Super Wine Recommender

Barbara Kharkheli

Goals of the project

Provide insights into wine export, import, production and consumption patterns

Part One

Develop a ML model that suggests wines based on meal images

Part Two

Project Steps

01

Data Gathering from Different Sources

02

Data Wrangling/Cleaning

03

Database Creation and ERD

04

API Creation and Documentation

05

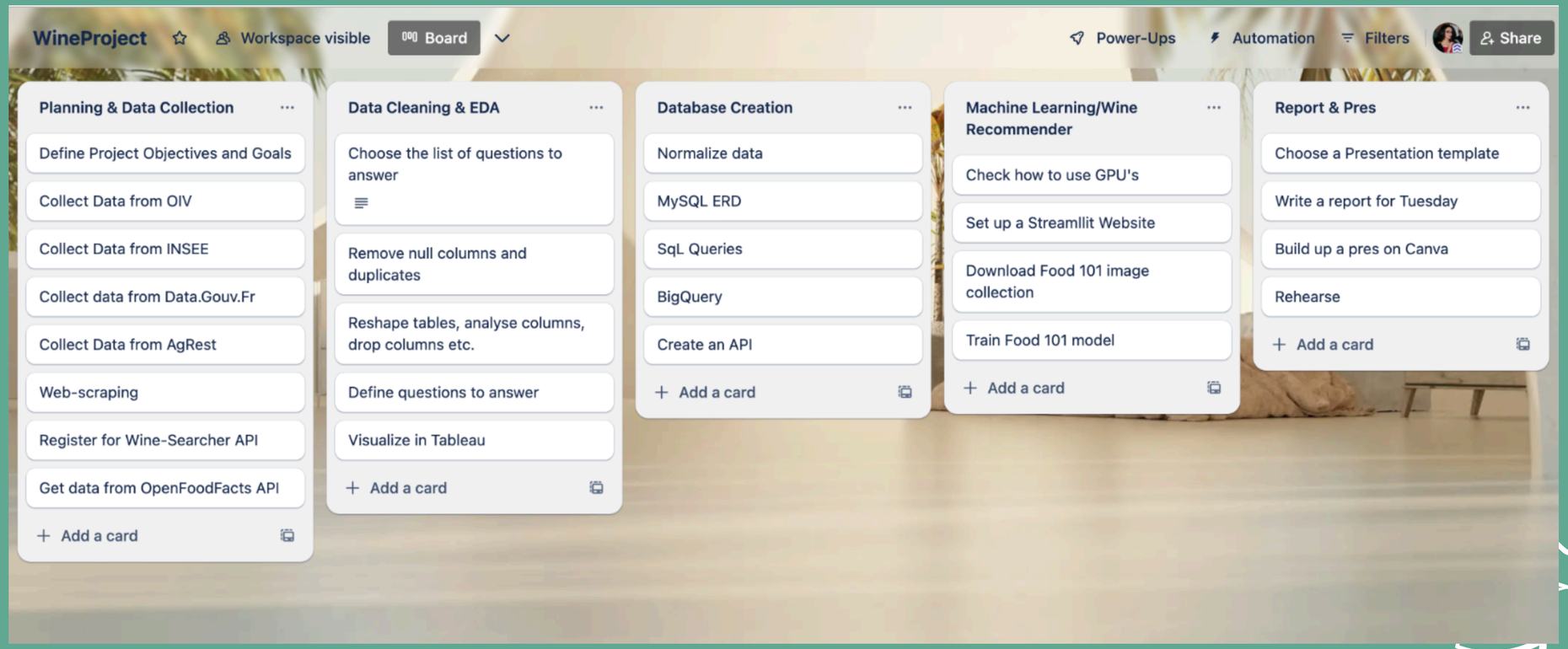
EDA and Visualisations

06

Machine Learning and Streamlit App

Project Management

Trello Board



01

Data Gathering

To achieve the project goals, I will collect data from multiple sources, ensuring a comprehensive and robust dataset for analysis and model training

