Republic of Tunisia
Ministry of Higher Education and
Scientific Research

General Directorate of
Technological Studies

**Higher Institute of Technological
Studies Sidi Bouzid**

A Final Year Project

On

Basic Technical analysis

Presented to
Department of Computer Science
The Higher Institute of Technological Studies. Sidi Bouzid

by

Fridhi HAMDI
Mnassri MOHCEN

In partial fulfillment of the requirements for the award of Bachelor's degree in Computer Science

June,2022

Defended in front of the jury members :

Mrs.	**Azeri ROUKEYA**	**President**
Mr.	**Zouidi NABIL**	**Rapporteur**
Mrs.	**Boukattaya FATAMA**	**Supervisor**

DEDICATION

I dedicate this work to my source of motivation, to those who have never stopped supporting me, encouraging me and guiding me throughout my life , to my dear parents.May God preserve them and grant them a long life until they see me fulfilling their expectations . I also would like to dedicate this work to all my friends with whom I have shared my moments of joy and happiness.

ACKNOWLEDGMENT

I would like to express my special thanks of gratitude to some people , without whom I would not have been able to complete this report, and without their help and proper guidance I would not have made it through my bachelor's degree . My special thanks goes to **Mrs Boukattaya Fatma** , my supervisor for her countless hours of reflecting, reading, encouraging, and for her patience throughout the entire process. I am also greatly thankful for my supervisor from Ahwa Solution , **Mr. Hamdi Hosni** , for providing guidance and feedback throughout this project. I also wish to express my deepest gratitude to all the members of this jury, **Mr** Azri Roukeya and **Mr** Zouidi Nabil

Table of content

List of figures

List of tables

List of abbreviations

RUP: Rational Unified Process

2TUP: 2 Track Unified Process

UML: Unified Modeling Language

JWT : Json web token

CSRF : Cross-site Request Forgery

XSS : Cross-site Scripting

SEO : Search Engine Optimization

JSON: Javascript Object Notation

HTTP : Hypertext Transfer Protocol

HTTPS: Hypertext Transfer Protocol Secure

WS : Websocket

WSS: Websocket Secure

API : Application Program interface

SSR : Server Side Render

SSG : Static Site Generating

HTML : HyperText Markup Language

JS: JavaScript

General introduction

In the modern world the key to success is gaining as much information as we can .In the world of investment and trade, knowledge makes the difference between good and bad investment . Providing information comes after many steps. You need to work on it to minimize risks and make the work easier .

Nowadays, computers have made 70% of the return . unfair access to the resources and technologies to building applications we hope to decrease the gap between individuals and institutions and empower the individual .

Now

Trying to solve this problem , we decided to divide this report into 4 parts:

- Chapter one **"Project context"** presents the Hosting company and describes the problem and solution , and goes through the methodology .
- Chapter two **"Requirements gathering"** treats the functional, nonfunctional requirements and use case diagram .
- Chapter three **"Detailed Conception"** details our conceptual diagram such as sequence diagram and class diagram .
- Chapter four **"Realization"** goes deeper into the technologies and the lessons we have learned .

Chapter I : Project context

Introduction

In this chapter we will give a brief overview of the project, hosting company , the general framework of the project, present the existing problem and how to be overcomed and finally mentioning similar available applications.

I. Hosting company

Ahwa Solutions is a one-man company (SUARL) created in 2013 with a tax number 1295118/F. Being a fully exporting IT company, its main activity is the manufacture of computer software as well as general IT services.

In 2017, the company has its online educational platform "SABOURA.net" .

It aims at supporting students to better understand their lessons before and after school. The platform Saboura.net is available to students and parents both in Tunisia and abroad.

Figure 1 *: Saboura.net logo*

Figure 2 : *AHWA Solutions*

II. Problematic

Nowadays the information is the key to success , and it really makes a difference in the world of investment .And the less financial knowledge and Understanding people, the worse they cope, to get the right information takes a lot of time and time is the main reason to succeed in investing , so we try to solve it by creating this web application . In our application we have collected data and news in website and make a simple analysis for a stock market and it is easy accessed to.

III. Analysis of existing applications

III.1. Description of the already existing applications

Our project is about collecting the hot news from other famous financial news websites like Forbes , cnbc , fox business… , helping the user watch the equities , the shares , getting notification each time the price changes and giving a simple technical analysis for every stock.

Below are similar websites and platforms :

1. Investing.com

Investing.com is a financial platform and news website; one of the top three global financial websites in the world. It offers market quotes,information about stocks, futures, options ,analysis, commodities and an economic calendar [1].

Figure 3 : *Investing.com logo*

2. Tradingview

TradingView is a social media network, analysis platform and mobile application for traders and investors. The company was founded in 2011 and today it has offices in New York, London and Russia. In 2018, TradingView completed a $37 million Series B round led by Insight Partners, and in 2019 it acquired TradeIT. Currently, the company ranks on the top 130 websites globally [2].

Figure 4 *: TradingView logo*

III.2. Critic of existing applications

In investment sector and trading have a lot of problems :

- Some investors inevitably have too little information, others have too much and this leads to panic ,bad decisions and trusting the wrong people. When people are exposed to too much information, they tend to withdraw from the decision-making process and reduce their efforts.
- Technical analysis and price action analysis seems so easy and simple. The patterns look very clear and obvious to an inexperienced trader but the truth is that technical analysis and price action trading is maybe one of the most subjective and difficult to master trading styles out there.

III.3. Proposed solution

To solve the previous issues we are using an external api from YAHOO finance that gives us real time data of stock and TradingView widgets and makes the technical analysis use the QuantConnect api to solve the problem of technical analysis

IV. Study of the development methodologies and Project planning

IV.1. Study of the development methodologies

The world of software development contains many methodologies like **Scrum** methodology , **RUP** method , **2TUP** method and many more .But , we have chosen the **RUP** for the following reasons :

- It allows us to deal with changing requirements regardless of whether they are coming from the customer or from the project itself.
- It forces integration to happen throughout the software development, more specifically in the construction phase.
- It is simple to use and less risky .

RUP : Rational Unified Process is an agile software development method in which the life cycle of a project, or the development of software is divided into four phases. Various activities take place during these phases: modeling, analysis and design, implementation, testing and application [3].

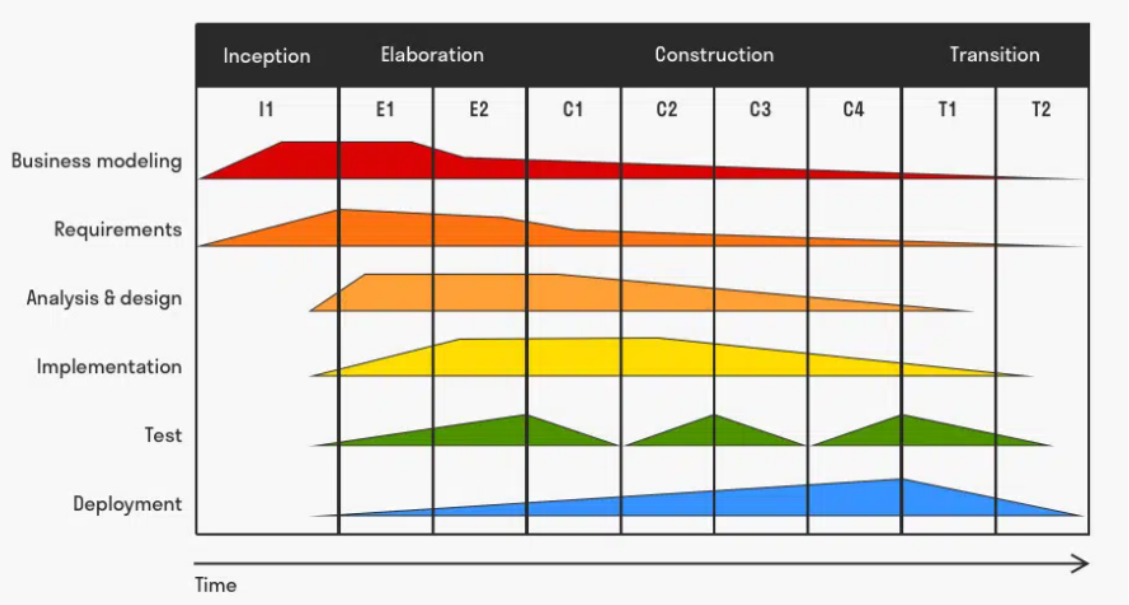

Figure 5 *: Rup methodology [4]*

IV.2. Project planning

The principal key of success in any project is good planning . Gantt chart is greatly used in project management because of its ease.

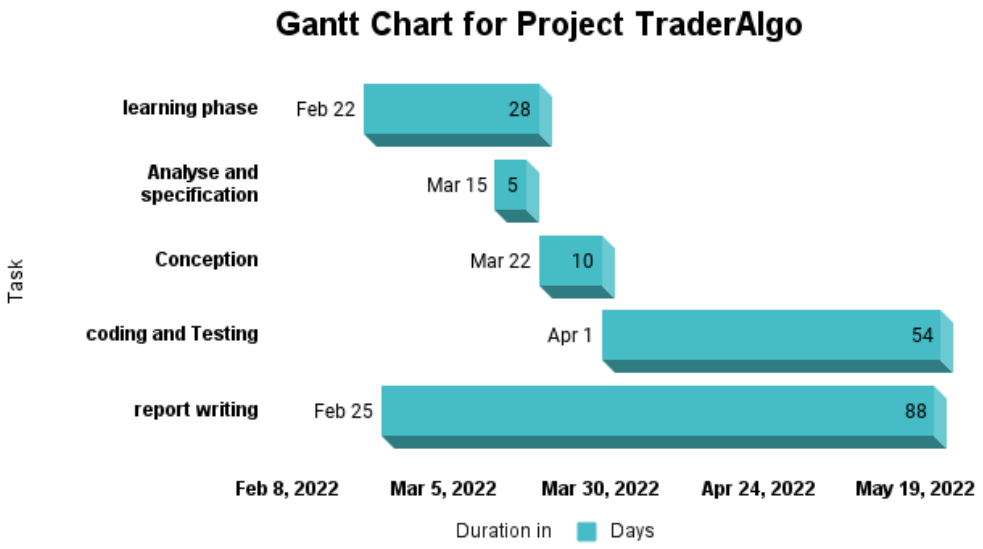

Figure 6 : *Gantt chart*

Conclusion

In this chapter, we have presented the hosting company and stating problems we have faced .Then, we have tried to find a solution by using the appropriate methodology in development process .

