

IT 592 – Graduation Project

Final Report

Project Title: A Financial Analysis Reporting Tool

Group Members:

Gizem TEKİNDAG

Gökhan GÜNAL

Supervisor(s):

Ahmet DEMİRELLİ

Altuğ TANALTAY

Date: 02.09.2022



1. EXECUTIVE SUMMARY

Financial data of publicly traded companies are available on the websites of regulatory and supervisory public institutions. However, it will be difficult for a credit rating analyst and an average investor trying to make financial decisions to see and interpret the financial data of almost all companies over the years.

For a credit rating analyst, the financial data of the companies are important for the healthy rating of a company. At the end of this evaluation, a rating is issued for the firm and the firm's credibility is measured. If the company gets a successful credit score, it will gain an advantageous position in the commercial relations it will enter into. Therefore, it is very vital for companies to get a decent credit rating.

On the other hand, investors determine their investment decisions by reading the financial data of the companies. If the financial data of companies could be easily read and compared, investors could make healthier decisions.

At this point, our starting point is to prevent people from making long and error-prone calculations, to save them time, to prevent them from drowning in complex word, excel, pdf files, and to present data that companies and industries could compare over the years. That's why we wanted to design a useful, user-friendly and interactive dashboard.

In this project, comparative financial data about companies and sectors will be given. Financial data that could give people an idea about companies and sectors will be presented and a comparative analysis of these data by years will be included. There are plenty of line graphs, bar graphs and pie charts in our project so that people could easily interpret financial data of the companies. We used Python and Plotly Dash together in the design of this dashboard. In addition, to save some data, we connected to the MySQL application via Python.

2. PROBLEM STATEMENT

Making easier the financial analysis, avoid from the miscalculations and the complexity the interpretation of the financial items. On the internet people could find companies balance sheet, income statement and other financial information, in order to understand these items, you should make calculations or compare them.

In order to avoid miscalculations, operational risks and loss of time, we are going to provide financial statement analysis for users. These analyses contain period based and sectoral comparison for each company. For the reasons stated above we developed a web application which is automatically generated our database. In this way, users will be able to access the analysis results directly without manual calculations.

2.1. Objectives/Tasks

Financial Analysis Dashboard developed as a web application. These are the objectives:

A. Database Creation

B. Retrieve Data from The Database

C. Basic Authentication

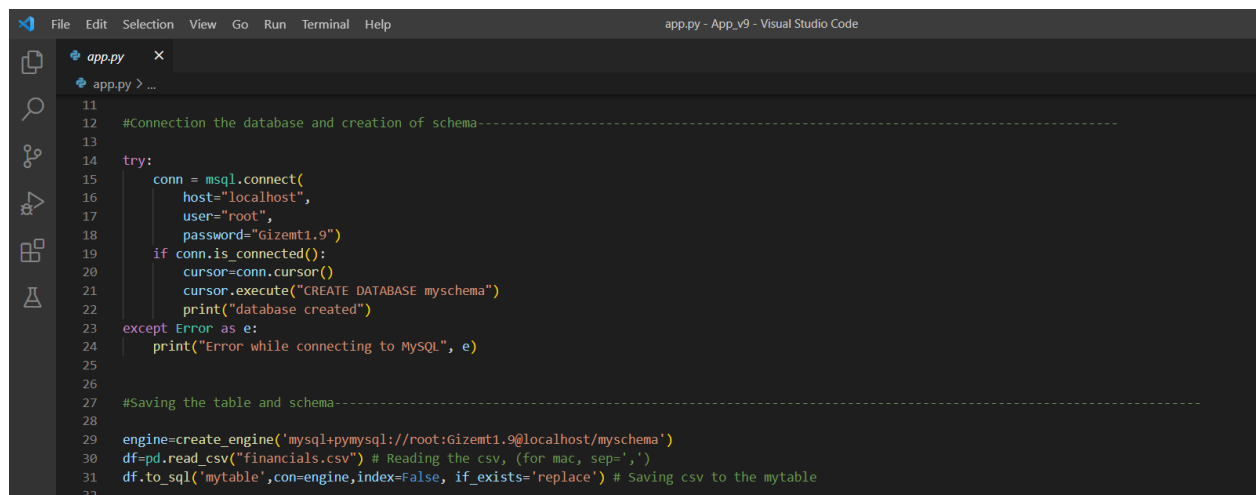
D. Financial Analysis Dashboard

A. Database Creation

We have financial data about the companies in the csv file. With the python we could save this information through the database. We have two schemas, one for financial data one for the users.

A.1. Financial Item Schema

We save financial data of the companies in the “mytable” this schema through the Python code which we show below.



```
11
12 #Connection the database and creation of schema-----
13
14 try:
15     conn = mysql.connect(
16         host="localhost",
17         user="root",
18         password="Gizemt1.9")
19     if conn.is_connected():
20         cursor=conn.cursor()
21         cursor.execute("CREATE DATABASE myschema")
22         print("database created")
23 except Error as e:
24     print("Error while connecting to MySQL", e)
25
26
27 #Saving the table and schema-----
28
29 engine=create_engine('mysql+pymysql://root:Gizemt1.9@localhost/myschema')
30 df=pd.read_csv("financials.csv") # Reading the csv, (for mac, sep=',')
31 df.to_sql('mytable',con=engine,index=False,if_exists='replace') # Saving csv to the mytable
32
```

Entities:

- Company Code
- Company Name
- Sector
- Year
- Total_Assets
- Total_Liabilities
- Liabilities
- Shareholders_Equity
- Gross_Income
- Profit

MySQL Workbench interface showing a query result for 'mytable'. The query is: `SELECT * FROM myschema.mytable;`

Company_Code	Company_Name	Sector	Year	Total_Assets	Total_Liabilities	Liabilities	Shareholders_Equity	Gross_Income	Profit
CMSA	Cmsa Cimento	Industry	2017	3242449563	3242449563	1820099752	1422349811	394889849	228351942
CMSA	Cmsa Cimento	Industry	2018	3483989356	3483989356	2032509976	1451479380	405980614	154925532
CMSA	Cmsa Cimento	Industry	2019	3795930288	3795930288	2310167049	1485763239	273750859	13169480
CMSA	Cmsa Cimento	Industry	2020	5129182413	5129182413	3290338076	1838844037	467752689	175746242
CMSA	Cmsa Cimento	Industry	2021	5248764237	5248764237	2475910129	2772854068	712959657	1016600770
DERIM	Derimod	Industry	2017	319050617	319050617	275725817	43324800	21793929	723269
DERIM	Derimod	Industry	2018	288532426	288532426	247752902	40779524	19793939	-723340
DERIM	Derimod	Industry	2019	331512393	331512393	254199519	77312874	26997171	-6326020
DERIM	Derimod	Industry	2020	320143177	320143177	248332078	71811099	20193149	-5339613
DERIM	Derimod	Industry	2021	393015779	393015779	319244559	73771220	54751468	2730784
DESA	Desa	Industry	2017	252664515	252664515	182747883	69916632	100494028	-186146
DESA	Desa	Industry	2018	279267716	279267716	2037749811	75402725	188614201	8378556
DESA	Desa	Industry	2019	369203380	369203380	275476251	93727129	161843391	18243208
DESA	Desa	Industry	2020	353866719	353866719	259282454	94584265	115860660	1646934

A.2. Users Schema

In this schema we keep the user's information in the "usertable". We create this schema and this table via MySQL.

```

1 CREATE SCHEMA 'userschema' ;
2 CREATE TABLE 'userschema'.'usertable' (
3   'userid' INT NOT NULL,
4   'username' VARCHAR(45) NOT NULL,
5   'password' VARCHAR(45) NOT NULL,
6   PRIMARY KEY ('userid', 'username'));
```

Entities:

