Tutorial to my Application

Content

Tutorial for the application

Hello and welcome to my tutorial on how to make the most of my Django and Dash application, tightly integrated with our MySQL database. I'll will guide you through this journey, and I'll help you learn how to use this powerful software solution effectively.

My application combines the strengths of Django and Dash to create a versatile and user-friendly environment for data management and visualization. Django, known for its flexibility and scalability, forms the core of our system, ensuring robust data handling and organization. Meanwhile, Dash empowers us to create interactive data visualizations, providing a dynamic interface for exploring and analyzing our data.

In addition to learning about the application, I'll also cover how to interact with our MySQL server, where all our crucial data resides. MySQL, a trusted relational database management system, plays a central role in storing, retrieving, and managing data within our application.

Throughout this tutorial, I'll explain fundamental concepts, walk you through essential functionalities, and provide real-world examples to ensure you can tap into the full potential of our Django and Dash application. We'll also explore how to access and interact with the MySQL server seamlessly.

I invite you to embark on this learning journey with me as we navigate the ins and outs of our technology stack. By the end of this tutorial, you'll have the knowledge and skills needed to make the most of our application and confidently work with our MySQL database.

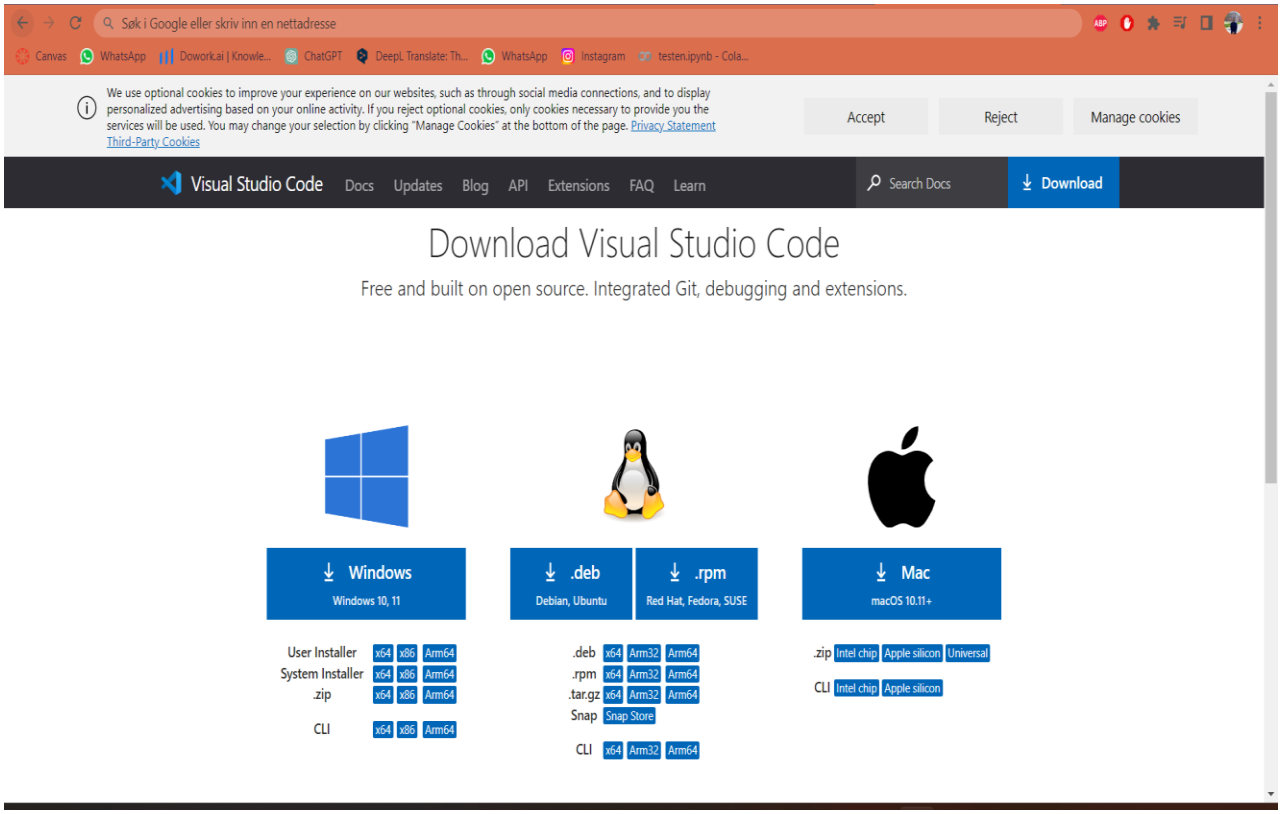
Firstly, I will describe the application shortly. This application is a solution made for Cesar as a more developed version of the project “LucyLattes”. My applications purpose is to take in a zip file containing many zip files, and each zip file contains a Curriculo.xml file. This file contains data regarding a professors/researchers projects, education, productions and much more. The task of my app is to analyze all the data from all the different zip files, put them into dataframes, and later add them all in their

respective database tables. Lastly, this information is presented in a Dash website, where every data is displayed (but not duplicated). So, with this, here is how you use this application.