Chapter II : Requirements gathering

Introduction

In chapter two we have to clarify information during the development stage and development time but costs may increase, as well as the probability that the project may fail.To avoid these problems, we have to identify functional and nonfunctional requirements for the project . While functional requirements may be clearly identified , some of the non-functional requirements are hard to specified and

I. Functional requirements

As we mentioned earlier, clearly defined requirements are the key to project success . The application offers to the **Guest** simple data about the currency market

But the **user** allowed to :

- Sign in : Guest can register by filling a form or use his facebook account or google account
- login : Guest can login again to dashboard
- Manage profile
    - change background profile
    - change profile photo
    - update his public data (name , email and facebook …)
    - update his private data (Location , City , Tax ID …)
    - change password
    - delete his profile
- View News : read all important and hot news
- create alert :
    - add the price of equity that if changed we will send a notification or email .
- create collection and watchlist :
    - create collection
    - delete collection
    - update collection

II. Non functional requirements

In fact, non-functional requirements are hard to identify . Yet , we have tried to cross the most influential ones .

II.1 Ergonomics of user interface and simplicity

We have built websites that provide a simple overview about each company .It's an essential part to provide the most compatible and easy user experience and interface. Our project's interface requires no previous knowledge of the stock market and it greatly conforms to design best practices.

II.2 Security

The Used framework provides a lot of security to our application :

- Passwords must at least be 8 characters , all the passwords are hashed when saved in the database.
- all sensitive parameters (database name or password) saved in the environment file.
- Using the third part as Google or Facebook to login.
- JWT are encrypted and saved in client side to avoid preventing CSRF and XSS attacks
- use Recaptcha to avoid spam and robots .

II.3 Performance

In the world of investment time is the key to success by using NEXT.js improved performance and making the application faster and improving SEO . FastApi is one of the fastest Python frameworks available because of its very high performance

II.4.Scalability

Scalability is one of the most critical conditions for the success of a web application. Next.js makes it easy to scale multiple pages because it can individually select how to render each page either on the client, the server, or hybrid.The result of writing an application using FastAPI framework is a minimal dependency and a scalable template repository .

III.Use case diagram

The main goal of a use case diagram is to summarize the details of our system's users and their interactions with the system

III.1. Identifying actors

User : is the main actor in our application . To get access to all means functionality , the user must register .

III.2. Use case diagram description

1. **General use case diagram**

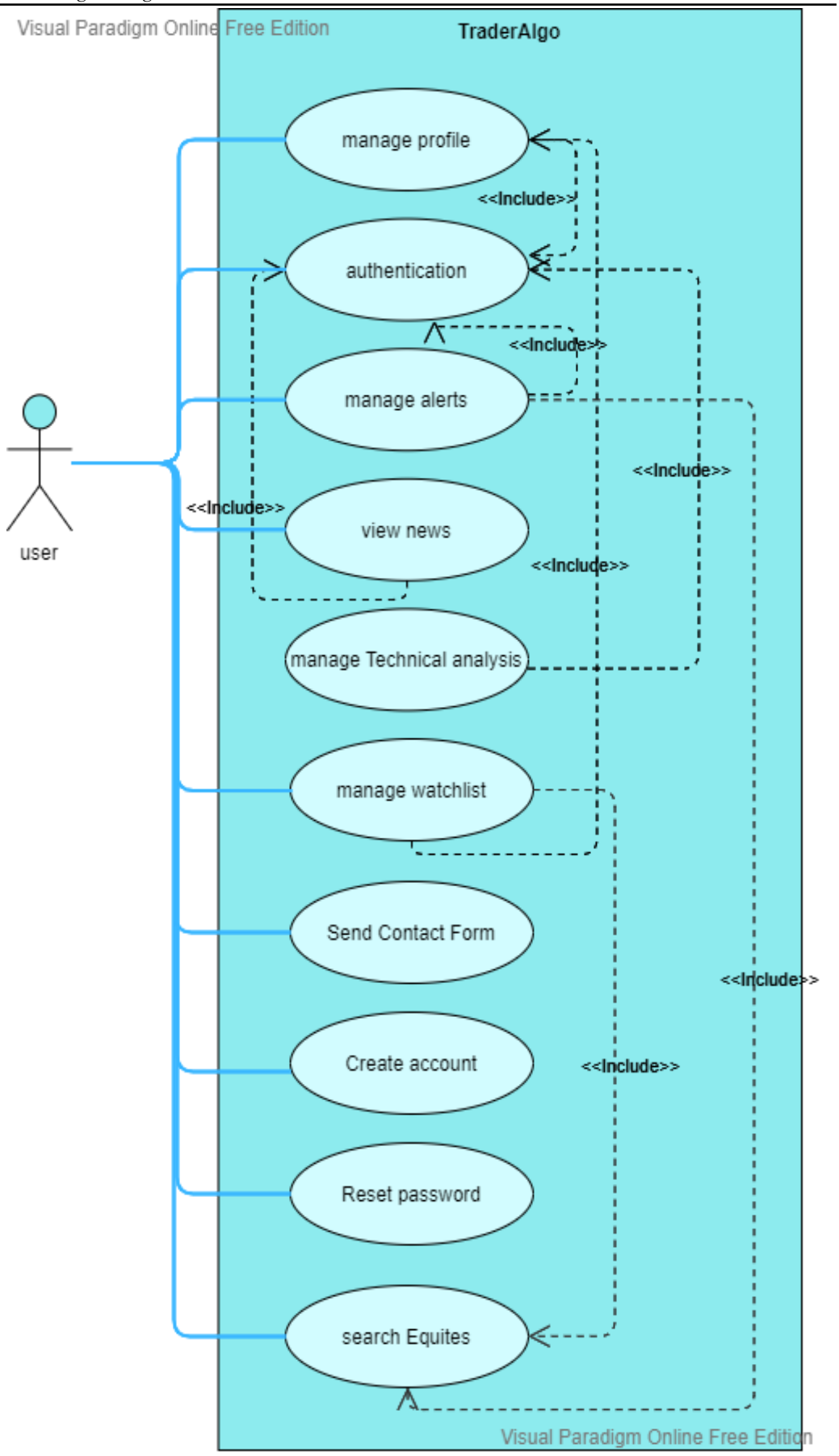

Figure 7 : *General use case diagram*

2. Detailed use case' diagrams

To better understand the previous diagram , we have divided it into small ones.

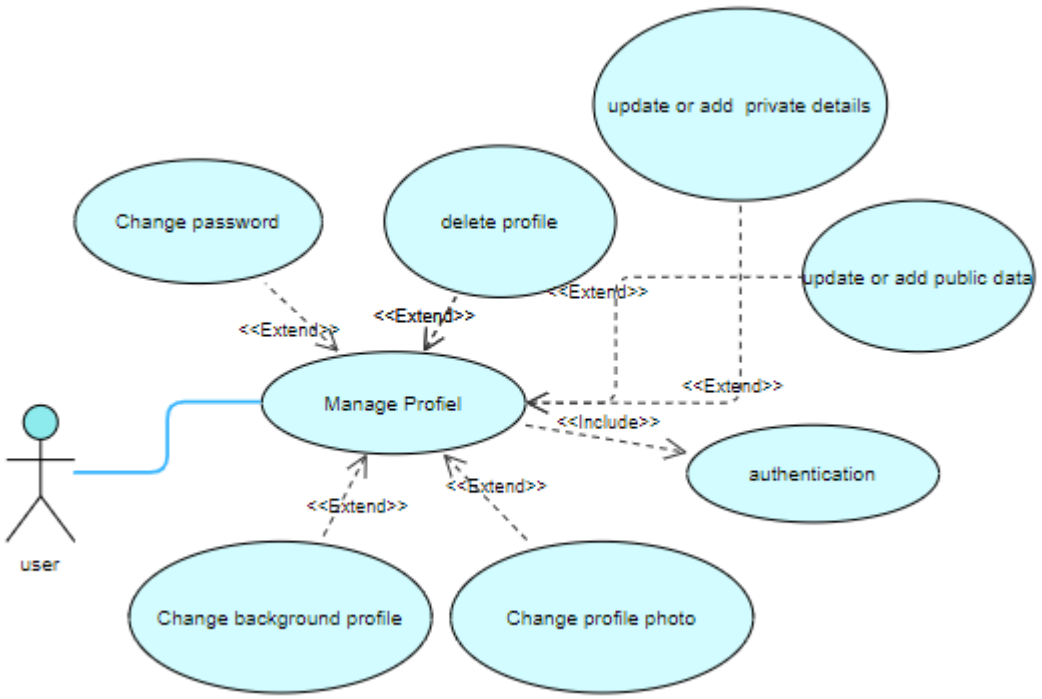

Figure 8 : *Use case diagram "Manage Profile"*

Title	Use case
Summary	Manage Profile
Actor	User
Pre-condition	user is authenticated
Nominal scenario	Many options are presented when the user is in the setting page ● Change the profile background : ○ choose a photo from your device, then upload it to Google drive where the image is stored then save the link its in the database. ● Change the profile's photo : follow the same steps as background photo ● Change password ○ if you login with facebook or google account you are able to create a new password without verifying your old password. ○ if you use email to log in, you need to verify the old password . ● Update or add public or private data : ○ For the first time you add data and save it into the database. You can also change data.

	• Delete profile: ○ when the user logs in with a facebook account or google account , he doesn't need to write a password to delete the account . Whereas, when he logs in using email, he needs to write the password .
Alternative scenario	• return to the login page . • redirect to the not found page . • show a toast failed message . • show under input error message.
Post Condition	• show a toast with success message

Table 1 : *Use case « Manage Profile » textual description*

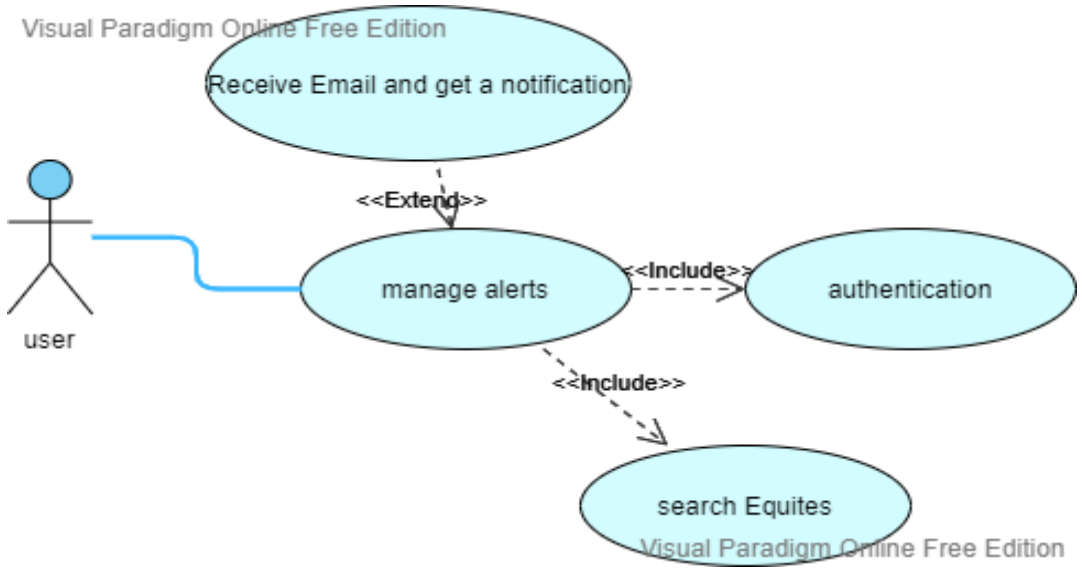

Figure 9 : *Use case diagram "Manage Alert"*

Title	Use case
Summary	Manage Alerts
Actor	User
Pre-condition	user is authenticated search for a company he needs.