-id

-username

-password

```

1 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('1', 'giztekindag', 'gizemsifre');
2 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('2', 'gokhangunal', 'gokhansifre');
3 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('3', 'kubrakoseoglu', 'kubrasifre');
4 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('4', 'orkunyerbasan', 'orkunyerbasan');
5 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('5', 'aykutbattal', 'aykutsifre');
6 INSERT INTO 'userschema'.'usertable' ('userid', 'username', 'password') VALUES ('6', 'anilmenis', 'anilsifre');
```

MySQL Workbench

Local instance MySQL80 x

File Edit View Query Database Server Tools Scripting Help

mytable mytable usertable x

Limit to 1000 rows

```
1 • SELECT * FROM userschema.usertable;
```

Result Grid

	userid	username	password
▶	1	gizemtekindag	gizemsifre
	2	gokhangunal	gokhansifre
	3	kubrakoseoglu	kubrasifre
	4	orkunyerbasan	orkunyerbasan
	5	aykutbattal	aykutsifre
	6	anilmemis	anilsifre
*	NULL	NULL	NULL

B. Retrieve Data from The Database

B.1. Retrieve The Financial Items

We retrieve the data though Python. By this way we have dataframe which could we use for the visualization. We also calculate financial ratio's when we create dataframe.

```
File Edit Selection View Go Run Terminal Help pg1_firm_analysis.py - App_v9 - Visual Studio Code
pg1_firm_analysis.py X
pages > pg1_firm_analysis.py > ...
18 veritabani=mysql.connect(
19     host="localhost",
20     user="root",
21     password="Gizem1.9",
22     database="myschema"
23 )
24
25 yeni=veritabani.cursor()
26 yeni.execute("select*from myschema.mytable")
27
28 data1=[]
29
30 for i in yeni:
31     data1.append(i)
32
33 all_data1 = pd.DataFrame(data1, columns = ['Company_Code','Company_Name','Sector','Year','Total_Assets','Total_Liabilities','Liabilities','Shareholders_Equity',
34     'Gross_Income','Profit'])
35
36 all_data1 = all_data1.reindex(columns = all_data1.columns.tolist()+["Financial_Leverage_Ratio","Debt_Equity_Ratio","Equity_Ratio","ROE","ROA"])
37
38 all_data1 = all_data1.assign(Financial_Leverage_Ratio = lambda x: round(x['Total_Assets']/x['Total_Assets'] - x['Shareholders_Equity']/x['Total_Assets'],2))
39
40 all_data1 = all_data1.assign(Debt_Equity_Ratio = lambda x: round(x['Total_Assets']/x['Shareholders_Equity']-x['Shareholders_Equity']/x['Shareholders_Equity'],2))
41
42 all_data1 = all_data1.assign(Equity_Ratio = lambda x: round(x['Shareholders_Equity'] / x['Total_Assets'],2))
43
44 all_data1 = all_data1.assign(ROE = lambda x: round(x['Profit'] / x['Shareholders_Equity'],2))
45
46 all_data1 = all_data1.assign(ROA = lambda x: round(x['Profit'] / x['Total_Assets'],2))
47
```

```
File Edit Selection View Go Run Terminal Help pg1_firm_analysis.py - App_v9 - Visual Studio Code
pg1_firm_analysis.py X
pages > pg1_firm_analysis.py > ...
51 guncel=veritabani.cursor()
52 guncel.execute("SELECT * FROM myschema.mytable WHERE myschema.mytable.Year=(select max(year) from myschema.mytable)")
53
54 guncel_data=[]
55
56 for i in guncel:
57     guncel_data.append(i)
58
59 datas= pd.DataFrame(guncel_data, columns = ['Company_Code','Company_Name','Sector','Year','Total_Assets','Total_Liabilities','Liabilities','Shareholders_Equity',
60     'Gross_Income','Profit'])
61
62 datas = datas.reindex(columns = all_data1.columns.tolist()+["Financial_Leverage_Ratio","Debt_Equity_Ratio","Equity_Ratio","ROE","ROA"])
63
64 datas = datas.assign(Financial_Leverage_Ratio = lambda x: round(x['Total_Assets']/x['Total_Assets'] - x['Shareholders_Equity'] / x['Total_Assets'],2))
65
66 datas = datas.assign(Debt_Equity_Ratio = lambda x: round(x['Total_Assets']/x['Shareholders_Equity']-x['Shareholders_Equity']/x['Shareholders_Equity'],2))
67
68 datas = datas.assign(Equity_Ratio = lambda x: round(x['Shareholders_Equity'] / x['Total_Assets'],2))
69
70 datas = datas.assign(ROE = lambda x: round(x['Profit'] / x['Shareholders_Equity'],2))
71
72 datas = datas.assign(ROA = lambda x: round(x['Profit'] / x['Total_Assets'],2))
73
```

B.2. Retrieve The Users Information

We retrieve the data though Python. We create a dictionary for the username and password. By this way, usernames and passwords combined and matched.

```

veritabani=mysql.connect(
    host="localhost",
    user="root",
    password="Gizemt1.9",
    database="userschema"
)

yeni=veritabani.cursor()
yeni.execute("select*from userschema.usertable")

users=[]

for i in yeni:
    users.append(i)

usersDict={}

for d in range(len(users)):
    usersDict[users[d][1]]=users[d][2]

```

C. Basic Authentication

The financial analysis dashboard first open with the login form. Users could not reach the financial analysis tool if they do not have username and password.

When the users open the tool, they must sign in with the username and password. If the user type the wrong username or password, the sign in page is re-opening.

Sign in

http://127.0.0.1:8050

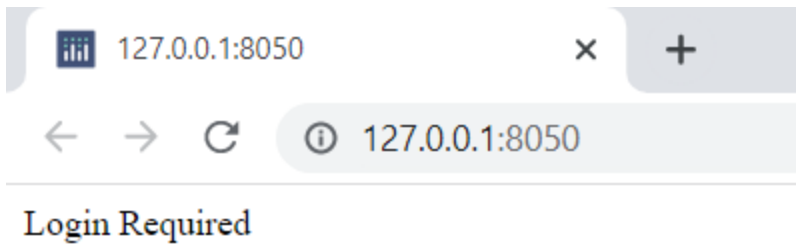
Username

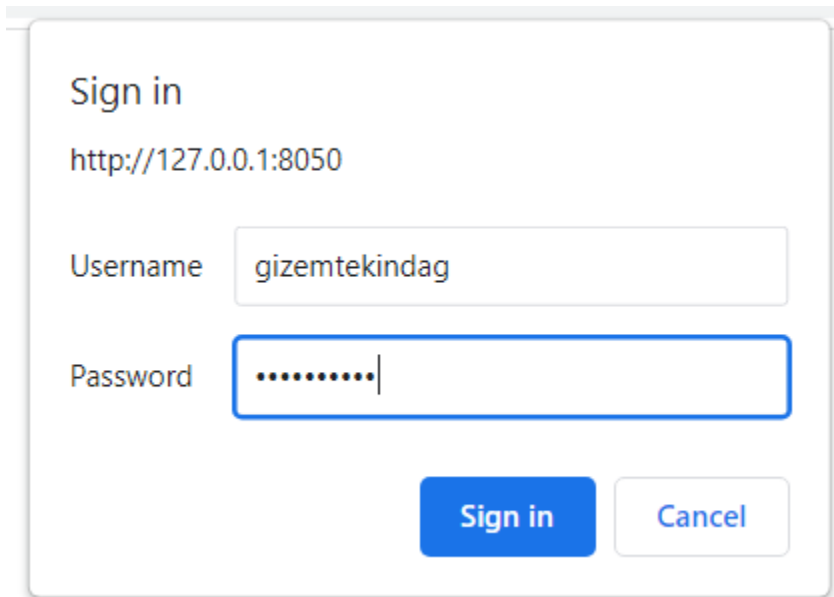
Password

Sign in

Cancel

If the user click the “Cancel” button, the tool will not open.