Before running the application!

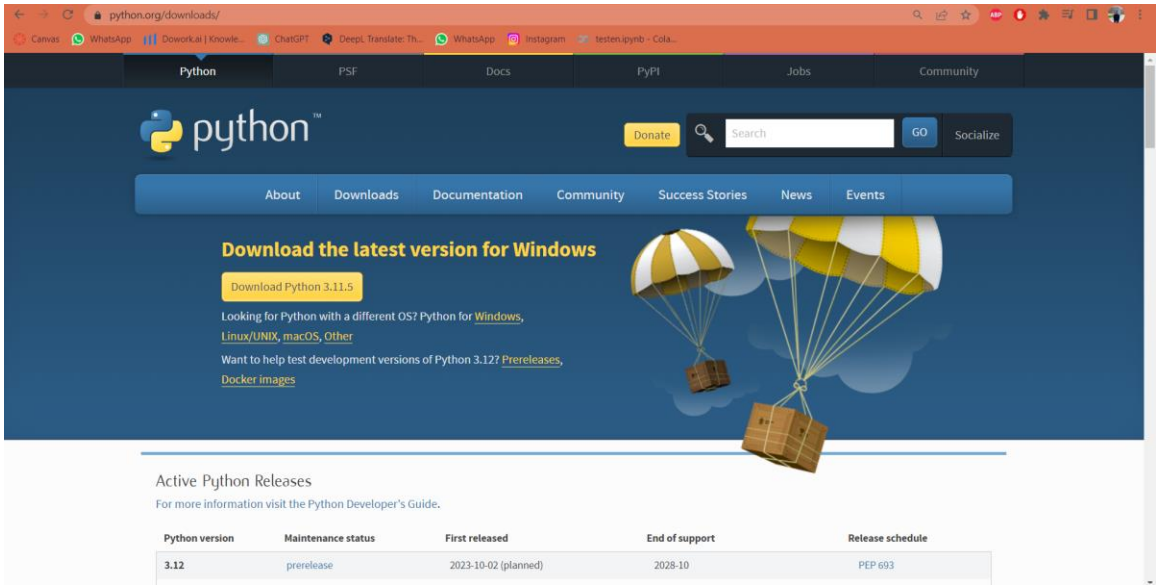
IDE

There are plenty of libraries that you have to install! Since this app uses different kind of technologies and different kind of libraries, its essential that you download it all! First and foremost, you need to download an IDE. My personal recommendation is the Visual Studio Code IDE. This is simple to use with Django, python and many other languages. With this link https://code.visualstudio.com/download, you can download the visual studio code to your own OS.



After this you need to setup the IDE with your own personal settings. Another thing that is essential, is that you download python (to get the pip function). Having pip is important to be able to download libraries like mysql.connector etc. To do this you have to go to the link here

https://www.python.org/downloads/ and download the most recent one (image presented under). After this, you have completed the first step on configuring your apps.