Nominal scenario	To create an alert :
	• After you find the shares your need :
	  ○ add price your want
	  ○ choose one of the options selecting above or below
	  ○ choose if you want to receive an email with notification or not .

	○ stored all details into database
Alternative scenario	● return to the login page . ● redirect to the not found page . ● show a toast failed message . ● show under input error message.
Post Condition	● show a toast with success message

Table 2. Use case " Manage Alerts" textual description

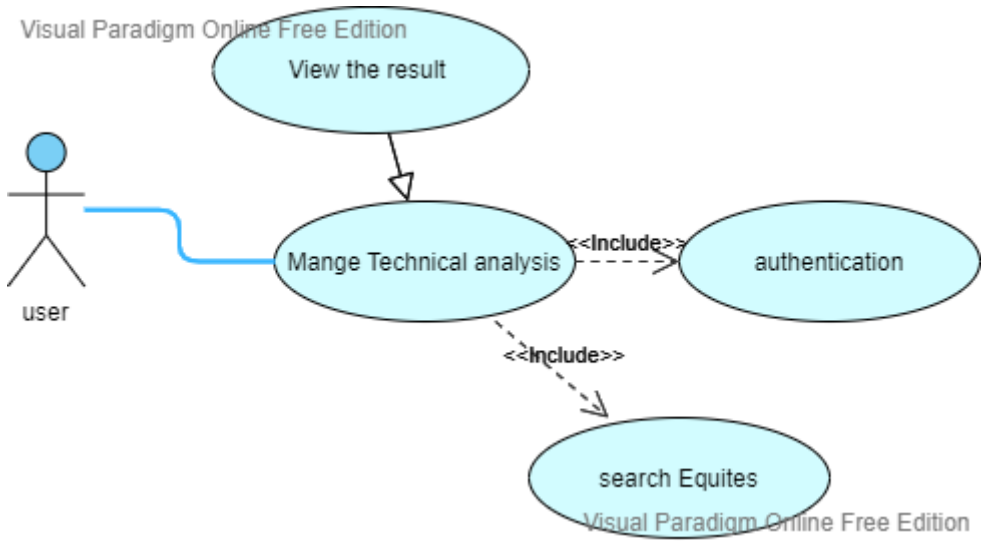

Figure 10 : Use case diagram "Manage Technical Analysis"

Title	Use case
Summary	Manage Alerts
Actor	User
Pre-condition	the user is authenticated .
Nominal scenario	To see the result, you must choose the company he wants and duration if the user doesn't make a decision he will get apple analysis by default .
Alternative scenario	● return to the login page ● redirect to "not found page"
Post Condition	● show result .

Table 3. Use case" Manage Technical Analysis" textual description

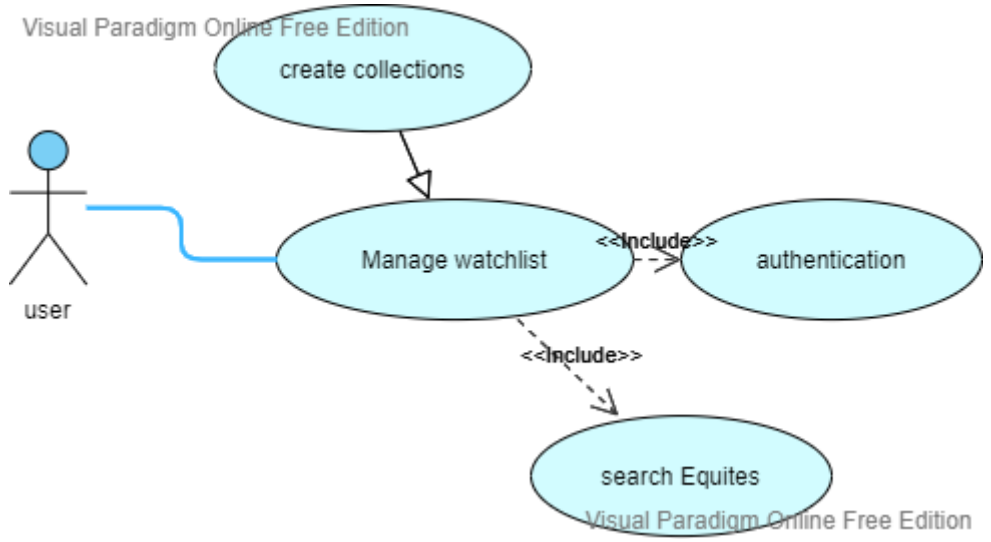

Figure 11 : *Use case diagram "Manage watchlist"*

Title	Use case
Summary	Manage Watchlist
Actor	User
Pre-condition	the user is authenticated .
Nominal scenario	To add a share to watchlist, the user has to make a search for the company he wants and to view his list must follow the following : ● Create a collection : ○ collections must have unique name ○ update or delete a collections ○ collections have a default collections called "watchlist" ● add the equity you have chosen to your collections .The icons change
Alternative scenario	● return to the login page . ● redirect to the not found page . ● show a toast failed message . ● show under input error message.
Post Condition	● show a toast with a success message.

Table 4. *Use case "Manage watchlist" textual description*

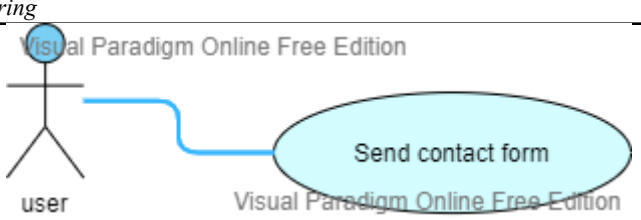

Figure 12 : *Use case diagram "Send Contact form"*

Title	Use case
Summary	Send Contact Form
Actor	User
Pre-condition	non need for any pre condition
Nominal scenario	anyone can send email : ● full name , email , subject and message are required fields. ● email must be a valid email . ● subject length must be greater than 20 characters . ● message length must be greater than 25 characters.
Alternative scenario	● redirect to the not found page . ● receive a toast with a failed message .
Post Condition	● receive a toast with a success message.

Table 5. *Use case "Send Contact form" textual description*

Figure 13 : *Use case diagram "Create an account"*

Title	Use case
Summary	Create an account
Actor	User
Pré-condition	non need for any pre condition
Nominal scenario	we have two ways to create an account:

	• Through a third party facebook account or google account.
	• Sign in by email :
	  ○ Email is a required field and must be valid and unique.
	  ○ The password must also be a required field and must be more than 6 characters .
	  ○ Verify you are not a robot by google Recaptcha.

Alternative scenario	• show an error message under each input . • receive a toast with a failed message .
Post Condition	• redirect to dashboard

***Table 6**. Use case "create an account" textual description*

***Figure 14 :** Use case diagram "Reset password"*

Title	Use case
Summary	Send Contact Form
Actor	User
Pre-condition	user already have an account
Nominal scenario	for reset your password : • check if your email exists or not • send a reset email to your email with a reset token. • check if the reset token is valid . • create your new password and confirm password. • password must be greater than 8 characters . • stored hashing password into the database .
Alternative scenario	• redirect to the not found page . • receive a toast with a failed message . • show an error message under the input field.
Post Condition	• redirect to the dashboard page .

***Table 7.** Use case "Reset Password" textual description*

Conclusion

The second chapter deals with the general use case . Studies use cases detailed identify the actors and mentions the functional and non-functional requirements.

Chapter III : Detailed conception

Introduction

Chapter treats conception fases with more detailed and how our application works

I. Justification of choice

In our project we have used the **UML . UML (**Unified Modeling Language) which is a very important part of the software development process. Using UML helps us to communicate, explore potential designs, and validate the architectural design of the software.we will give detailed ideas about our project in each type of Diagram , we will use Class diagram and Sequence diagram.

Class diagram : is one of the most useful types of diagrams in UML as it clearly maps out the structure of a particular system by modeling its classes.

Sequence diagram : shows the sequence of messages passed between objects. and the control structures between them.