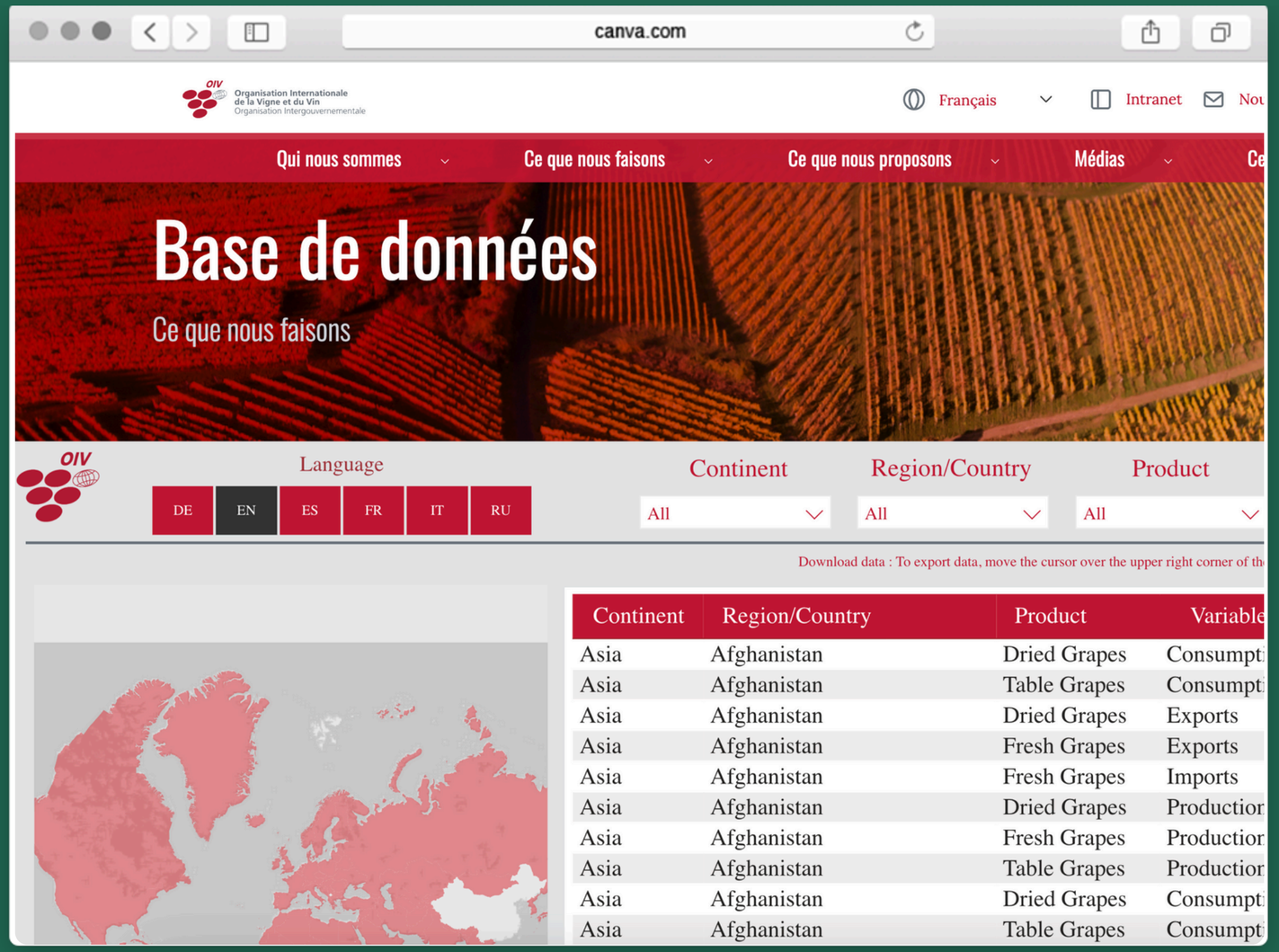
Flat File Sources:

