field.
- Password: Text input field with a toggle icon (eye) on the right.
- Repeat your password: Text input field.
- ☐ I agree all statements in [Terms of service](#): Checkbox and link.
- SIGN UP**: A blue-to-green gradient button.
- Already have an account ? [Login here](#): Link for existing users.

Fig 3.7.3- Sign Up Page

As the name suggests, this page is used to create account.

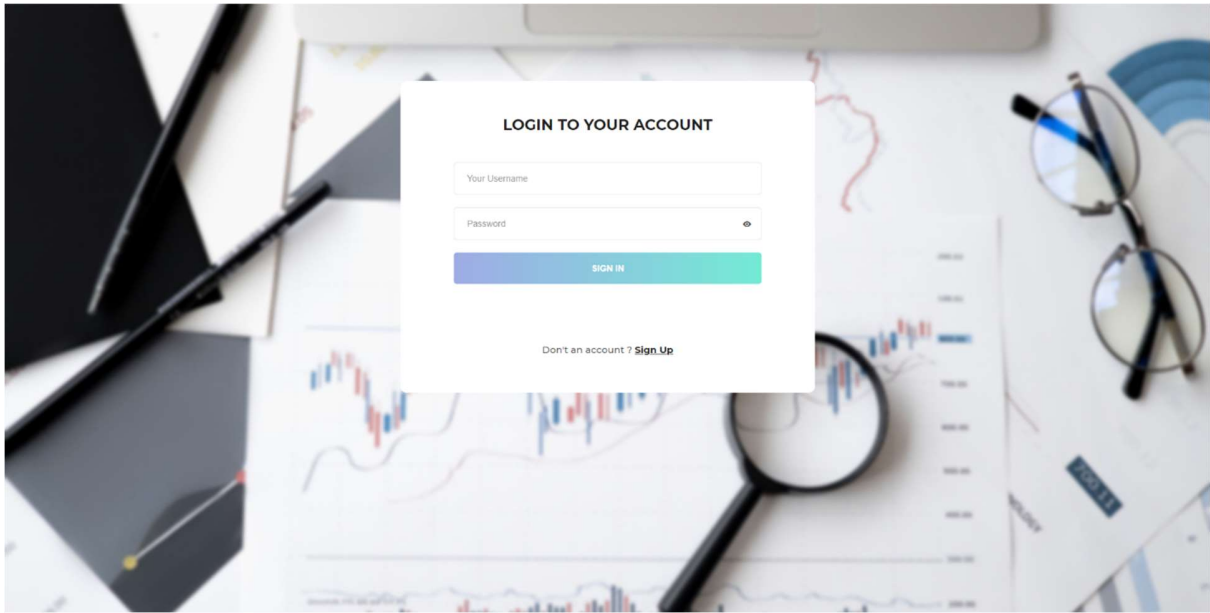


Fig 3.7.4- Sign In Page

This page is used to sign into the user account.