II. Class diagrams

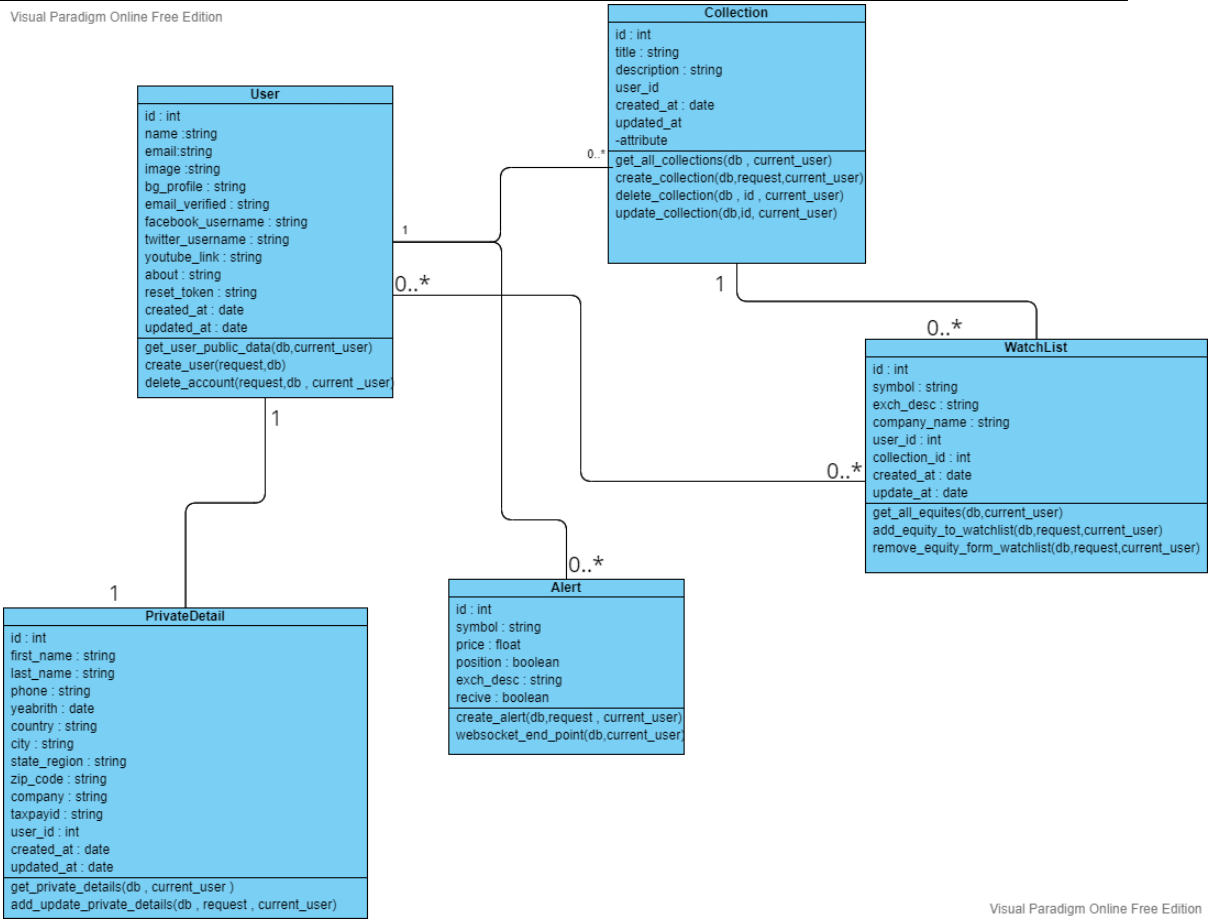

Figure 15 : *Class diagram*

III. Sequence diagrams

III.1.Authentication sequence diagram

For authentication we have two methods :

- Log in by Email and password
- Login by Gmail or facebook account

- Log in by Email and password :

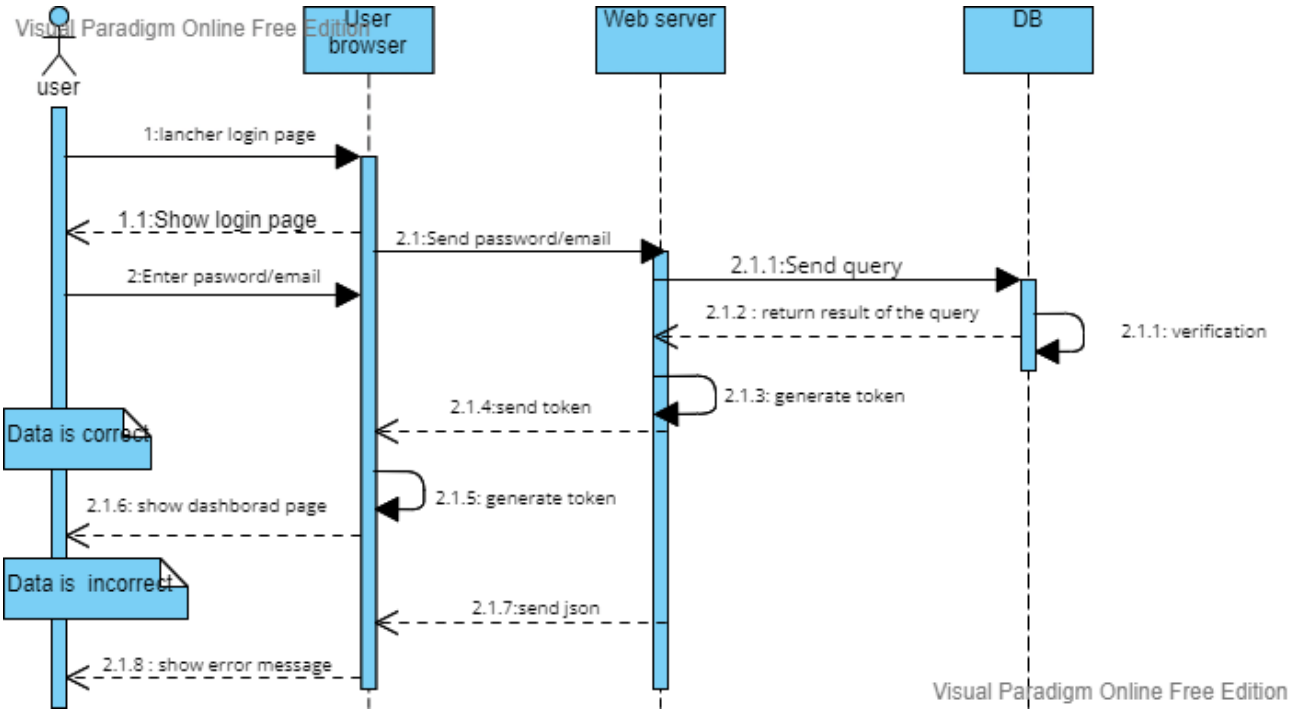

Figure 16 : *Sequence diagram "Authentication using email and password"*

- Login by Gmail or facebook account (Facebook authentication example)