A sign-in dialog box with a title bar. The title is "Sign in". Below the title is the URL "http://127.0.0.1:8050". There are two input fields: "Username" with the text "gizemtekindag" and "Password" with masked characters ".....". At the bottom are two buttons: "Sign in" (blue) and "Cancel" (white with blue border).

If the user sign in with the correct password, the financial analysis dashboard will open.

D. Firm Analysis Dashboard

In our project users could found financial analysis and information about the companies which are active in one of the industry, banks, factoring and insurance sectors. Also, comparative analyzes about them.

A Financial Analysis Dashboard Tool is a multi-page web application. In this app have a navbar top on the page. The name of the tool seen on the left of the navbar, pages are the right of the navbar. All pages have same navbar, the only difference opened page's name color seems different according to the others.

The pages are listed below:

- Firm Analysis
- Comparison of Companies
- Sector Analysis

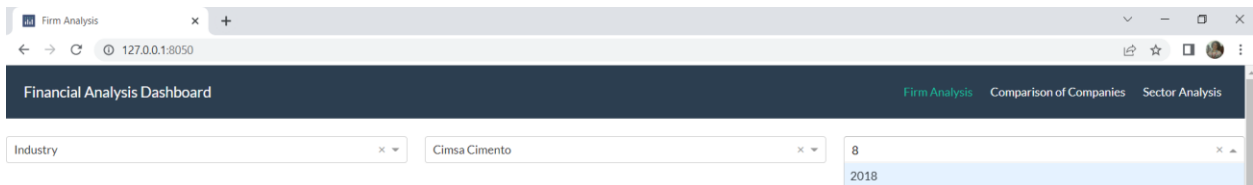
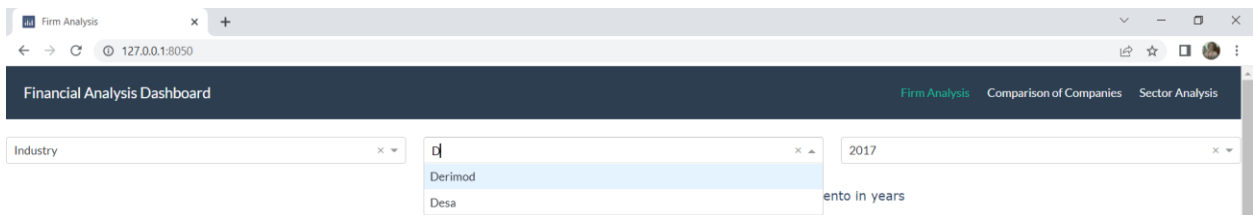
We want to explain our dashboard page to page.



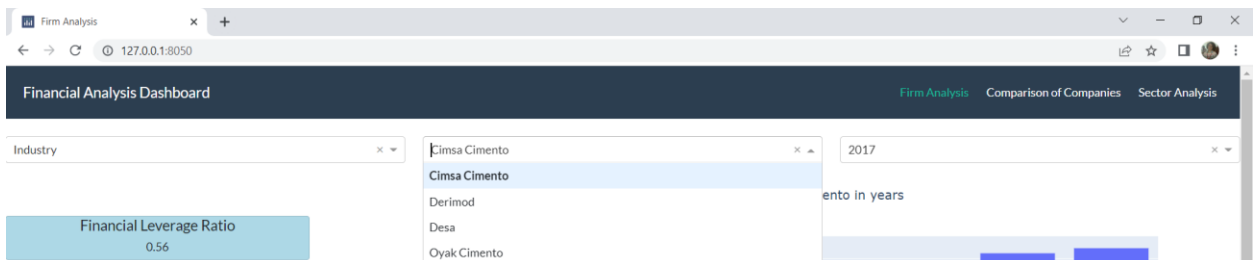
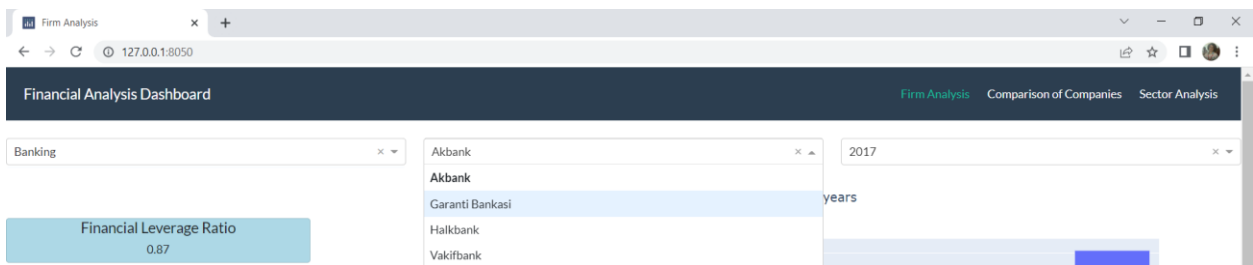


D.1. Firm Analysis

In the Firm Analysis page, we have three dropdowns on the top. The first dropdown is for sector, the second one is company and the third one is for year selection. Users could do search in the dropdown.

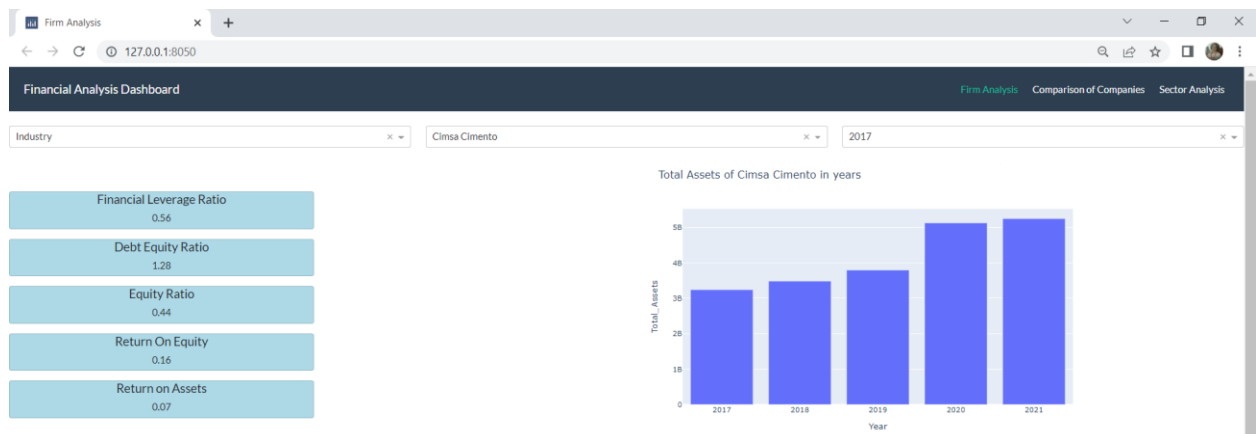


When the user chooses one of the sectors, in the second dropdown listed the companies which are operating the chosen sector.



In this page we have cards that shown information about selected company. There are five cards which gives information about the company's financial ratios for selected year in the third dropdown.

In addition to cards, we have also five charts for show the total assets, profit, shareholders' equity, total liabilities and indebtedness (liabilities) in five years. By this way, users could easily see the development of the companies in the five year and without the calculation could see the financial leverage ratio, debt equity ratio, equity ratio, return on equity and return on assets.