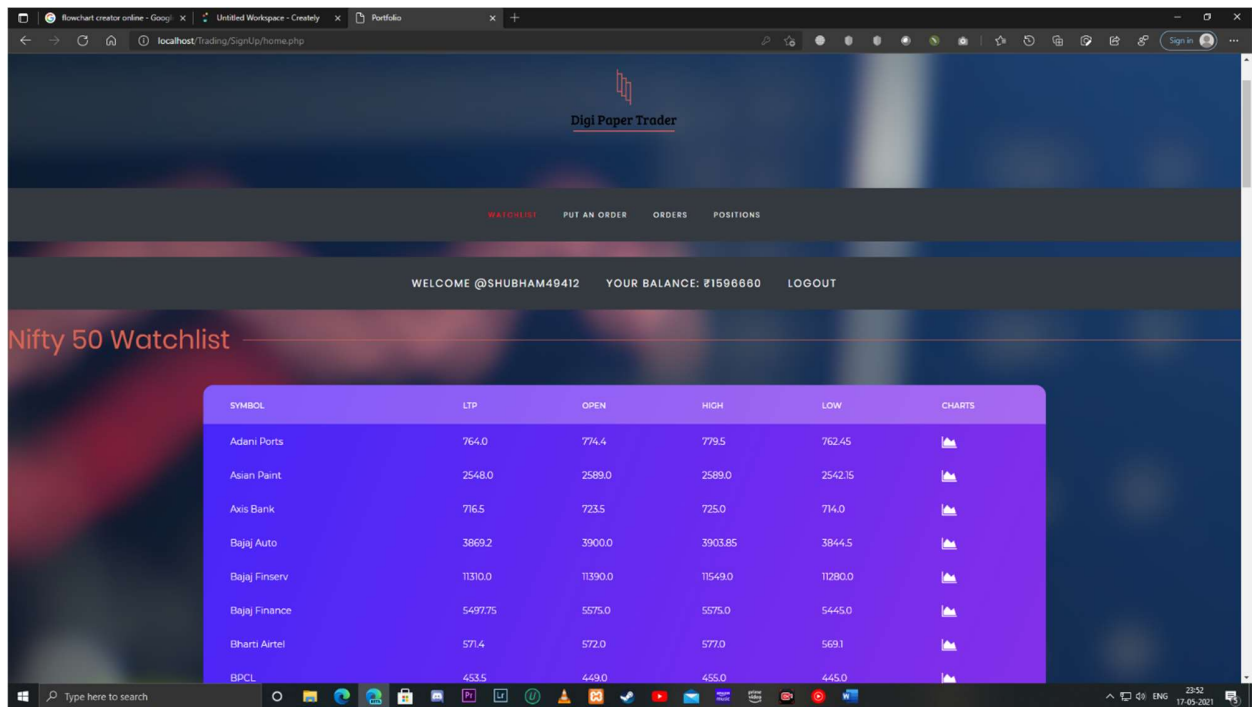


Fig 3.7.5- Portfolio- Watchlist

This section shows the Nifty 50 Stock Details like LTP in real time.

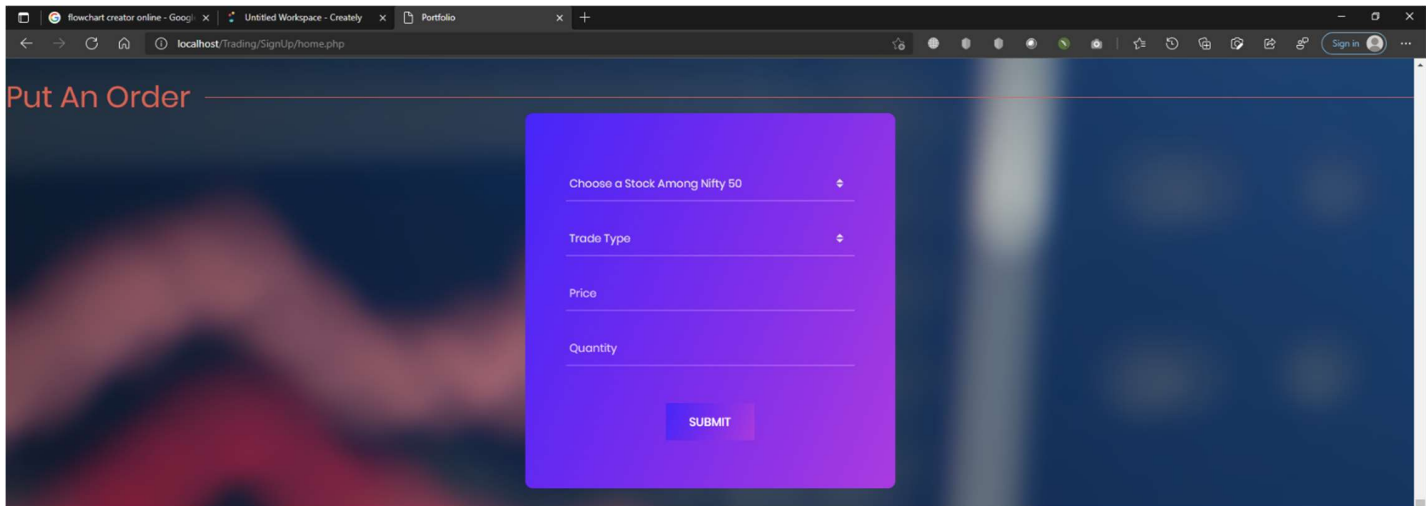


Fig 3.7.6- Portfolio- Put an order

This section of the portfolio is used to Put Buy and Sell Orders.