Using PIP

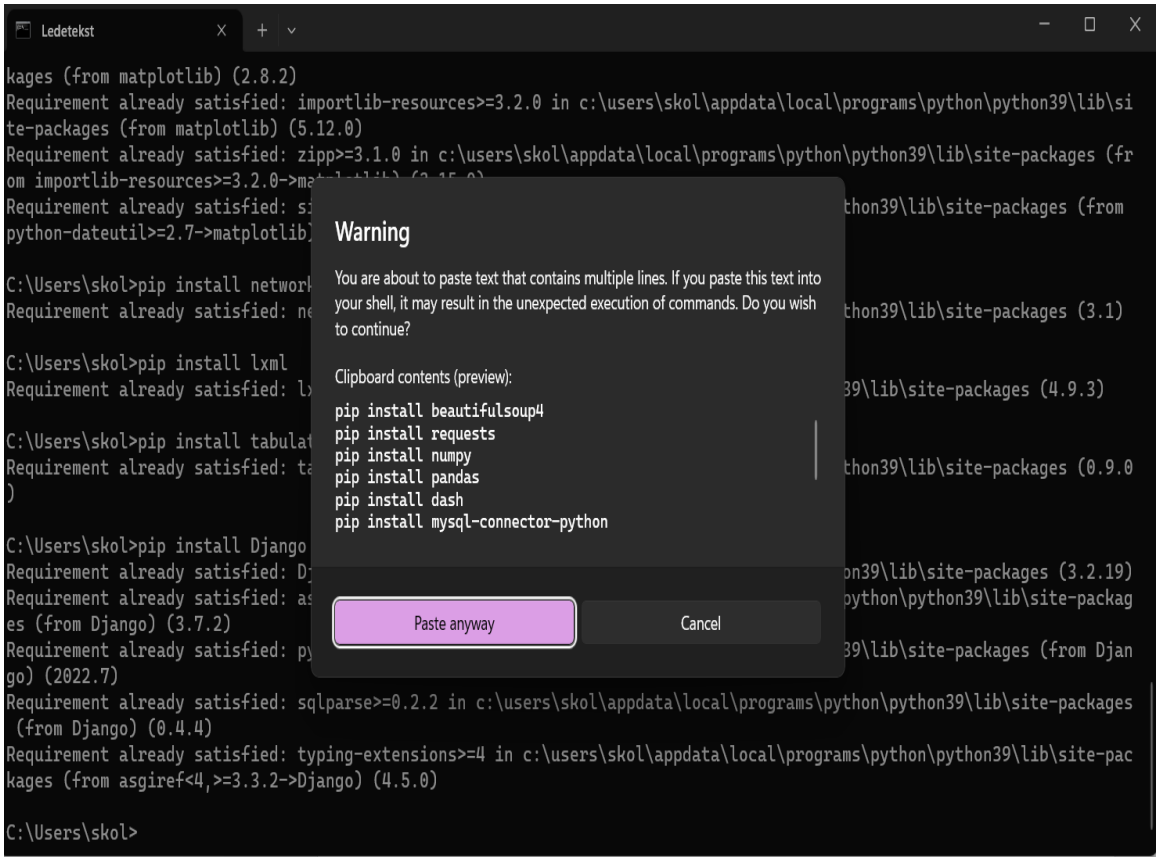
The next step is to actually use Pip to install all the libraries. I have added the code here which you only need to copy and paste into your CMD.

```
pip install beautifulsoup4
pip install requests
pip install numpy
pip install pandas
pip install dash
pip install mysql-connector-python
pip install matplotlib
pip install networkx
pip install lxml
```

pip install tabulate

pip install Django

When you copy it in the terminal (CMD) you will potentially get a message

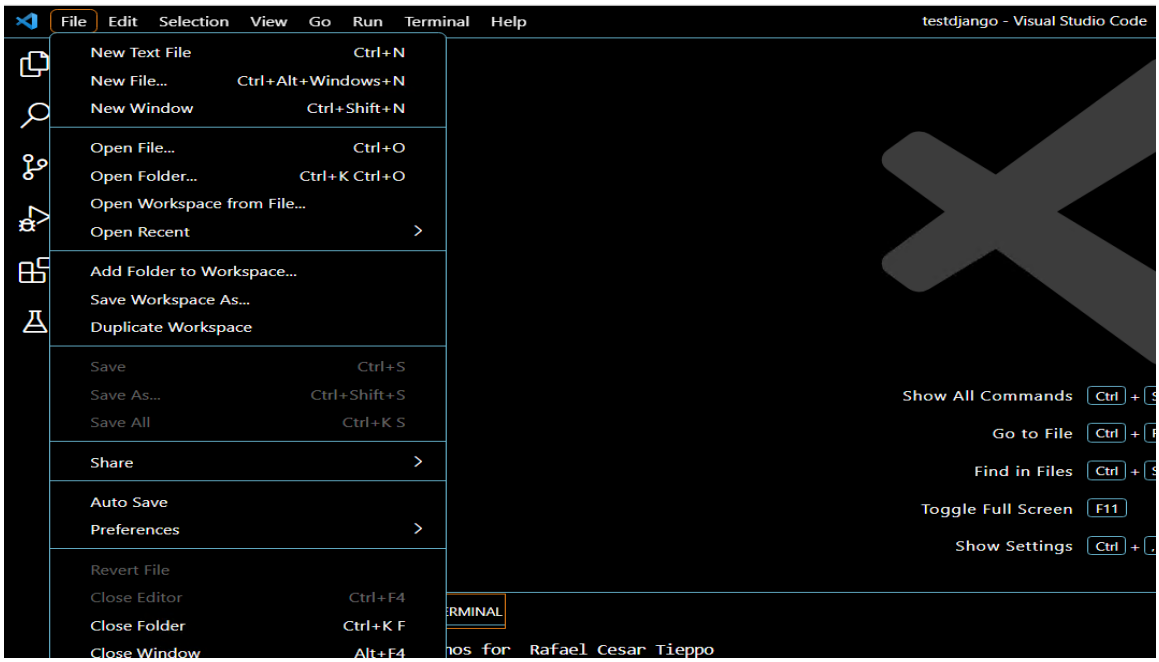


If this comes, just paste anyway and let all the libraries get installed. Once this is over, there might be some imports that gets some errors. If this happens, either install them or alter them to a way which works.