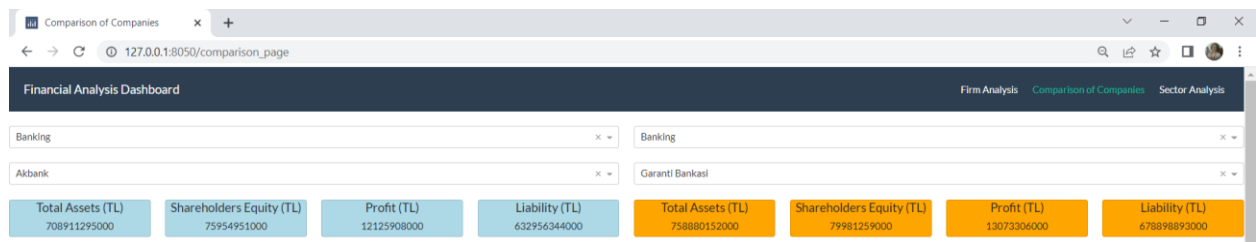


D.2. Comparison of Companies

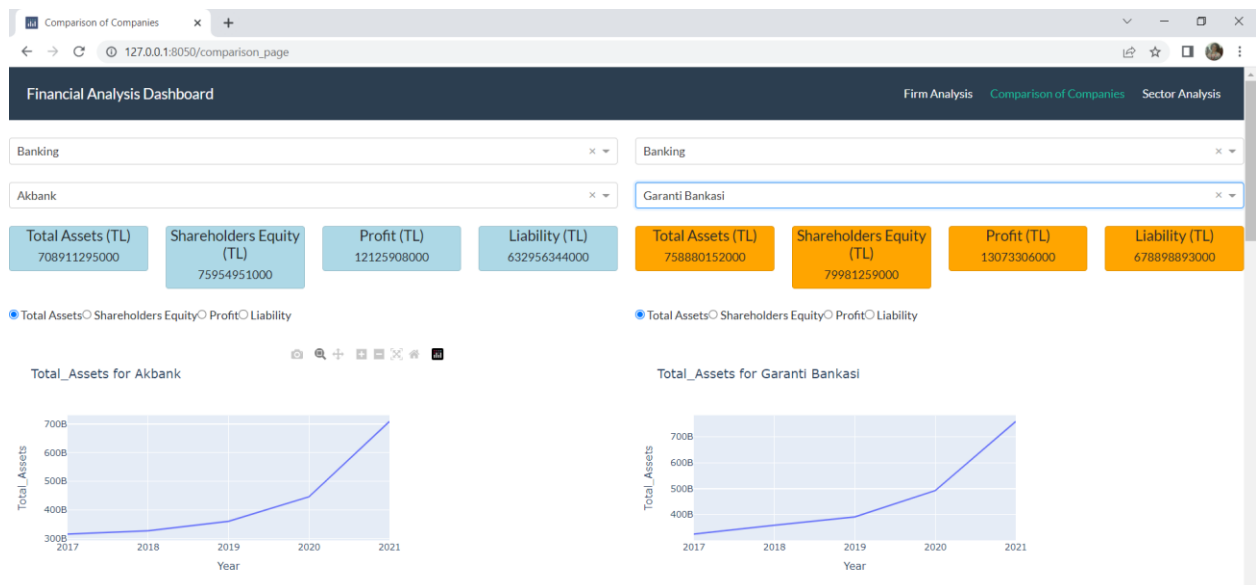
In the Comparison of Companies page, we divide the page in two sections vertically. There are four dropdowns, eight cards, eighteen radio-items, four line charts, four bar charts in total. User could compare two companies in this page easily.

Top of the sections there are dropdowns. Like in the firm analysis page dropdowns, these have same feature, the user could make a search. First dropdowns for the sector selection, second one is the company selection. When the user chooses one of the sectors, in the second dropdown listed the companies which are operating the chosen sector. The user chooses two companies which she/he wants to compare with dropdowns.

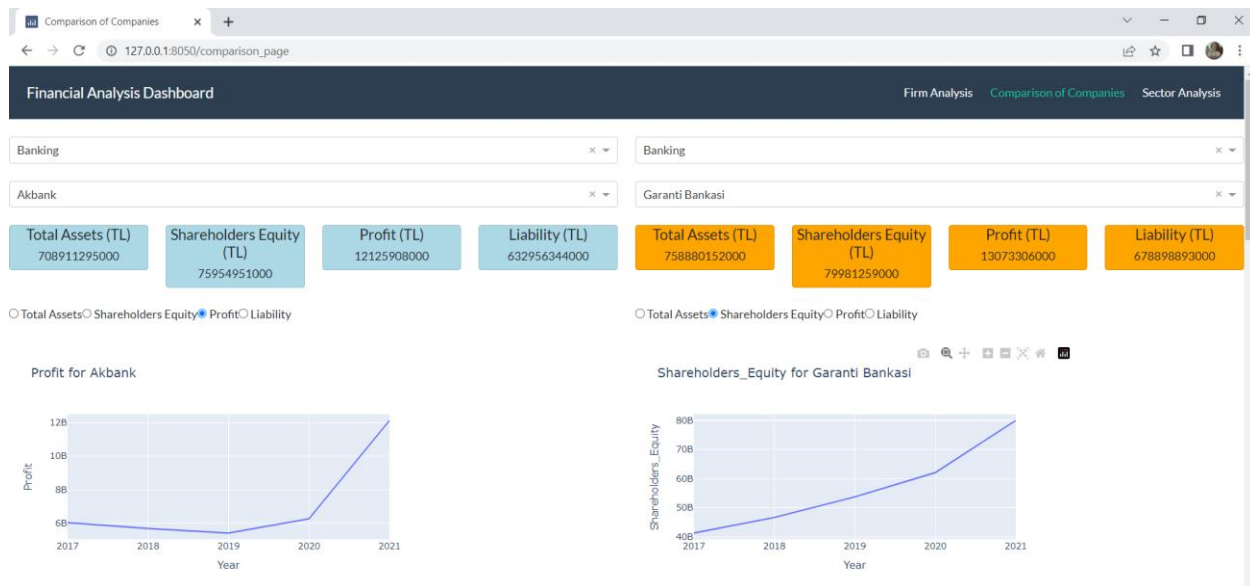
The asset, shareholders equity, profit values and liability of the companies are shown on the cards below the dropdowns. These values are from the closest end-year data.



Below the cards there are radio – items which are connected line charts. Via the radio – items users could choose which financial item they want to analyze throughout the years. The company information of these charts comes from the dropdowns.



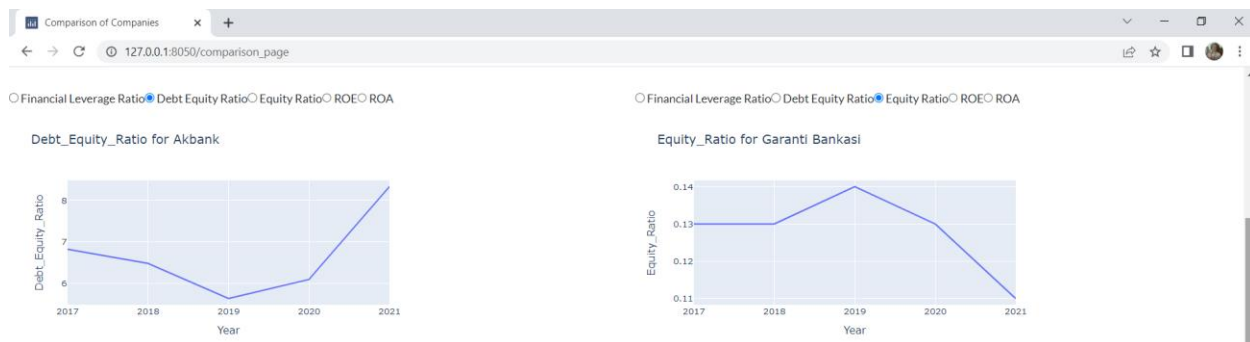
Regardless of sector, firm and financial item; the user could choose any value she/he wants to observe. By this way users could easily see the difference between companies which are selected at the dropdowns.