The screenshot shows a web browser window with the URL `localhost/Trading/SignUp/home.php`. The page has a dark blue background with a purple gradient overlay. The title 'Orders' is in orange. Below the title is a table with the following data:

SYMBOL	BUY PRICE	SELL PRICE	QUANTITY	BUY STATUS	SELL STATUS
reliance	1916		100	Pending	
reliance	1920	1929.1	100	Executed	Executed
itc	203	202.75	200	Executed	Executed
wipro	503		100	Pending	

Fig 3.7.7- Portfolio- Orders

This section of the portfolio shows the current status of the trade, both buy and sell.

The screenshot displays a web application interface for a trading portfolio. The page is titled 'Positions' and features two tables of trading data. The browser's address bar shows the URL 'localhost/Trading/SignUp/home.php'.

SYMBOL	BUY PRICE	SELL PRICE	QUANTITY	BUY STATUS	SELL STATUS	L.T.P	PROFIT/LOSS
reliance	1920	1929.1	100	Executed		1927.1	709.9999999999

SYMBOL	BUY PRICE	SELL PRICE	QUANTITY	BUY STATUS	SELL STATUS	L.T.P	PROFIT/LOSS
itc	203	202.75	200	Executed		205.8	560

Fig 3.7.8- Portfolio- Positions

This section of the portfolio shows the current Profit and Loss in real time.

Computer Networking:

```
Ethernet adapter Ethernet:

Connection-specific DNS Suffix . : 
Description . . . . . : Realtek PCIe GbE Family Controller
Physical Address. . . . . : 40-B0-76-9E-01-12
DHCP Enabled. . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
Link-local IPv6 Address . . . . . : fe80::4cca:80cd:cc4e:f7e2%24(Preferred)
IPv4 Address. . . . . : 192.168.1.2(Preferred)
Subnet Mask . . . . . : 255.255.255.0
Lease Obtained. . . . . : 12 May 2021 16:15:21
Lease Expires . . . . . : 19 May 2021 20:25:04
Default Gateway . . . . . : 192.168.1.1
DHCP Server . . . . . : 192.168.1.1
DHCPv6 IAID . . . . . : 255897718
DHCPv6 Client DUID. . . . . : 00-01-00-01-25-E9-2A-9C-40-B0-76-9E-01-12
DNS Servers . . . . . : 192.168.1.3
                        8.8.8.8
NetBIOS over Tcpip. . . . . : Enabled
```

Fig 3.7.9- DNS Check

In above figure we can see that we successfully created DNS and Set our Local DNS as Primary DNS.

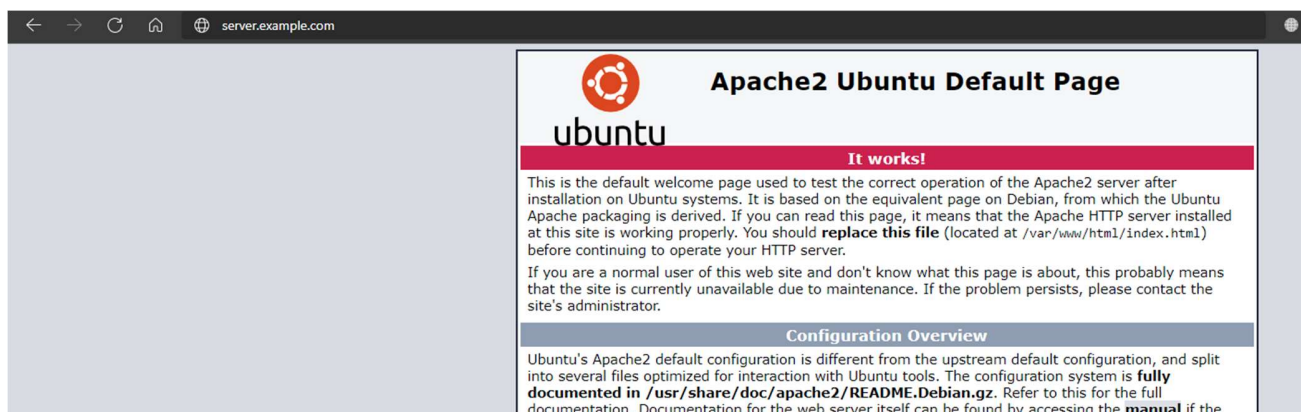


Fig 3.7.10- Domain Name Check

In the Above figure we can see that we can access the website using Domain Name as well.

Discussions on results achieved

➤ **Frontend:** In front-end we were able to create a user interface which is easy to use for each and every individual. And we have kept things easy to use as well as easy to access. Further we have used dark theme which nowadays has become new choice of individuals as it has its own benefits, it causes less strain on eyes as well as help the system endure longer because it consumes less battery. We have also given Chart button in the watchlist which will redirect to the chart of the said stock.