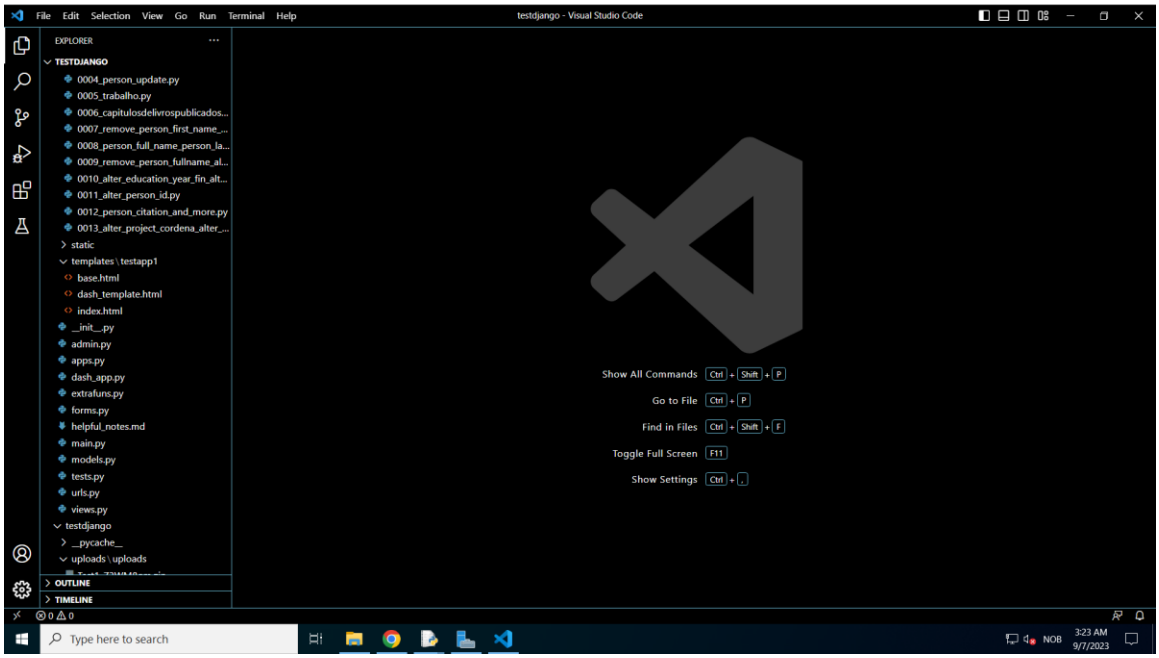
After this you have finally installed everything that needs to be installed. After this, you can begin with the application

Starting out with the Application

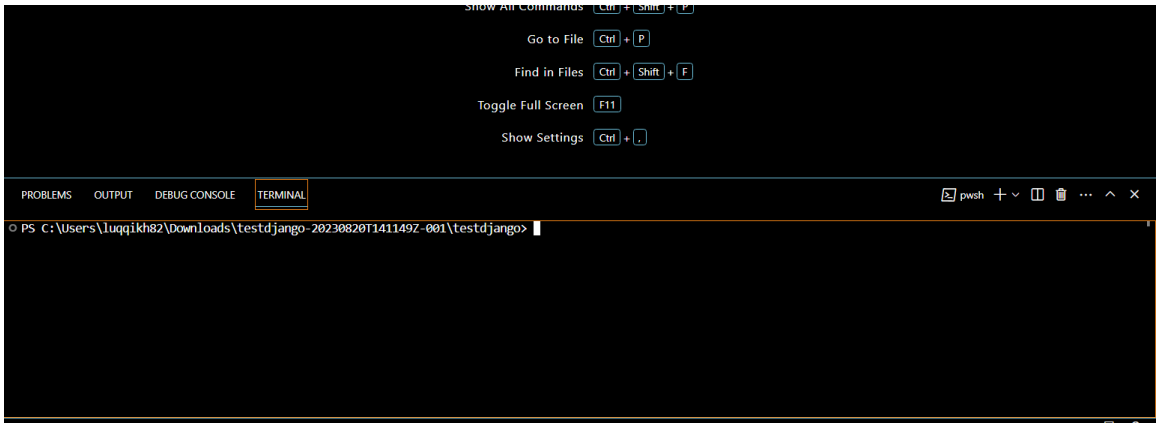
The first thing you have to do is to open the Django project in your IDE. I will do this tutorial on Visual Studio Code, so I recommend you to use the same IDE. The first thing you have to do, is to open the zip file in your visual studio code like this



You have to go to the alternative “Open folder” where you will open the Django project that I have uploaded. This is how it should look like after you open the zip file.