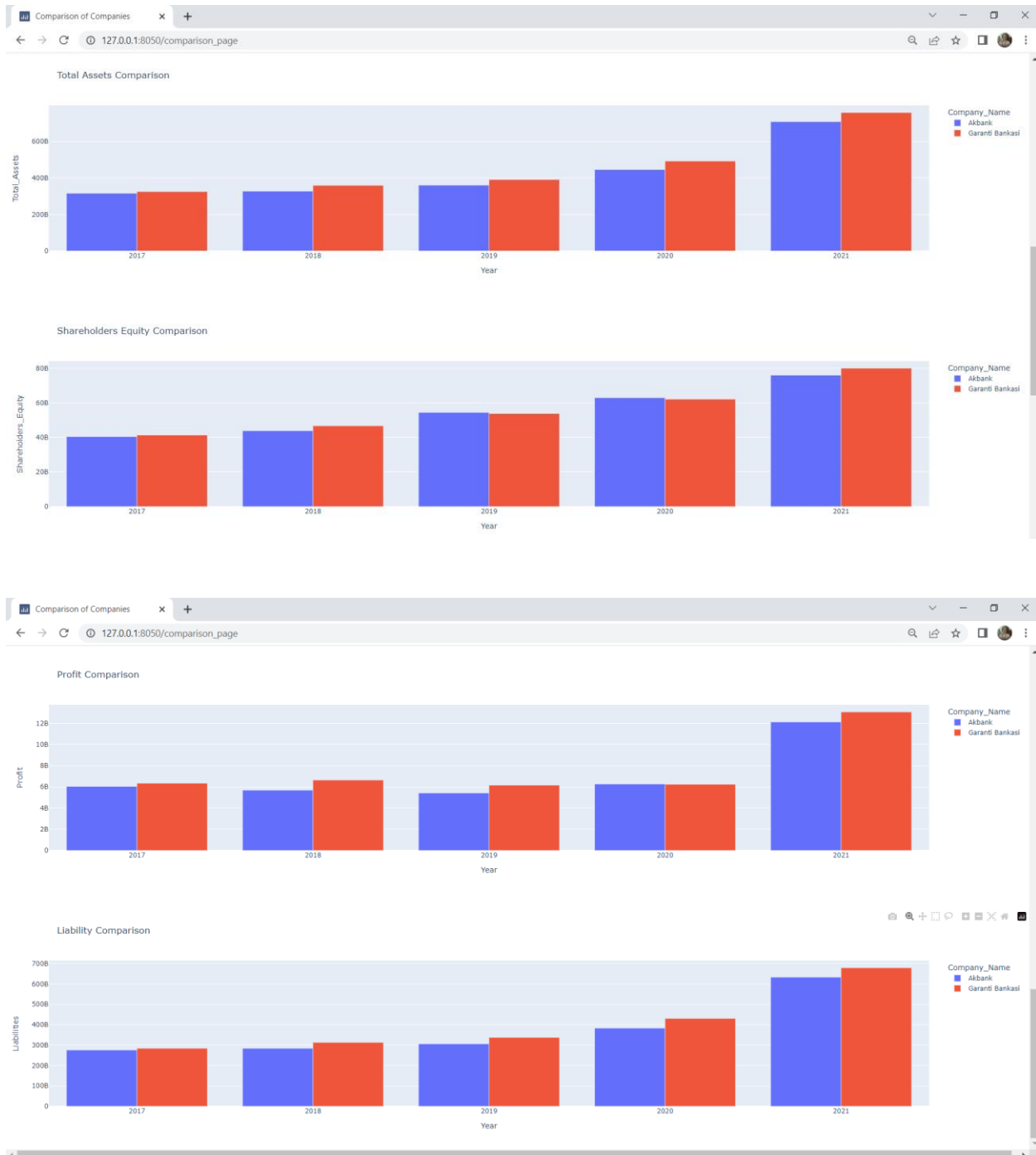
After these line charts there are again radio – items which are connected other line charts. Via the radio – items users could choose which financial ratio they want to analyze throughout the years.



Regardless of sector, firm and financial ratio; the user could choose any value she/he wants to observe. By this way users could easily see the difference between companies which are selected at the dropdowns.



Below these line charts, there are also bar charts. Total asset , shareholders equity, profit and liability bar charts show the users to the difference between the asset values more clearly and effciently. The year information of the bar charts is on the x axis and the asset/shareholders equity/profit/liability information is on the y axis. Company information comes from the dropdowns which are top of the sections. By this way the user compare the financial items year-based.