➤ **Backend:** In the back-end we were able to create database using sql server, we created two databases named “**stockmarketsimulator**” and “**stock**” which consists of various tables which provide the functionality like *Creating an account, fetching stock details, putting buy sell orders*. We created a python scripts using nsetools package in order to fetch data from the National Stock Exchange in real time, which after fetching, stores that data in the database. We have used php for various purposes like, connecting the website to the database, fetching and uploading data from and into the database in the project. We have used apache server in order to host the website within a local area network. Above mentioned tasks and procedures were successfully completed.

➤ **Logics Implemented:** In this project we were able to create and implement logics in order to resolve the transaction between the user as well as the system, and were able to create logics for retrieving as well as sending data back to the Database. Logics for validation while signing up and signing in were implemented successfully.

Applications of the minor project

- **Social Applications:** Every individual can use the project in order to learn about the stock market and practice trading, how it works and how he/she can invest in the market wisely and make smart decisions to increase their profits. So, it becomes easy for individual to learn about stock market without even spending a penny. Therefore, our project is more of a risk-free environment where you can learn more and more about stock market and trading.
- **Industrial Applications:** This project can also be used for industrial application, it can help big investment firms to help them train their employees in the stock market without even investing real money thereby decreasing risk of any losses, which makes it more cost efficient. Furthermore, we have also created a local DNS server so that it can only be used within a given local area network making it exclusive and secure within that local area network. With the help of Local DNS, users/Employees wouldn't need to remember the IP address of the server to access the website, they can use the Domain Name set up in the local DNS. As we have implemented local DNS, it can be used to set Domain Name for other Ips inside a network.

Limitations of the project

- **Only Nifty50 Stocks:** As of now we were able to create the script which could fetch only Nifty50 Companies, we chose to fetch those as, those are the blue-chip stocks and most people prefer to trade in them only because there's very low chance of those companies failing, or even saying impossible wouldn't be so farfetched. But nonetheless, we were only able to fetch those nifty 50 stock details. So, for now, people can only trade among those nifty 50 stocks.
- **No Feature of Stoploss:** It's a basic and important practice to put stoploss while trading, as it would minimize the loss. Stoploss is a feature in which the trader sets a price, if a trade starts making loss, and reaches that certain price level it would trigger the price and book the loss. Due to time restrictions, we weren't able to create logic to implement Stoploss.
- **No Feature of Short Selling:** We can trade in two ways in stock market, we can either buy first at low price if we think it has potential to go up and then sell it at higher price, or, we can Sell First at higher price if we think the stock will fall, and then we can buy that stock later, completing the Buy and Sell Transaction.
- **Lag:** We can experience a little bit of lag while fetching the data from National Stock Exchange using nsetools.

Future work

- **Adding more Stocks:** We can use other APIs to fetch the data from the Stock Exchanges which would allow us to trade in more stocks.
- **Adding Future and Option Trading:** Future and option, abbreviated F&O is a type of trading which provide higher volatility even in a Slow market, allowing people to make more money, although it is very risky as the volatility is very high. In Future we can try to provide Future and Option trading.
- **Adding Stoploss Feature:** As discussed above, stoploss is very important feature while trading as it would allow to minimize the loss, in future we can add the feature of putting stoploss.
- **Adding Short Selling feature:** In bearish market, it's very unwise to enter in a buy first sell later kind of a trade but sell first, buy later i.e., Short Selling could prove very beneficial. In future we can create logic for short selling.
- **Reduce Lag:** We have already figured out that the lag we're facing is due to nsetools package, but due to lack of availability of efficient APIs we have no other option as of now, but in near future we can look for a solution.
- **Adding more features:** In future we will be adding more features in order to make user experience better and also important trading features like GTT order, Cover Order etc so it becomes more intuitive as well as practical.

References

1. <https://nsetools.readthedocs.io/en/latest/usage.html#getting-a-stock-quote>
2. <https://stackoverflow.com/questions/20089758/how-to-retrieve-data-from-database-without-refreshing-page-using-ajax>
3. <https://linuxtechlab.com/configuring-dns-server-using-bind/>
4. <https://colorlib.com/wp/templates/>
5. https://www.w3schools.com/python/python_mysql_getstarted.asp
6. <https://www.tradingview.com/>