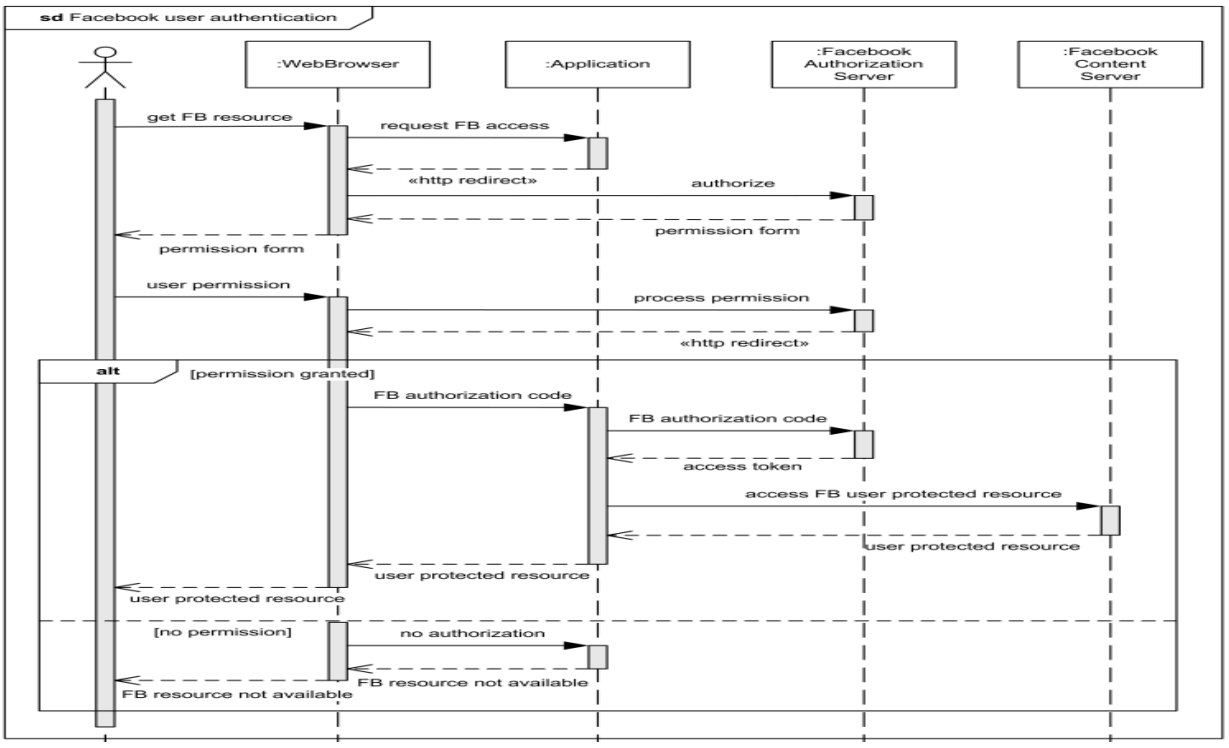

Figure 17 : *Sequence diagram "Authentication using facebook account"[5]*

III.2.Create the watchlist sequence diagram

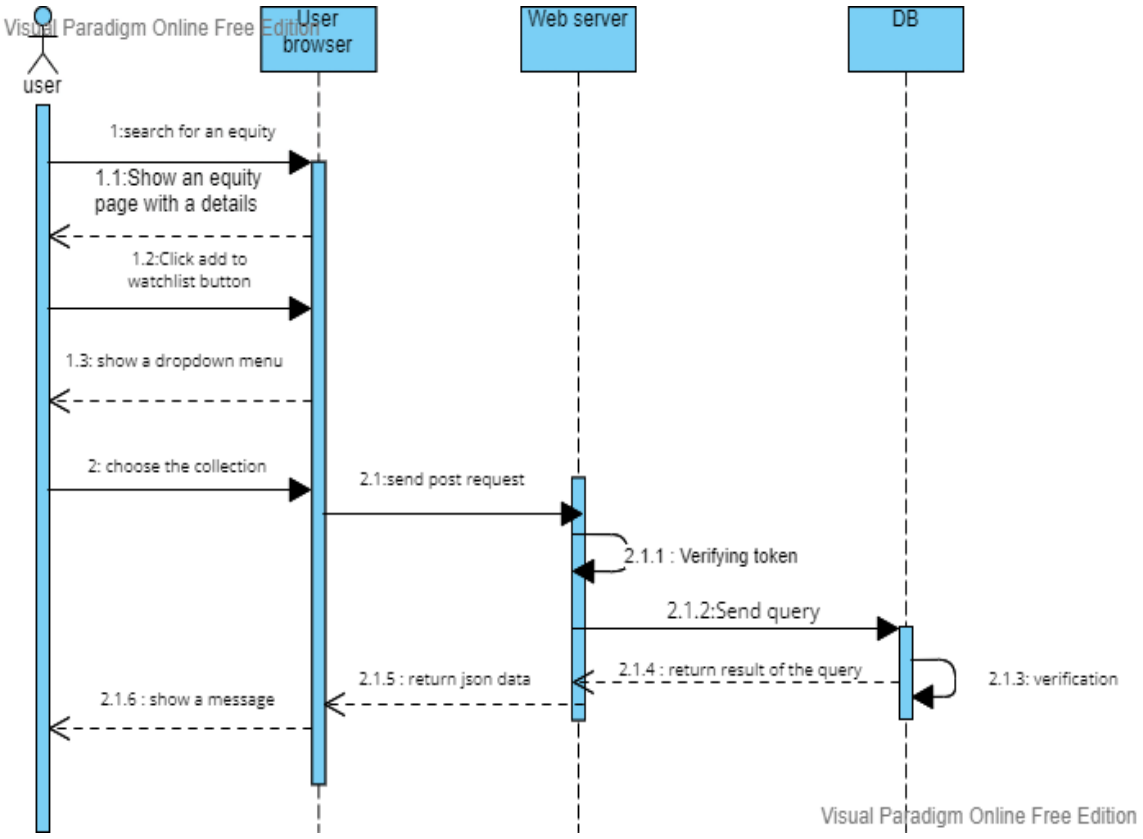

Figure 18 : *Sequence diagram "Add to watchlist"*

III.3. Create an alert sequence diagram

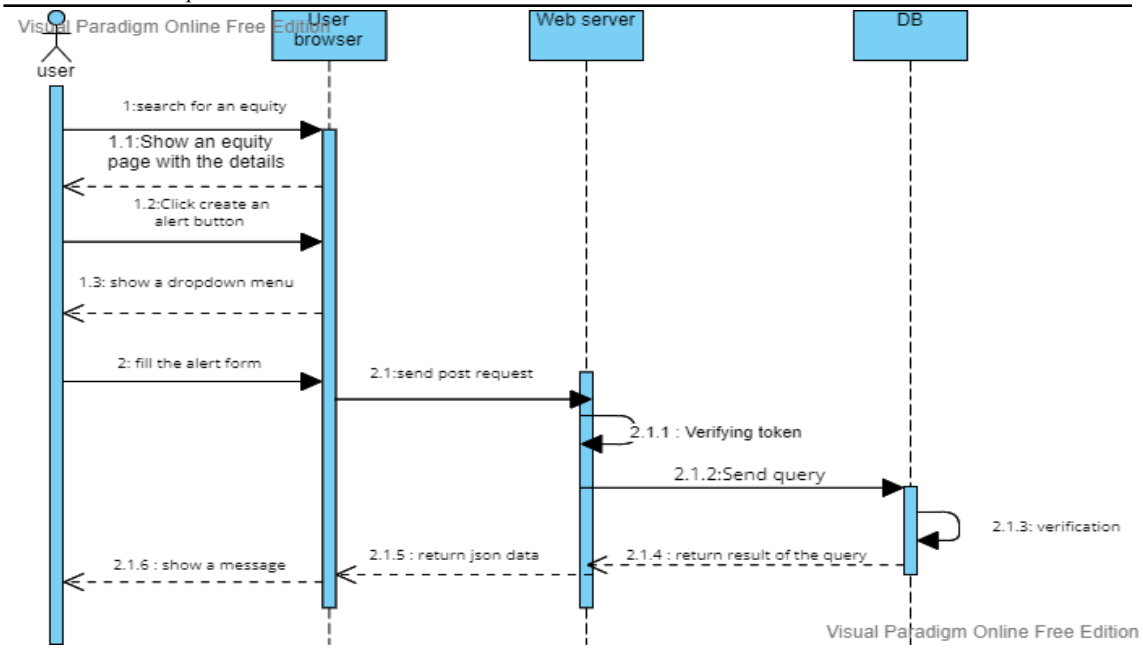

Figure 19 : *Sequence diagram "Create an alert"*

Conclusion

In this chapter we have tried to come over the reason for using UML explain the class diagram. We have illustrated the sequence diagram in more details .

Chapter IV : Realization

Introduction

In this chapter we aim at presenting the technical choice and different available tools so as to develop "TraderAlgo" application available . We have also explained how this application works and presented the different interfaces to our project.

I. Technical needs specification

I.1 QuantConnect

QuantConnect is an open-source, cloud-based algorithmic trading platform for equities, FX, futures, options, derivatives and cryptocurrencies. QuantConnect serves over 100,000 quants from 170+ countries, with customers including hedge funds and brokerages, as well as individuals such as engineers, mathematicians, scientists, quants, students, traders, and programmers [6].

Figure 20 : *QuantConnect logo*

QuantConnect supports a lot of libraries such tensorflow , nltk ,keras , numpy , pandas , scikit-learn and 100s more. it is a very flexible platform that lets you create any algorithm you want .

How do traders' algorithms work ?

When you try to build traders algorithms , you basically try to model and quantify reality . Your algorithm is trying to make a decision based on your models and for the previous reasons you need to have as mush data as possible :

- Time and price data
- Fundamental data
- News data

- Sentiment data …

After receiving this data the algorithm is trying to analyze it to make a decision then pass it to your broker . Fortunately all this work is carried off by QuantConnect.

Figure 21 : *QuantConnect process*

The process for developing trading algorithm strategy ?

1. **Create a strategy**

To build an algorithm trading strategy is not about writing some code and forgetting it. In fact, the process is quite different. Initially we have to clarify the difference between a good and bad trading algorithm is **EDGE**. you must identify first an edge then build around that edge a strategy not to bring a random strategy and add to it an edge.

The edge is a decent piece of all the entire strategy.having an edge does not necessarily mean that your strategy is profitable.

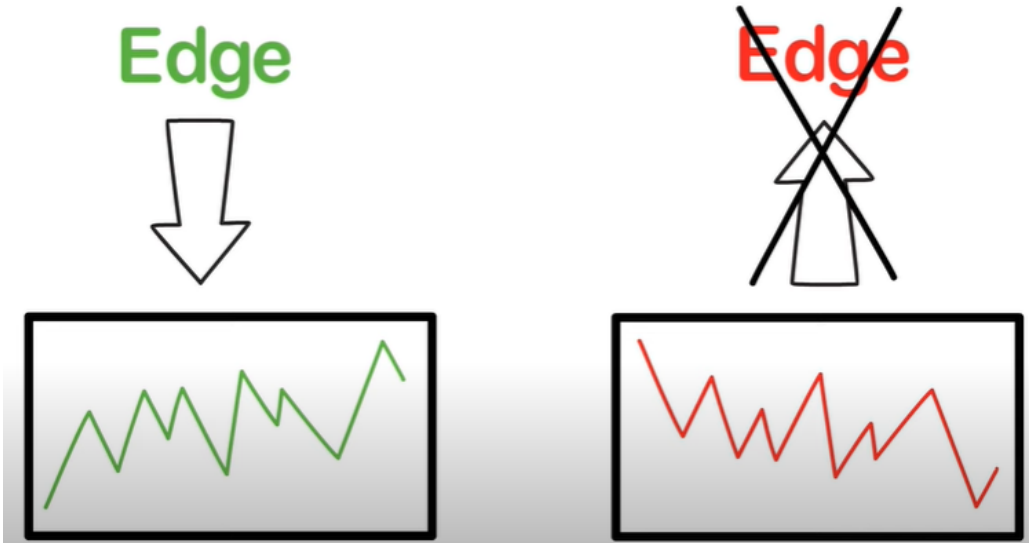

Figure 22 : *Edge and non edge chart*

On quantworld, profitability is not our main aim .Whereas we have to end with having an edge by comparing our strategy to a relevant benchmark (usually is some index in a sector like S&P 500).

Figure 23 : *Difference between strategy and benchmark*

you must also look for risk and volatility coming with this strategy. Risk is adjusting your return as a great deal of comparability.

Figure 24 : *Edge chart*

In step we must define α (edge) by testing different hypotheses that are based on scientific manner. QuantConnect offers a lot of research papers or uses Quantpidia. Strategies have limited capacity.

2. **Coding and implementation**

After you make a strategy , there is an implement and coding part depending on the skill of each one :

- Basic (IF X Buy , else sell)
- intermediate (adaptive trading algorithm)
- Advanced (machine learning bot)

3. **Backtesting**

Backtesting basically allows you to analyze how strategy will perform if we use old data. Lean engine is powerful tool to test your strategy without risking any real money

Note : Backtesting Results < Real Results

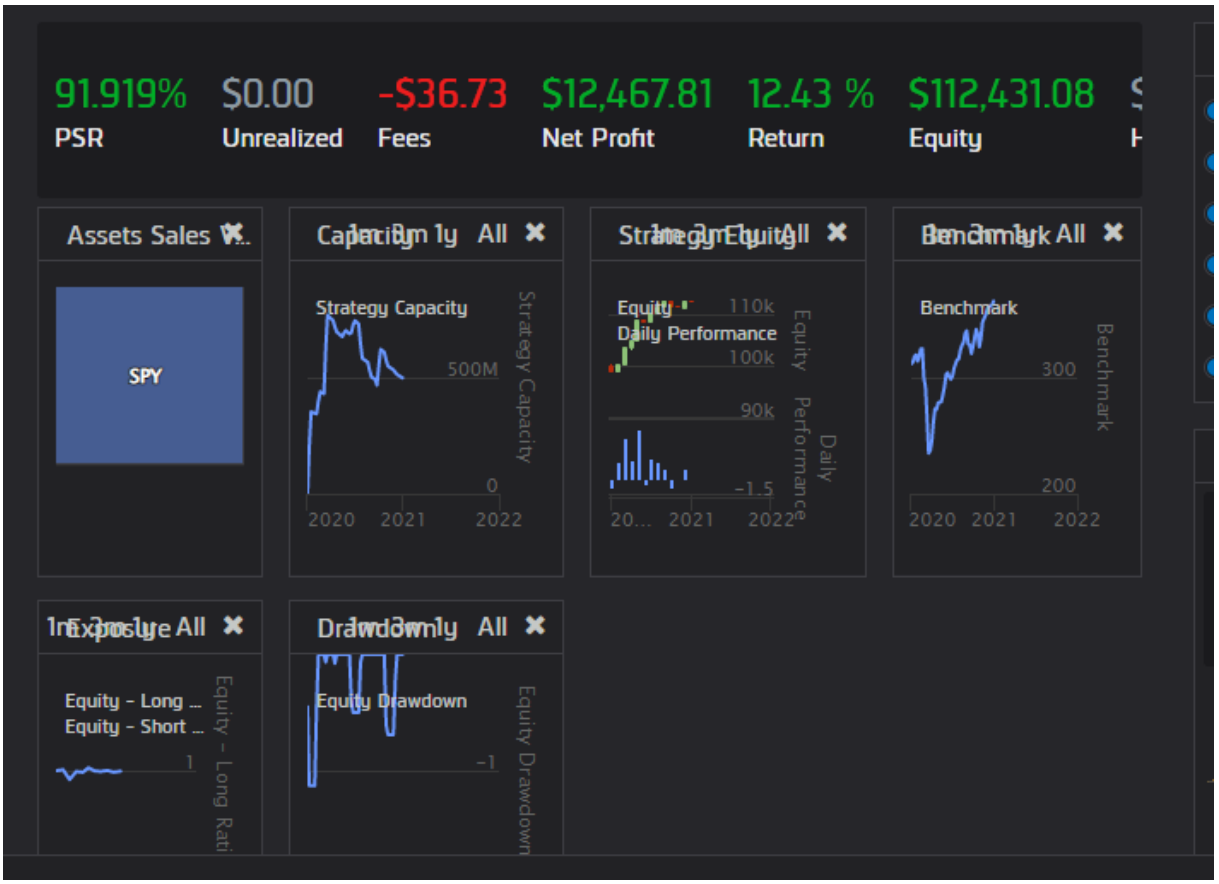

Figure 25 : *Result of analysis*

The Full steps to create an algorithm strategy in the real world .