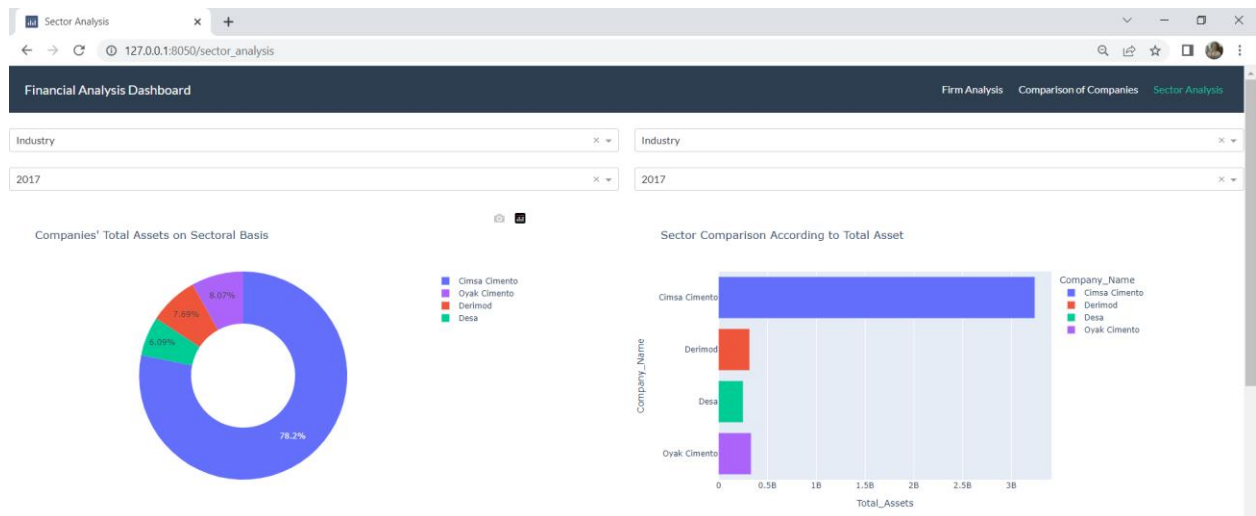
D.3. Sector Analysis

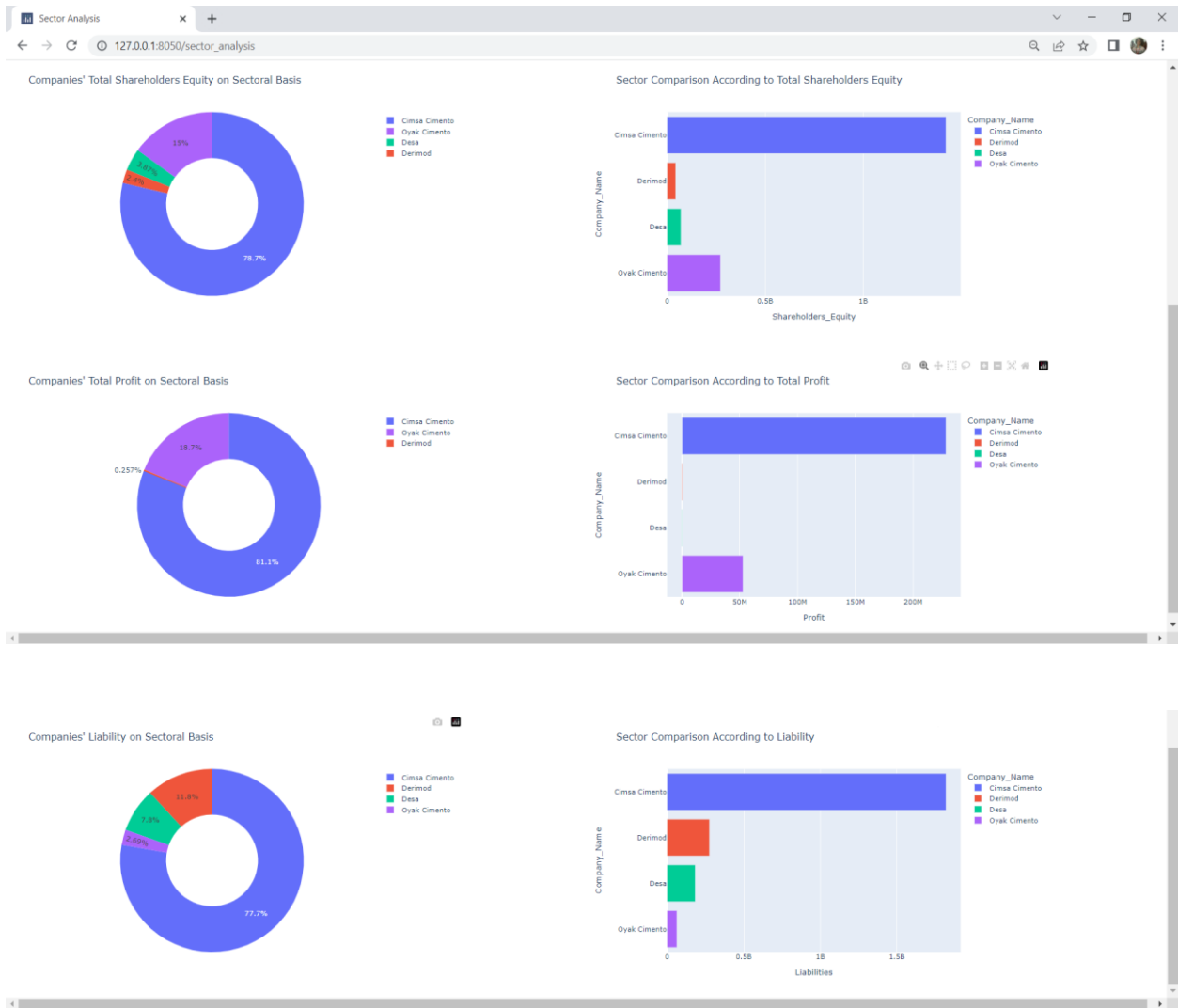
In the Sector Analysis page, we divide the page in two sections vertically. There are four dropdowns, four pie charts, four bar charts. Users could compare two sectors in this page easily.

Top of the sections there are dropdowns. Like in the other pages dropdowns, these have same feature, the user could make a search. First dropdowns for the sector selection, second one is the year selection. When the user chooses one of the sectors, in the second dropdown listed year value which used in the pie charts and the bar charts. Two dropdowns connected each other, by this way when the sector selected year value automatically comes. The user chooses two sectors which she/he wants to compare with dropdowns.

Left section of the page there are pie charts. Right section of the page there are bar charts. The user free to select the sector both sections. She/he could make the comparison however she/he wants; the selection of sector and year is completely left to the user's choice.

The pie charts show the user, the percentage distribution of the assets, shareholders equity, profit and liability of the companies operating in the selected sector in the dropdown. The bar charts allow the user to see the assets, shareholders equity, profit and liability of the companies operating in the selected sector in dropdown quantitatively.

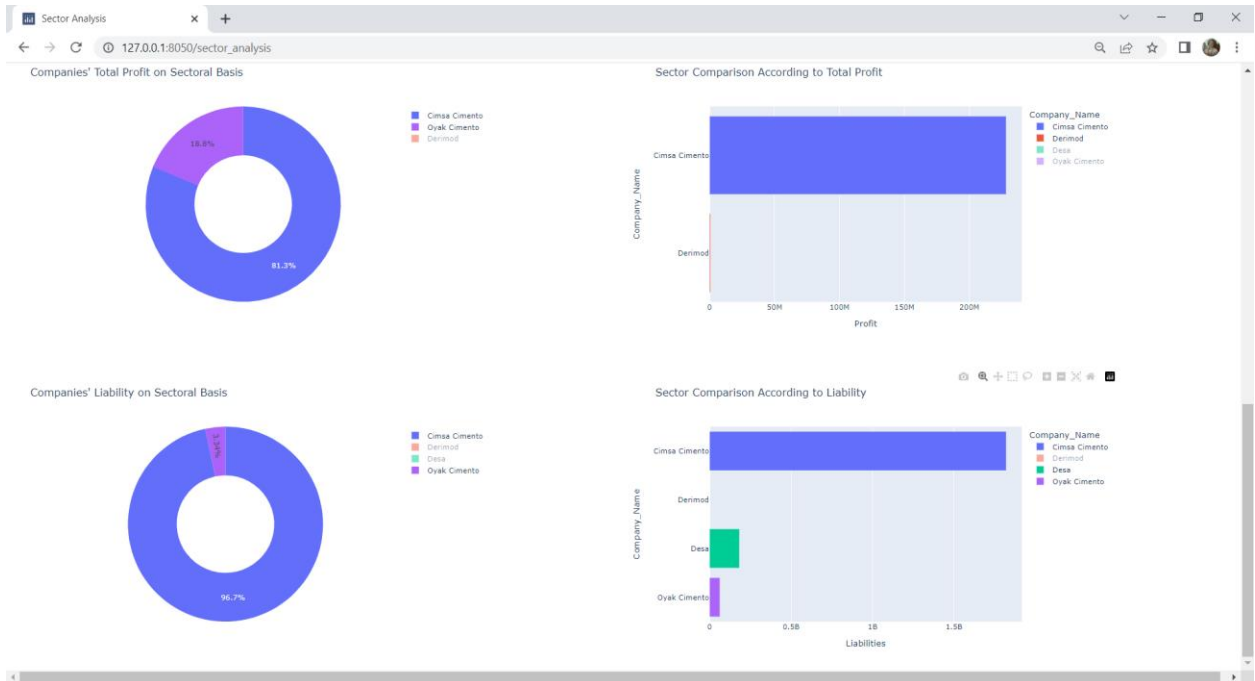




Firms may not always be able to make a profit, they may be in loss periods. Loss periods also could shown the bar charts. For example; Derimod, operating in the industrial sector, made a loss according to 2020 year-end financial data.



If the user do not want to see the all companies, thanks to dash graph component, when the user click the company name, that company clicked company remove the pie chart or bar chart.



2.2. Realistic Constraints

We have no available budget. There are products that serve financial analysis but most of them for investors not financial analysts. Big companies employ the professional, small companies paid fee applications or take services from companies like JCR Eurasia Rating. There is no need for budget except our labor.

We think there is no environmental, health, safety and manufacturability constraints.

3. METHODOLOGY

We used Python in our project. In Dashboard design, we got help from Plotly Dash.

3.1 Libraries we use to connect to the database in our project

3.1.1 from mysql.connector

We used MySQL to transfer our data to the database via Python code and to retrieve data from the database. Therefore, we needed a connector to transfer our data over Python.

3.1.2 import pymysql

We needed a connector to connect to the database. In this way, we were able to transfer data to the database and retrieve data from there. For this reason, we used the connector, which is the most common among the alternatives among users.

3.1.3 from sqlalchemy import create_engine, types

We used this library to create a new schema.

3.1.4 import pandas as pd

It was used to read csv files containing financial information of companies, to convert them into a dataframe and to save them in the database as they are.

3.2 Libraries we use to use Plotly Dash to design a dashboard

3.2.1 import dash

Code that allows us to use Plotly Dash.