All the files on the left should be there. If its there, then you have done it correctly. After this, you need to open a terminal. The way to do that is by going to the top bar, pressing terminal and opening a new terminal. This is what you will get up after



Running the server and the project

To run the server is a pretty simple task. You have to type the following in the terminal:

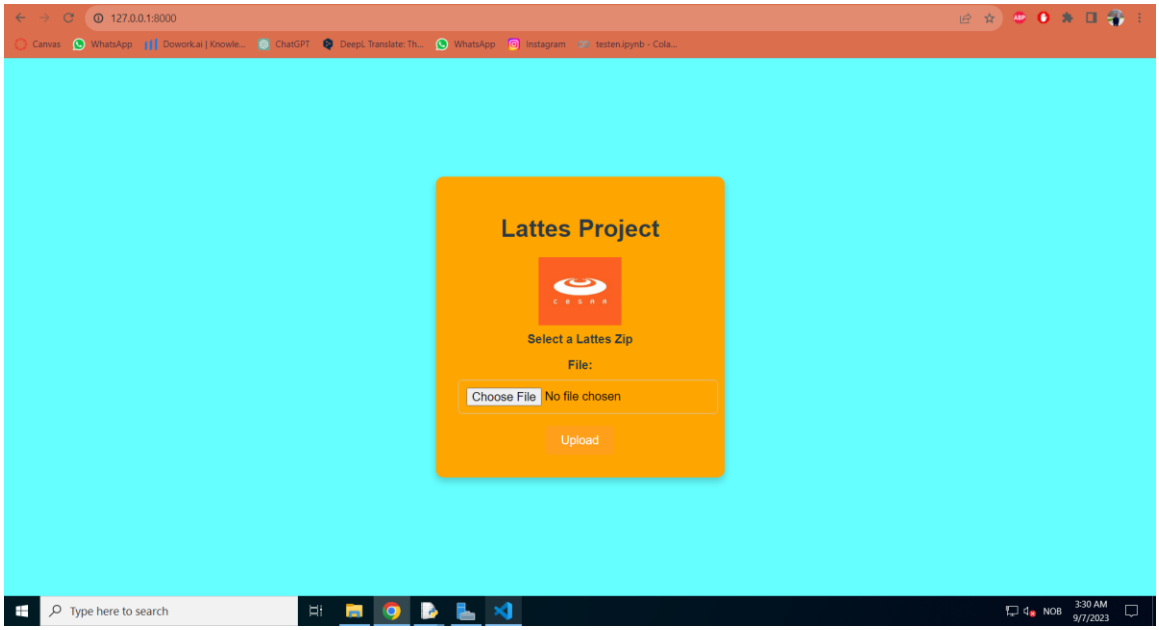
Py manage.py runserver

Sometimes this doesn't work, and you have to write this instead

Python manage.py runserver

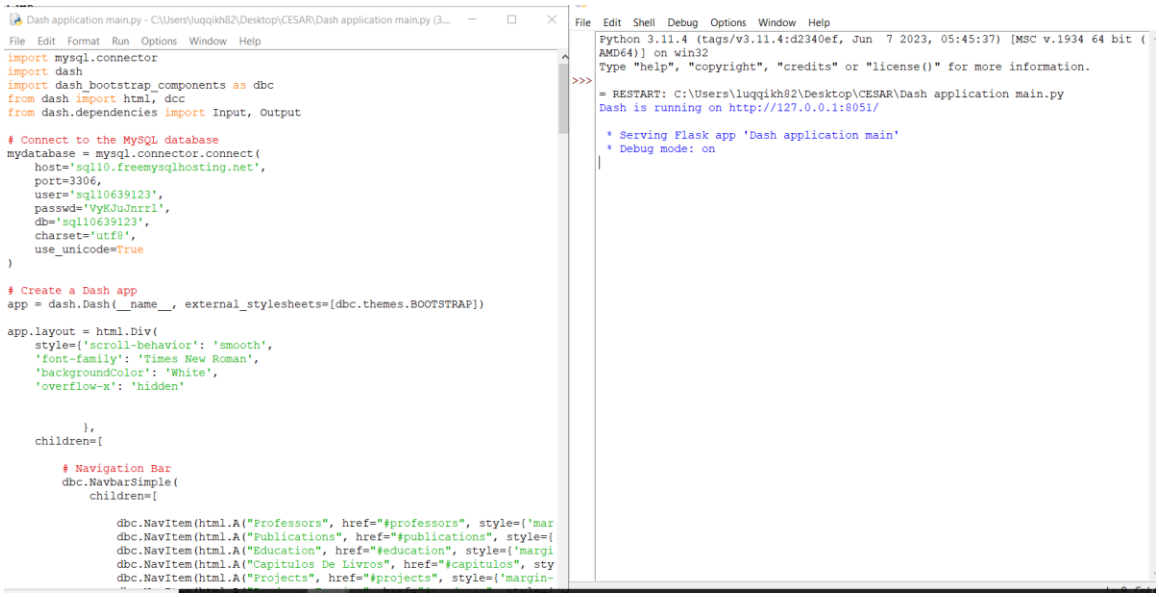
After this you will get this message

This is saying that you have now successfully ran the server and you have an url to see the app. Go to the link that is provided when you do this, and you will arrive here



This is the website of uploading the zip file. The last thing you need to do before uploading, is to run the dash file that I have uploaded in a separate folder. I personally run it in separate IDE, but this is optional. This is how the output will be:

Dash



As you see, you have the dash application running on another link. After the Django project is finished, it will automatically go to the dash website to display your datas.