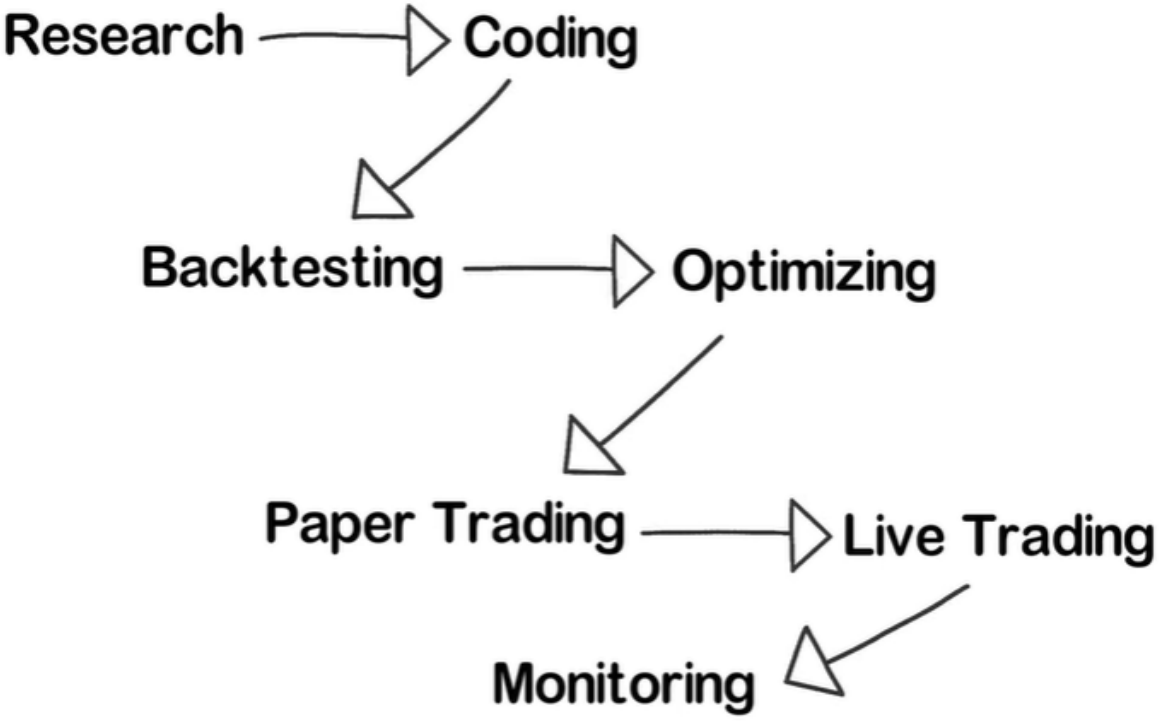

Figure 26 : *Full steps to create a trader algorithm*

I.2 Backend and database

FastAPI is a modern, fast (high-performance), web framework for building APIs with Python 3.6+ based on standard Python type hints[7].FastApi has many features :

- Automatic docs (Swagger UI ,ReDoc) instead of using postman :
  - check all your routers
  - easy and simple testing
- Using Pydentic library
- Based on JSON Schema and OpenApi
- Community support
- Security and authentication :
  - HTTP Basic
  - OAuth2 (also with JWT tokens)
  - Api key in
    - Headers
    - Query parameters
    - Cookies , etc
- Dependency Injection Unlimited “plugs-ins” Testing .
- Starlette Features
  - WebSocket support
  - GraphQL support
  - in-process background tasks

- Startup and shutdown events
- SQL database support , NoSQL support

Figure 27 : *FastApi logo*

we have used a lot of packages such as :

- uvicorn handles server run or stop .
- SQLachemy handles databases .
- PYjwt helps to improve authentication , etc..
- Aiomysql help to connect to database and secure the connection

I.3 Frontend

Next js is a React frontend development web framework created by Vercel (formerly Zeit) that enables functionality such as server-side rendering and static site generation [7].

Server-side Rendering unlike a traditional React application where the entire application is loaded and rendered on the client, Next Js allows the first page to load directly from the server, which is great for SEO & performance.

other benefits :

- Easy page routing
- API Routes
- Out of the box typeScript & sass
- Static site generation
- easy deployment

Figure 28 : *Next.js logo*

we have used a lot of packages such as :

- React Table handles the table and are easy to use.
- Formik handles the input form and error .
- Yup validation input .
- Bootstrap handle design
- Axios handles request Api.
- Next-auth handles authentication ,etc.

```
"dependencies": {
  "axios": "^0.26.1",
  "bootstrap": "^5.1.3",
  "chart.js": "^3.7.1",
  "formik": "^2.2.9",
  "jwt-decode": "^3.1.2",
  "jwt-encode": "^1.0.1",
  "mysql": "^2.18.1",
  "next": "12.1.0",
  "next-auth": "^3.27.3",
  "react": "17.0.2",
  "react-bootstrap": "^2.2.1",
  "react-dom": "17.0.2",
  "react-google-recaptcha": "^2.1.0",
  "react-icons": "^4.3.1",
  "react-pure-loaders": "^3.0.1",
  "react-script": "^2.0.5",
  "react-table": "^7.7.0",
  "react-ts-tradingview-widgets": "^1.0.9",
  "typescript": "^4.6.3",
  "yup": "^0.32.11"
},
```

Figure 29 : All frontend dependencies

I.4 Third-parties Service

1. **Google Sign in Button**

Google Sign-In manages the OAuth 2.0 flow and token lifecycle, simplifying your integration with Google APIs. A user always has the option to revoke access to our application at any time .

2. **Facebook Sign in Button**

Facebook Login is a fast and convenient way for people to create accounts and log into our application .

3. **Google drive**

Google Drive API is a tool that allows us to create applications leveraging Drive cloud storage. By this feature we can develop applications integrating with Google Drive and create powerful functionality in our application.

4. **Google reCAPTCHA (v3)**

reCAPTCHA protects your website from fraud and abuse without creating friction. It uses an advanced risk analysis engine and adaptive challenges to keep malicious software from engaging in abusive activities on your website. Meanwhile, legitimate users will be able to login, make purchases, view pages, or create accounts and fake users will be blocked

5. **YAHOO API**

The API serves real-time and historical data for crypto and stock markers. It provides extensive financial data for public companies, mutual funds, etf, bonds, crypto currencies, and national currencies, including option chains and market analysis.

I.5 Web Application Architecture

By using Next.js we have two types of architecture depending on our code :

1. **Server side render**

Server-side rendering (SSR) is an application's that converts HTML files on the server into a fully rendered HTML page for the client

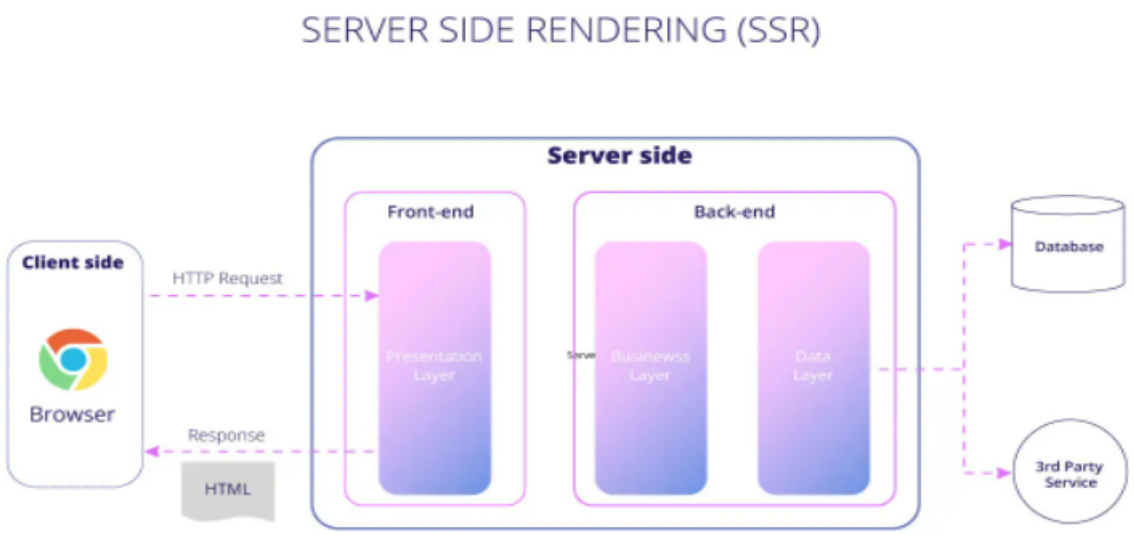

Figure 30 : *SSR Architecture*

2. Static side generation

Static-Site Generator (SSG) is a program or a tool used to generate static HTML websites and pages based on raw data and templates. Static-Site Generator automates the process of having to code HTML pages manually