3.2.2 from dash import html

It allows to use html components.

3.2.3 from dash import dcc

dcc means dash core components. It is a library containing useful components for developers who want to design dashboards with dropdowns, checklists, sliders.

3.2.4 import dash_bootstrap_components

It is a library containing bootstrap components for developers who want to design dashboards.

3.2.5 from dash import callback

It is a library that allows us to use the callback method. Callback functions that are automatically called by Dash whenever an input component's property changes, in order to update some property in another component (the output).

3.2.6 from dash import Output, Input

They are written together inside the callback method. The Input method is where we specify the name and type of the components from which the data is received. Output, on the other hand, allows us to specify what type of data will be converted and which component it matches. More than one input and output could be coded into a callback method.

3.2.7 from dash.exceptions import PreventUpdate

The library that prevents us from getting an error when we don't have any options in our dropdowns. It does this by stopping the page refresh.

3.3 The library we used to create the authentication form

3.3.1 import dash_auth

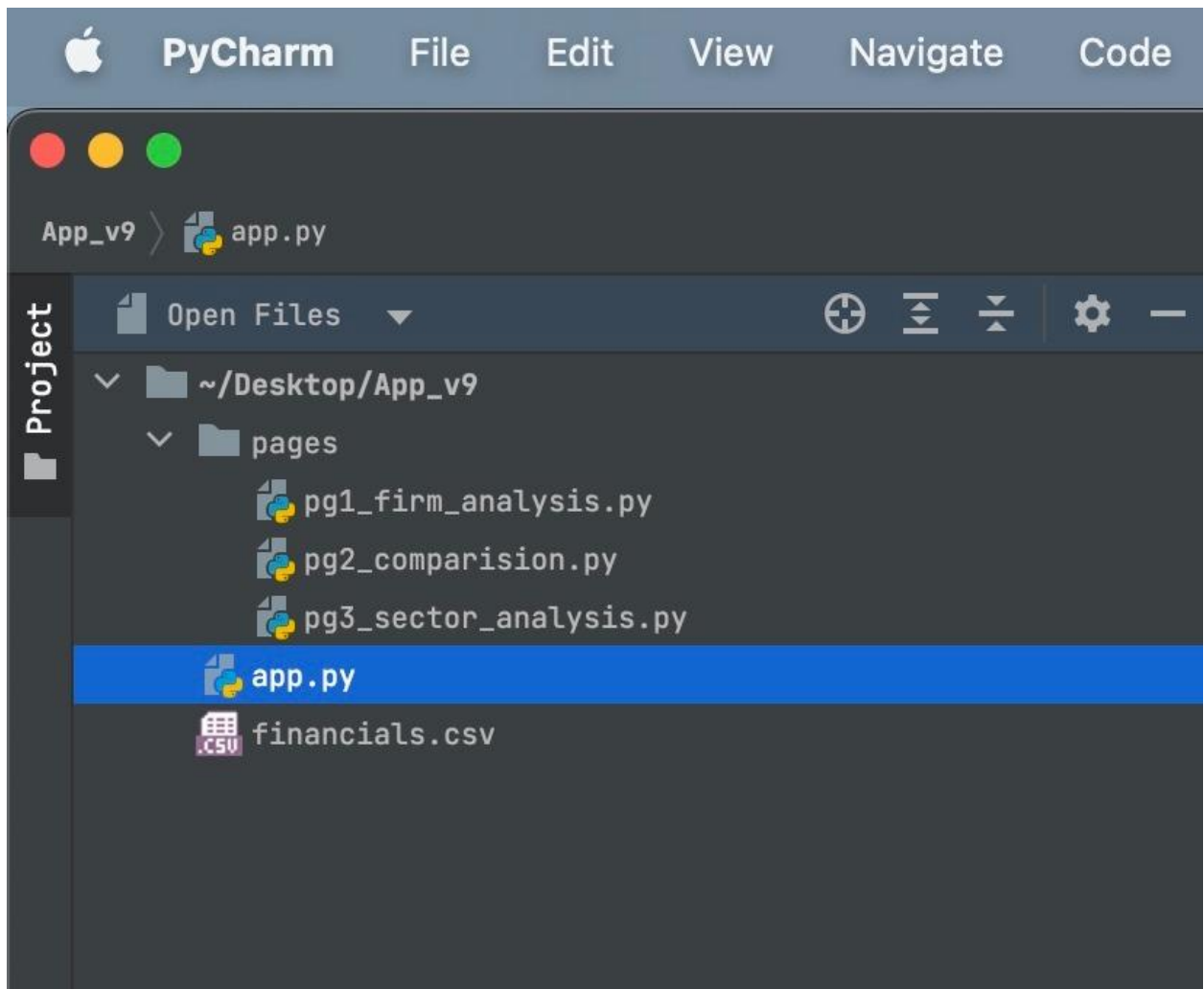
We used it to create an authentication form

3.4 The library we use for alternative graphics

3.3.1 import plotly.express

The library we use to draw bar graph, pie chart and line graph.

3.5 Project Files



3.5.1 financials.csv

Our data file with csv extension, which contains companies and financial information. We will save this data file to MySQL database, then we will retrieve the data from database and design a dashboard for the screen. Although the data of the companies are public, this file was obtained from our workplace. As a precaution, we have limited company and financial information.

3.5.2 app.py

The file from which we run the application. In this file, we provide the database connection. We save the data in financials.csv to the database in this file. This is where

we create our schema and table. We also retrieve our data on this page. Again, on this page, we bring the Authentication form. We design the page structure and provide the creation of alternative pages.

3.5.3 pages (pg1 firm analysis, pg2 comparison, pg3 sector analysis)

Since our project includes multipage design, we have shown the page designs here.

pg1_firm_analysis contains only one company's data and graphics. It has three dropdowns. The first dropdown is the sector selection, the second dropdown is the company selection, and the third dropdown is the year selection. Dropdowns are connected to each other by the callback function. Therefore, when a sector is selected, only the financial information of the companies in the selected sector is reflected on the screen.

In the card information on this page, the company's data for the year selected in the third dropdown is reflected on the screen. The bar charts on this page enable the monitoring of the nominal data of the companies by years.

pg2_comparison includes comparative data and graphics of the companies. On this page, there are four dropdowns side by side in pairs. In this way, a credit rating analyst or any user could easily see the data, compare the financial data of the companies and select different companies in different sectors. She/He could choose the rate she/he wants via radio-buttons. At the bottom of this page are bar graphs. Since bar charts are grouped, it is aimed that companies could compare variables such as profit, liability, shareholders equity or total assets in terms of size.

pg3_sector_analysis includes comparisons of sectors by years. Bar chart and pie chart are used on this page, which contains the data of the sectors by years. The purpose of the bar chart is to see the leading company in the sector more easily and to determine the financial data. The purpose of using pie chart is to easily see which company is dominant in the sector chosen by the user.

In summary, the first page presents the data for only one company, the second page is used for company comparison, and the third page is used for the comparison of sectors by years.

4. RESULTS & DISCUSSION

While designing our project, there were a number of goals we wanted to reach. We defined these goals at our proposal as:

- Our software use database which is we created.
- In our project users could found financial analysis and information about four sector which are industry, banks, factoring and insurance

- In our project users also found companies financial statement analysis from every sector
- Comparative analyzes will be presented with visual graphics

When we come to the end of the project, we could state that; we implemented all of the things we thought about during the design phase of the project.

First, we created the database via Python code. We created a schema called "myschema" in MySQL for the financial data of the companies. At this point, we have written a component where users could add any csv file to the dashboard and save csv files to the database. However, considering that such a situation may contaminate the database, we have concluded that it would be appropriate to remove this component later.