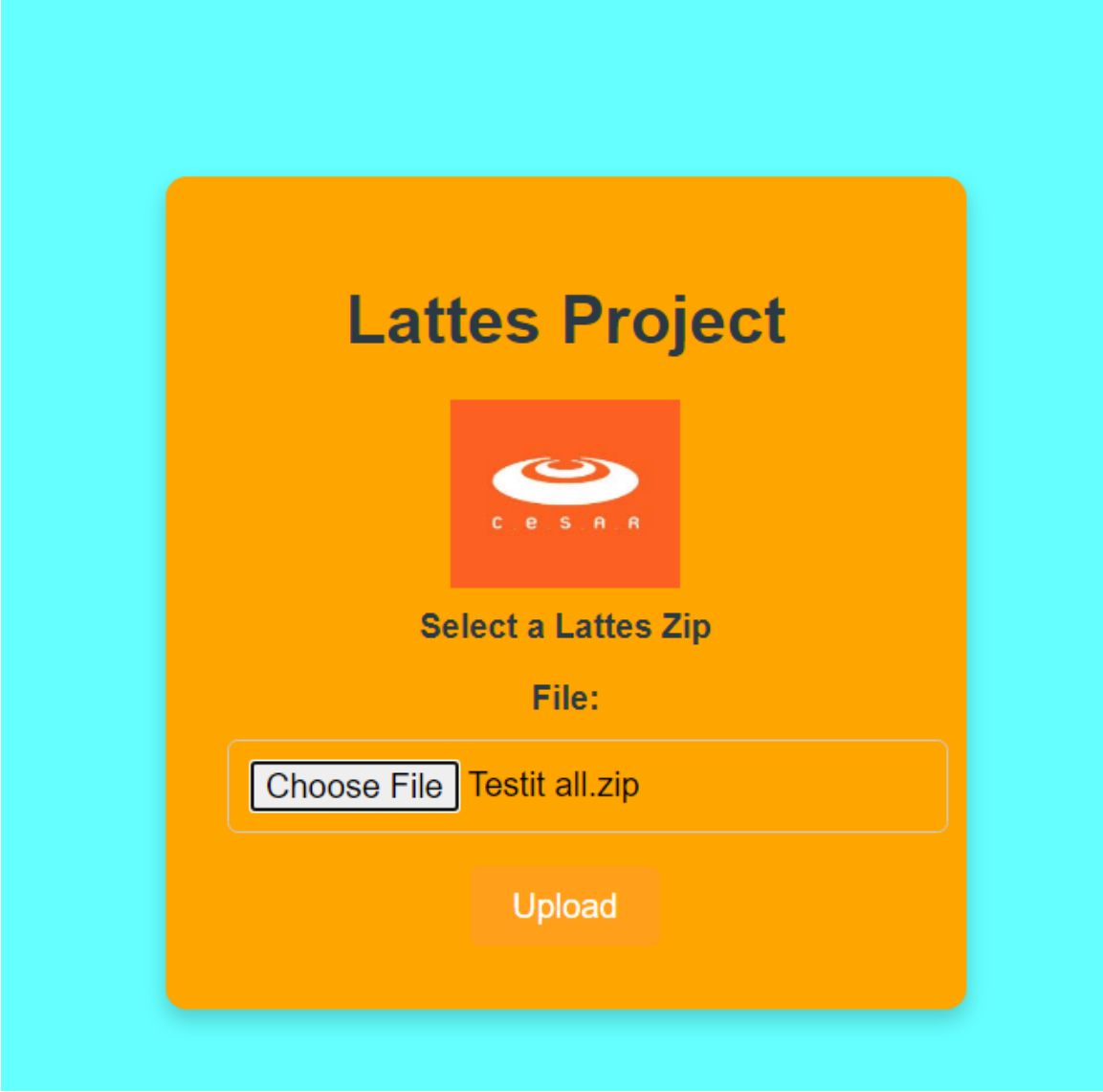
When that is finished, you can start to run the project. But first, you need to make sure that you have the format correct. The format should be that it is a big zip file, containing a folder that is containing many zip files, that each have a curriculo in them. This is an important detail to remember, for using the app optimally. This is how your skeleton should look like:

Skeleton

```
- mainzip.zip
  - mainzip (folder)
    - subfile1.zip
      - Curriculo.xml
    - subfile2.zip
      - Curriculo.xml
    - subfile3.zip
      - Curriculo.xml
```