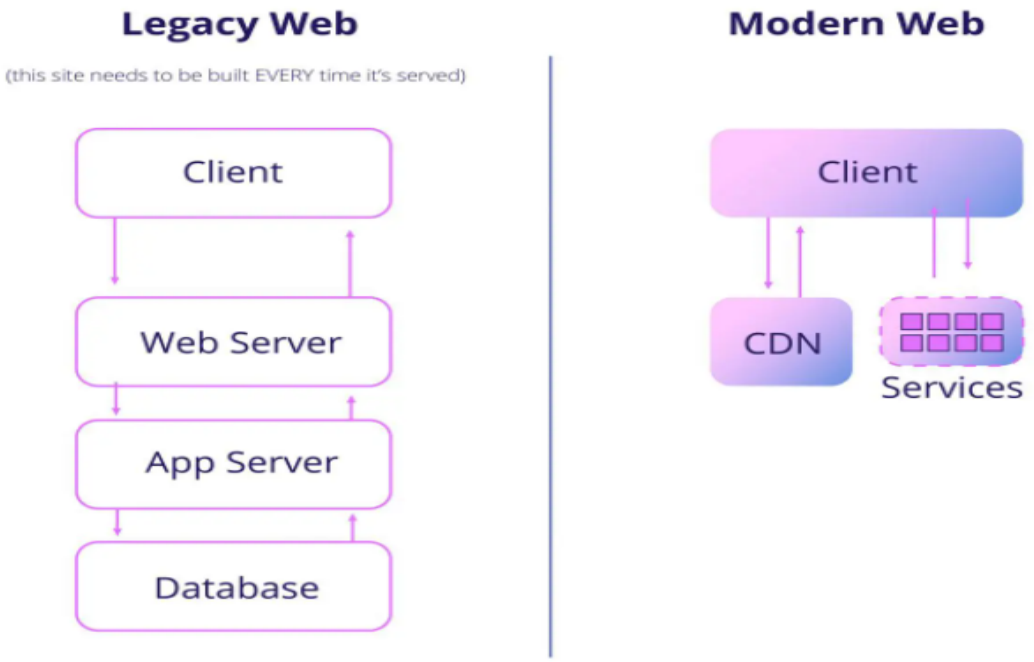

Figure 31 : *SSG Architecture*

I.6 Websocket and RestApi

the basis of comparison	Websocket (WS/WSS)	Rest api (HTTP/HTTPS)
HTTP	The use of HTTP occurs in the initial connection.	HTTP is a common protocol in RESTful web services.
Communication	Bi-directional in nature.	Uni-directional in nature.
nature	Socket-based concept.	Resources based concept, rather than commands.
Scenario	Real-time chat application.	having various requests.
Dependency	Rely on IP address and port number.	Based on the HTTP protocol and uses HTTP methods to relay data.
Cost	The cost of communication is lower.	The cost of communication is comparatively higher than WebSocket.
Performance	Better with high loads.	Great for occasional communication.
State	WebSocket is a stateful protocol.	REST is based on HTTP, which is a stateless protocol.

Table 8. *Difference between HTTP and WS*

I.7 Development Tools

1. Hardware part

	Laptop 1	**Laptop 2**
Brand	Lenovo	Hp
CPUs	intel core i5 8th generation	intel core i3 7^{th} generation
GPUs	• Nvidia Geforce MX 110 • Intel UHR Graphic 620	• Nvidia Geforce MX 110 • Intel UHR Graphic 620
RAM	8 go	4 go
OS	Microsoft Windows 11 Home	Microsoft Windows 10 professional

Table 9*. Hardware part*

2. **Software part**
   - **Vscode**
     Vscode is the integrated development environment used throughout the development process, it's highly extensible ide and it supports many programming languages.

   - **Xampp**
     XAMPP is completely free and easy to install Apache distribution containing MariaDB, PHP, and Perl.

   - **Opera**
     Opera is a free web browser and used to develop our application .Opera has great and powerful tools .

   - **Postman**
     Postman is a tool used for rapid testing of an api by generating custom http requests.

   - **Github**
     Github is a web application used to deploy git projects on the cloud.

   - **QuantConnect platform**
     QuantConnect is an open-source, cloud-based algorithmic trading platform for equities, FX, futures, options, derivatives and cryptocurrencies

   - **Visual paradigm online**
     Online tool suite that comprises infographic maker, EBook maker, chart maker, collage maker and more. Create stunning visuals, UML, BPMN, ArchiMate online.

   - **Heroku**
     Heroku is a platform as a service (PaaS) that enables developers to build, run, and operate applications entirely in the cloud.

   - **Vercel**

Vercel combines the best developer experience with an obsessive focus on end-user performance. Our platform enables frontend teams to do their best work.

II. Project interface

1) Loading interface

Loading pages are screens that users see when the application they are using is being loaded. They are important because they help users focus on the loading progress, not on the loading time. It's also a confirmation that things are moving forward, and the application is going to work.

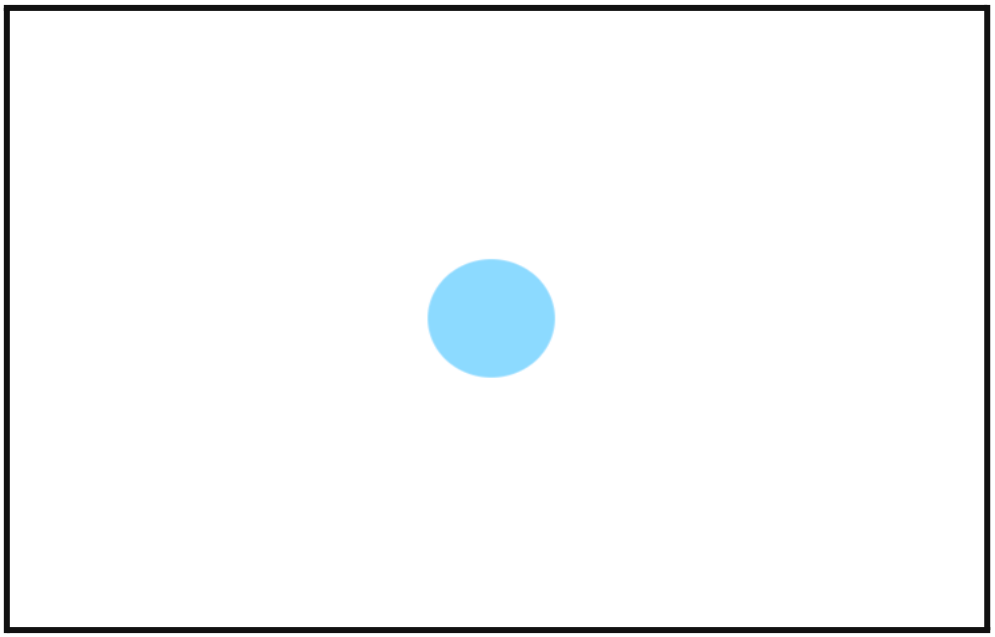

Figure 32 : *Loading interface*

2) Login interface

A login page is a web page or an entry page to a website that requires user identification and authentication, regularly performed by entering a username and password combination.

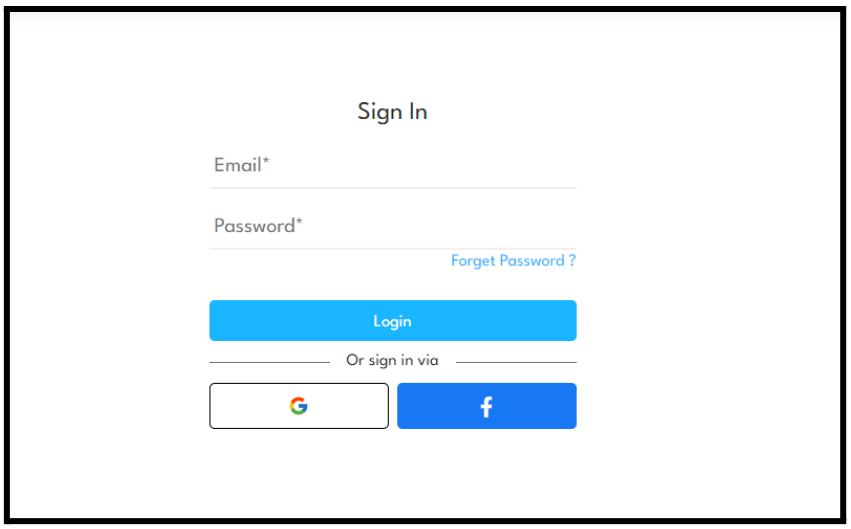

Figure 33 : *Login interface*

3) Register interface

Register interface enables users to independently register and gain access to your system

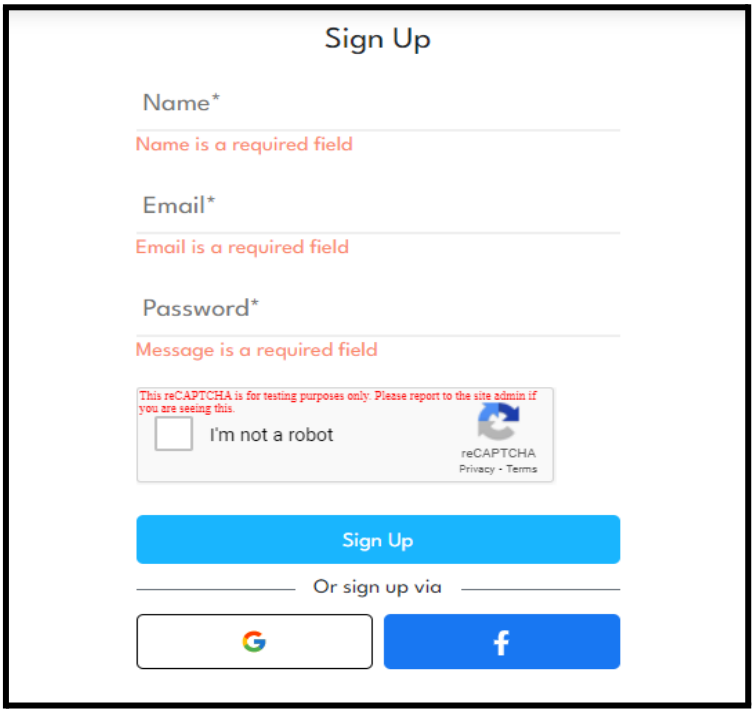

Figure 34 : *Register interface*

4) Dashboard interface

A dashboard interface is a visual display of all our data. While it has many usesand can be used in different ways . Its primary goal is to provide information at a glance.