Secondly, we worked on the project by collecting data on four companies from four sectors. The number of data for companies and years could be increased optionally. However, the fact that the company structures are different and the accounting accounts of the companies in different sectors are different from each other has made it difficult to compare the companies in different sectors with each other. Despite this, the user could compare different companies in different sectors or different companies in the same sector by looking at the basic financial parameters and easily use the dashboard we have designed.

Thirdly, users could get an idea about the financial data of the companies on a yearly basis by using the cards in our project. It could be compared with any company from any sector it chooses through graphics. In fact, it could see the dominance of the company in the market in the parameters it chooses in the sector it is in.

Finally, users could compare the company they choose with another company, as well as see the status of the selected company in the sector. At this point, pie charts, bar charts and line charts are designed to assist users.

However, we also wanted to use the extra components in Plotly Dash in order to improve the user experience and enrich our project.

In this regard, it must be said that our project has developed positively and has become even better than we imagined. Although it is not in our goals, we have designed an authentication form to make the user experience better. This form makes the entered username and password information by establishing a database connection. We created some users in MySQL database and set passwords for them.

Secondly, while designing our project, we imagined our dashboard to be a single page. However, in order to improve the user experience and enrich our project, we made our project multipage and included three-page graphics and reviews.

In line with the points, we have emphasized above, we could proudly state that the project has been successfully completed.

5. IMPACT

Our main purpose when designing our project was to facilitate the work of credit rating analysts and users who want to invest but do not have sufficient financial literacy, and to present them with the financial data they want to see in a very short time. In addition, if they wished, they could compare these financial data with the financial data of other companies. In this way, users would not have to go through the trouble of visiting many websites and downloading many documents, and they would be able to see the data they wanted in a very short time comparatively. So, at this point we have achieved our goal.

Any credit rating analyst could make a comparison between the financial data of a previously rated firm and the firm she/he is currently working with. If she/he wants, she/he could do this over the years. As a result, it could produce a more accurate rating score and give credit rating scores of large firms more fairly.

On the other hand, an investor could see the financial data of companies in a healthy way while making an investment decision. She/He could track them over the years and make a prediction of the future. In addition to these, it could find the best company to invest in by comparing the companies it wants to invest in. He could even see the total assets, profits, shareholders equity and liability in the sector of the company she/he wants to invest in. In addition, the investor could determine to what extent the company to which she/he will invest dominates the market in the parameters mentioned above.

We think we have designed an app that is readable and easy for almost anyone with a basic level of financial literacy.

In addition, we think that the hard work and effort we spent before the delivery of the project has reached a successful result. As a matter of fact, we think that if the data on which our project feeds are enriched sufficiently, it could be used for commercial purposes easily and will be appreciated.

There are no Freedom-to-Use issues.

6. ETHICAL ISSUES

There are no ethical issues.

7. PROJECT MANAGEMENT

When designing our project, we wanted it to be used for commercial purposes and to help people. After we decided to type our code in Python, we did some research on how to code a web application. In this direction, we often researched the Django web framework. However, we realized that it would take a lot of time to design the way we wanted in Django, and we might have problems with some of the components we wanted to put. Since Django has little similarity with Python, we decided not to write our project with Django. As we mentioned before, we turned to alternatives such as Plotly Dash.

We knew how to set up basic algorithms and commands in Python. The richness of educational content on Python encouraged us.

On the other hand, we thought that learning a new language and new syntax would be productive, and in line with the suggestions of our supervisors, we started to review Plotly Dash's document. It should be noted at this point that Plotly Dash is a fairly new library, so we had our basic drawbacks here. First of all, there could be a consistency problem in the components in the library, the version updates of the components are not made, some components do not work in the current Plotly Dash versions, Plotly Dash does not yet have a large and skilled developer community that we could consult when we get errors while typing code, the Plotly Dash documentation is insufficient, for this reason, we could not put the components we wanted by only examining the official documentation. We constantly received support from other sources. (Udemy, YouTube etc.)

However, despite these negative sides, Plotly Dash's is a Python library and not having a very different syntax than Python, making interactive graphics easier, and thus being able to fully meet the needs of people, brought Plotly Dash to the forefront for us. In addition to these, users could perform many different functions such as zoom in, zoom out and save the graphics in Plotly Dash graphics.

We often resorted to the internet and training content at the points where we got errors during the coding phase, at this point we regret to state that the Plotly Dash documentation is not at a level that could meet our needs. But they have a small and helpful community.

While coding, we continued to watch training videos and benefit from Plotly Dash documentation. In this way, we thought that it would be a good decision to include the components that we did not plan to include in our project at the beginning and that we did not plan. Hence, we realized that we could improve the user experience and enriched our project. In this direction, we made an authentication form, multi-page design. When we realized that we got an error when we left the dropdowns blank, we stopped the pages from updating and prevented the error from being received.

Another difficulty we experienced while coding our project was the issue of which graphics the company data would be displayed. When we realized that some of the graphics, we projected on the screen were not as we imagined, we had to delete the graphics we projected on the screen with difficulty. For example, we noticed that there is a gap between the total assets of banks and the total assets of some companies in the industry, but we saw that this difference was not well understood when we showed it with a line chart. That's why we decided to put up a bar graph.

At the beginning of the project, just as if we had signed a contract, we had basically decided what we would put in and what data we would use. However, we said that if and only if we have some time, we will review the Plotly Dash documentation and put components that will improve the user experience. For this reason, we did not encounter any extra difficulties in project management other than the coding phase.

The most instructive parts of project management for us were solving unforeseen problems, determining the scope of the project clearly and not going out of it. As a matter of fact, we solved them through dialogue.

8. CONCLUSION AND FUTURE WORK

First of all, we would like to proudly state that we have successfully completed our project. As a result of our long efforts, we have developed a useful application for credit rating analysts and investors trying to make investment decisions.

During the development of our project, we gained valuable information not only about coding, but also about project management. We designed together what will happen in our project at the very beginning. However, there were moments when we were undecided about which components to use in the later parts of the project.

We obtained the financial data of the companies included in our project from the institutions we work with. For this reason, we thought it would be prudent to keep the number of firms and financial data limited.

In the future, we plan to make our project a full real-life application by increasing the number of companies and the diversity of financial data. Currently, our application could be used for people with basic financial literacy and credit rating analysts who want to compare data instantly. However, in the future, we want to make our application suitable for professional users as well.

At this stage, we may decide to add the components that we have given up on over time, as we do not want users to be involved in the database. For example, we could design pages that only users could use, import and analyze csv files with the drag-and-drop command.

If there is any type of chart or data they want to see by getting feedback from professional users, we could show them in our application.

Finally, if we consider that the Plotly Dash library is currently being developed, we could consider adding other components that will be useful to us in the future.

9. APPENDIX

The code files of our project have been uploaded to our school system SUCourse. In addition, there is no other document that we want to upload.

10. REFERENCES

Charming Data – YouTube Channel

<https://www.youtube.com/c/charmingdata>

Stack Overflow

<https://stackoverflow.com/>

Plotly Dash Documentation

<https://dash.plotly.com/>

<https://plotly.com/python/plotly-express/>

<https://plotly.com/python/creating-and-updating-figures/>

Dash Bootstrap Components

<https://dash-bootstrap-components.opensource.faculty.ai/>

Bootstrap Documentantation

<https://getbootstrap.com/docs/5.2/getting-started/introduction/>

W3 Schools

<https://www.w3schools.com/>

Udemy

<https://www.udemy.com/course/python-interactive-dashboards-with-plotly-dash/>

<https://www.udemy.com/course/create-interactive-dashboards-in-python-by-plotly-dash/>

Github

<https://github.com/>

<https://github.com/Coding-with-Adam>

<https://github.com/plotly/dash>

<https://github.com/PyMySQL/PyMySQL>

Towards Data Science

<https://towardsdatascience.com/dash-for-beginners-create-interactive-python-dashboards-338bfc66ffa4>