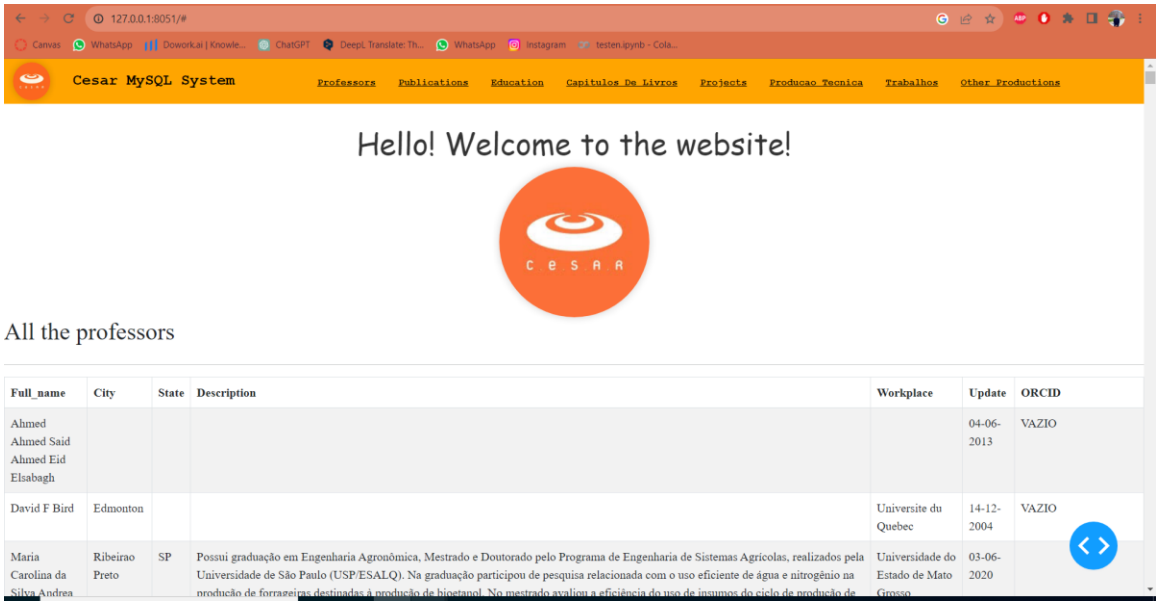
Pretty much, this example has a zip file containing 3 xml documents in 3 zip files that is zipped. You can have more or less, as the minimum of documents has to be one and you can have as many as you want over 1.

This is how the project is looking like when you upload a zip file

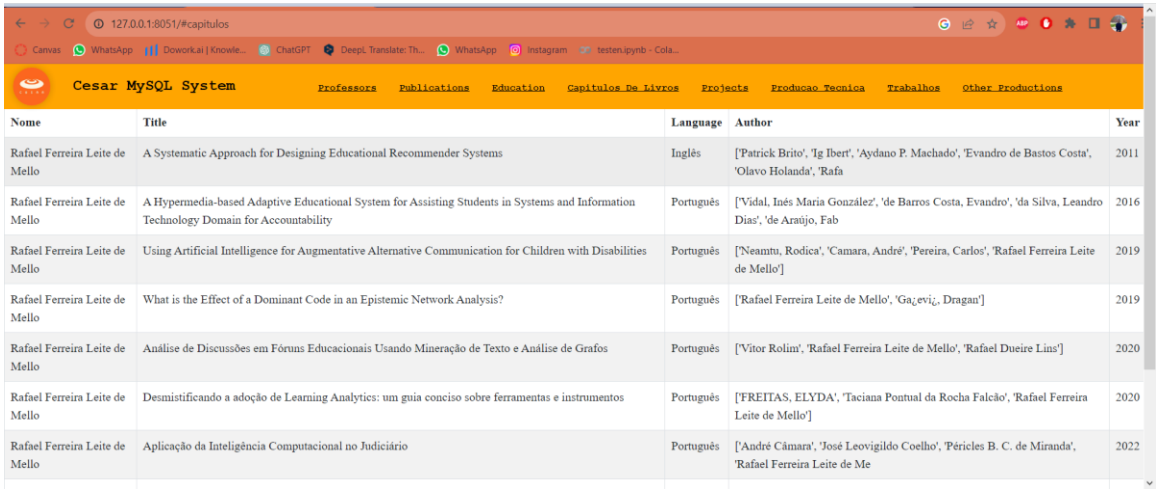


This zip file “testit all.zip” contains 5 zip files that have 5 curriculos. When you are ready, you just press upload and wait. If you are curious to see what is happening, open your terminal in visual studio code. I have made sure that you as a user is always updated on what is happening in each second of the code.