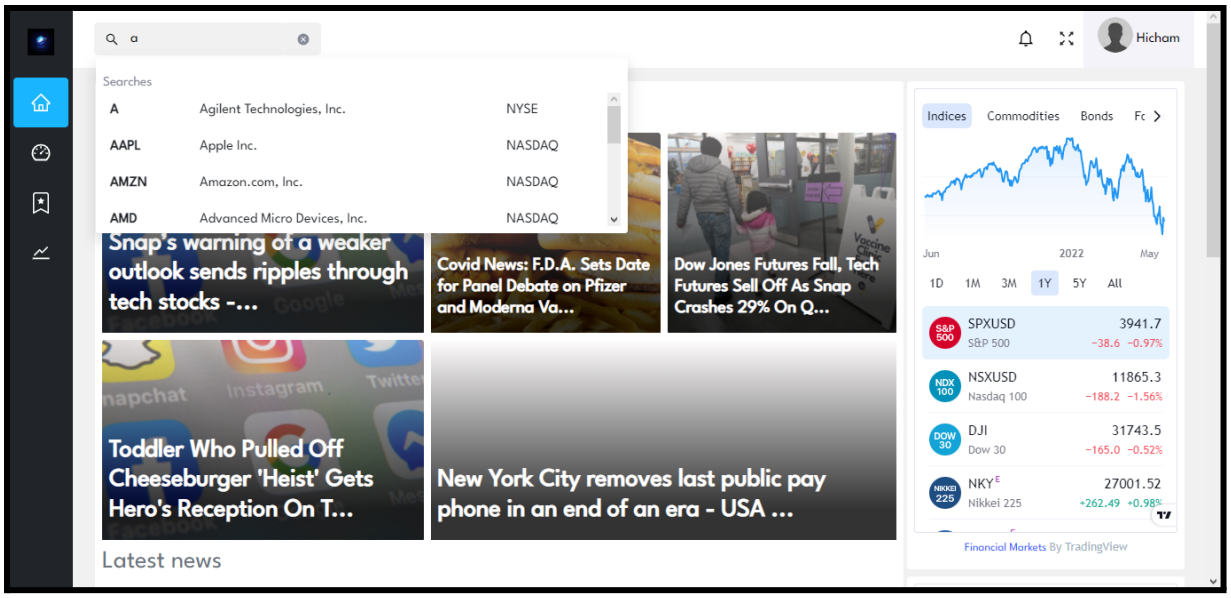

Figure 35 : *Dashboard interface*

5) Setting interface

The page settings are used to change the settings of document pages and other data, including: change background,change password, update or add public data … .

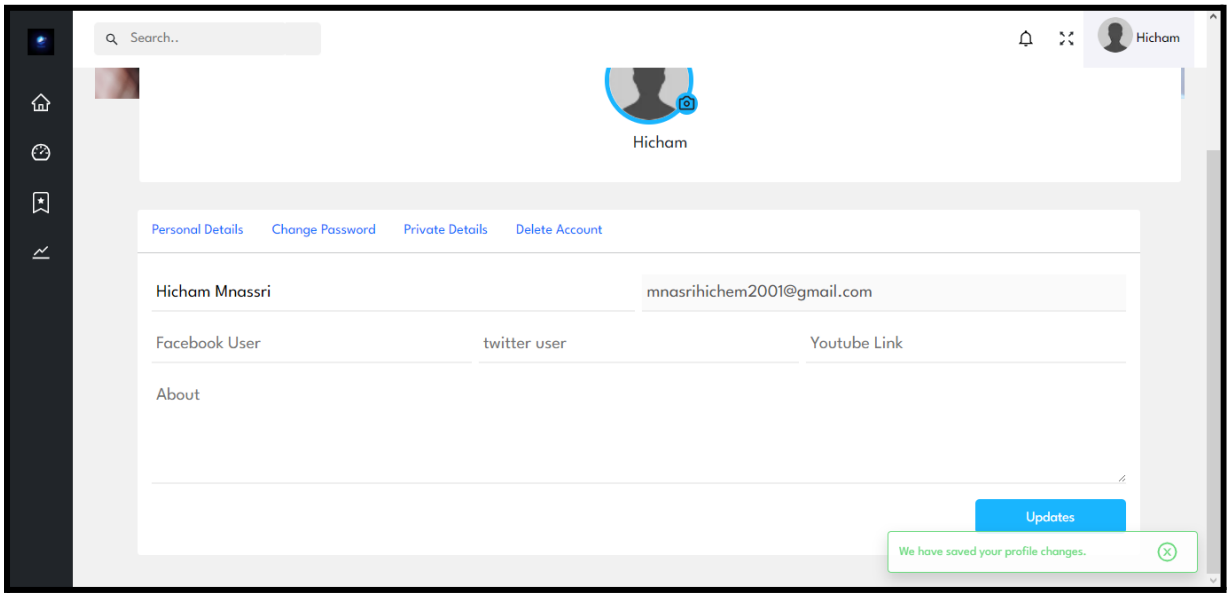

Figure 36 : *Setting interface*

6) Reset password email interface

Email is the best way to reset a user's password because it's frictionless.Customers consider email as a quick and easy way to reset a password .

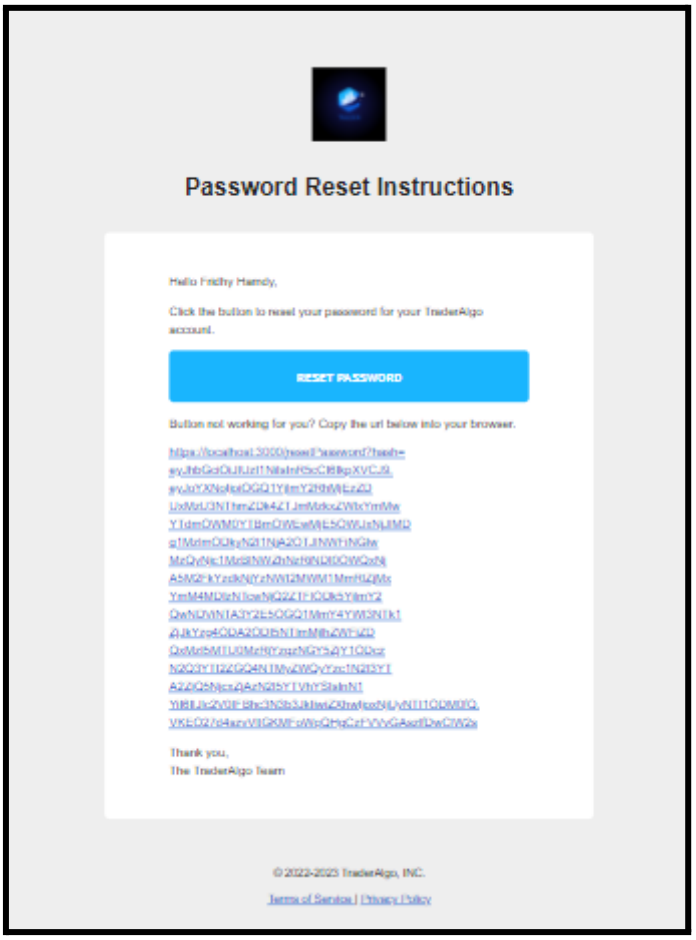

Figure 37 : *Reset password email interface*

III. Difficulties and learned lessons

III.1 Difficulties

While developing our application we have met some problems :

- It was hard to find a way to purchase the services we need . The free version is good for development but has limited requests . Yahoo api give us 100 requests per day , we could not complete the notification page, we haven't only written the code and we haven't done a test .
- Poor internet connection . In our application, the performance is most important aspect

- Quantconnect does not have a big community . that's why we met an error it was hard to be fixed .
- The given periode to achieve our project it is not sufficient to learn trading and to make a good trader algorithm as we stated previously.
- We do not have enough experience in trading and quant trading so we work with the basic knowledge .

III.2 learned lessons

In the last 4 months we have learned some valuable lessons : How to manage time and work within a team . We have learned trade is one of the hardest jobs but we have had fun .We have also learned about stock market :

- What is trading and market ?
- What are shares and equities ?
- Technical analysis and indicator .
- Pattern and shape of chart .
- Read the rapport of the company and many more .

Conclusion

In this chapter we have dealt with different technical choices ,the hardware and software we have used to build our application , displaying the interfaces and finally state the difficulties we have faced all along our project .

General Conclusion

Our project is an attempt to minimize the gap between the retail investment and the investment companies . In this context , we have firstly dealt with the project framework ,then with the requirement gathering, to the detailed conception and finally ended with the first trial to realize our own application "TraderAlgo".

The main aim is to empower individuals to invest. Same as “QuantConnect”, built into the core motto “Democratizing Finance, Empowering Individuals”,we have tried as beginners non experienced in finance to provide tools for investment .

Our expected work was about making a sophisticated algorithm , but for the lack of sufficient knowledge and the shortage of time, we have narrowed our plan of action to what it is this project .

To invest in stocks , options , futures , bonds and cryptocurrencies you don’t need a lot of money . What you necessarily need is having knowledge and information since the world of investment wold is not that easy as many think .

Last but not least our project is just an attempt individual investors to learn more about investment to make that easy for them .

Webography

[1] wikipedia,“investing.com”,[Online].Available:https://en.wikipedia.org/wiki/Investing.com

[2] wikita,”tradingView”,[Online].Available:TradingView - Wikitia

[3],[4]toolshero,”Rup”,[Online].Available:https://www.toolshero.com/information-technology/rational-unified-process-rup/

[5] uml-diagram.org,”Sequence diagram “authentication using facebook account”” ,[Online].Available:https://www.uml-diagrams.org/facebook-authentication-uml-sequence-diagram-example.html

[6] wikipedia,"QuantConnect",[Online].Available:https://en.wikipedia.org/wiki/QuantConnect

[7]fastapi,”fastapi”,[Online].Available:https://fastapi.tiangolo.com

BASIC TECHNICAL ANALYSIS

Fridhy Hamdi
Mnassri Mohcen

الخلاصة : يتكون الحل المقترح هنا من منصة ويب تقوم بجمع الأخبار والمعلومات وإجراء تحليل تقني بسيط للأسهم. لبناء مشروعنا ، استخدمنا مجموعة متنوعة من التقنيات مثل FastApi و Next.js و SQLalchemy و QuantConnect و visual paradigm online

الكلمات المفاتيح: FastApi , Next.js , SQLalchemy , QuantConnect

Résumé : La solution proposée ici consiste en une plateforme web qui collecte les nouvelles et les informations et effectue une analyse technique simple pour les actions. Pour construire notre projet, nous avons utilisé une variété de technologies telles que FastApi , Next.js , SQLalchemy , QuantConnect et visual paradigm online.

Mots clés : FastApi , Next.js , SQLalchemy , QuantConnect

Abstract : The solution proposed here consists of a web platform that collects the news and information and makes a simple technical analysis for stock . To fulfill our project we have used a variety of technologies such FastApi , Next.js , SQLalchemy , QuantConnect and visual paradigm online.

Key-words : FastApi , Next.js , SQLalchemy , QuantConnect