Lastly, when its finished it will automatically direct you to the dash website, that contains all the data. Here is how the end website will look like



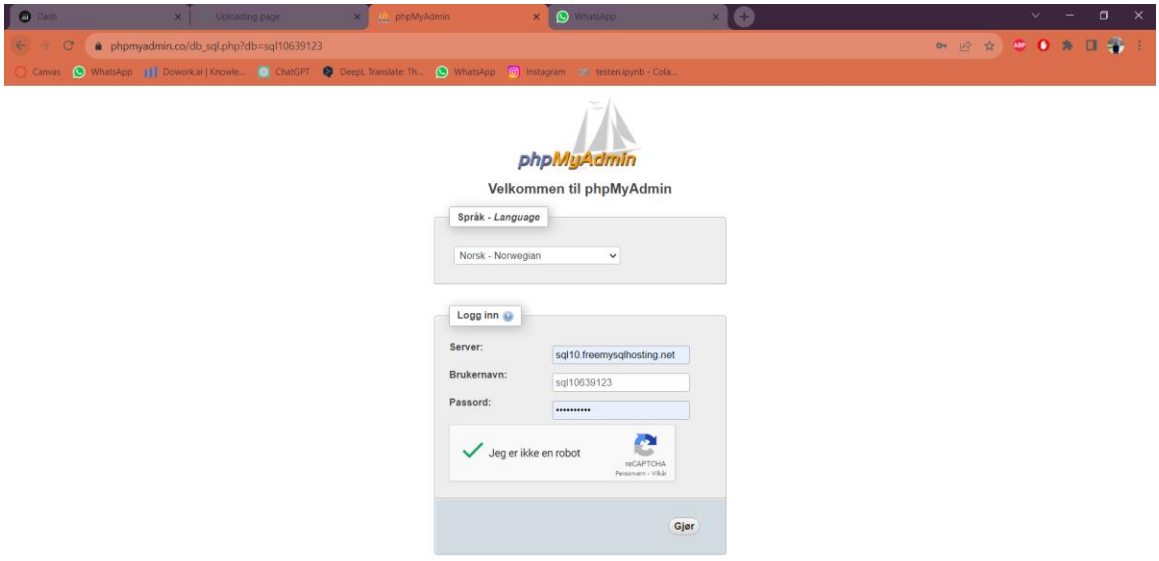
Here you have all the professors and all the different fields that are mentioned in the top bar. All of those menu items are different kind of tables. If you click on them, they will direct you to wherever the data of that menuitem is. For example if you click on capitulus you will come to the capitulus part as shown here



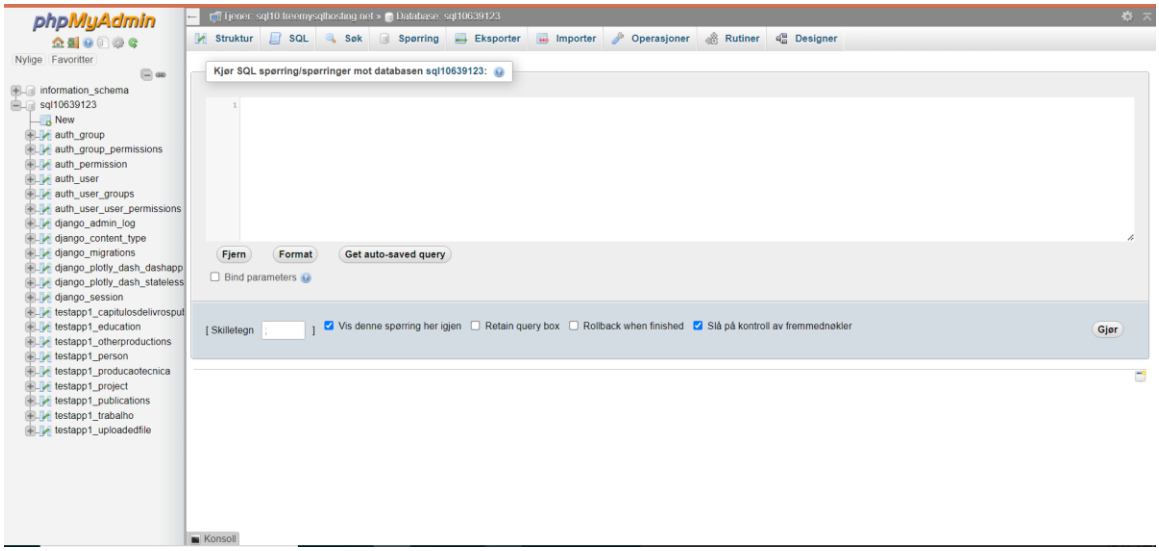
This is all the data of all the professors and its ordered in year and grouped by each professor.

Database

If you want to access a database, there are many methods. But since the free database I currently use require this, I need to access it like this. There is a software called PHPmyadmin which lets you administrate your database in different kind of methods



This is how it looks like. If you log in you will get this screen



You have all your tables, and you can send queries to the database here and commit them. This is a great way to have control of your database. Good luck with using the application!