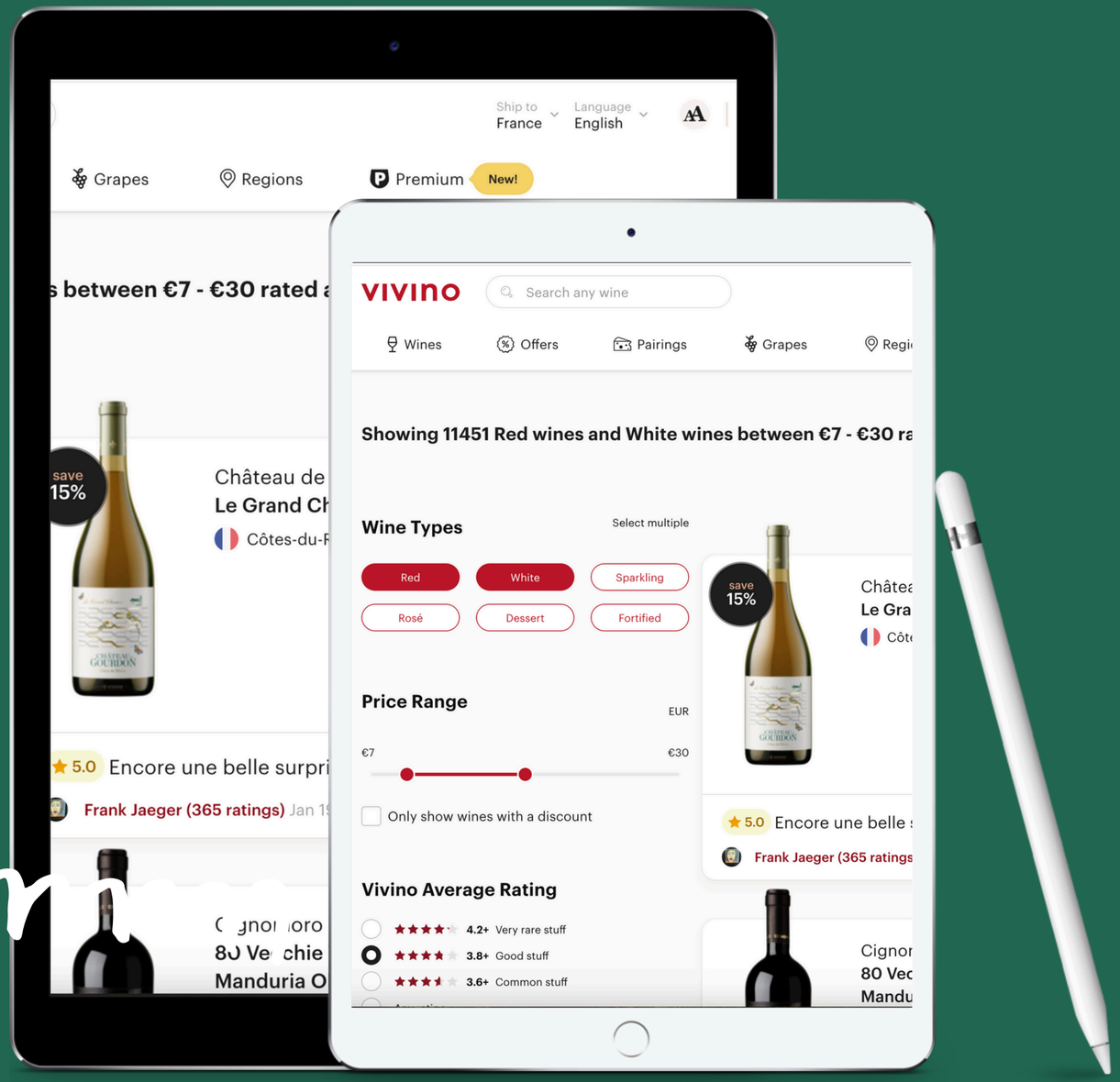
OIV - Organisation Internationale de la Vigne et du Vin

Data.Gouv.Fr

AgReste (Ministry of Agriculture)

INSEE





Vivino.com - Webscraping

Top 100 wines worldwide

Retrieve Name, Producer, Country, Price

Data from Open Food Facts API

Fetch up to 5000 wine records in France

Create a dictionary with basic information (Name, Category, Ecoscore, Country)

canva.com

world.openfoodfacts.net/cgi/search.pl?action=process&json=1&fields=product_name,brands,categories&page_size=100&p

Impression élégante


```
{
  "count": 7286,
  "page": 1,
  "page_count": 100,
  "page_size": 100,
  "products": [
    {
      "brands": "Saguaro",
      "categories": "Boissons, Boissons alcoolisées, Vins, Boissons gazeuses, Eaux, Eaux de sou
gazeuses, Vins pétillant, en:Boissons  Boissons gazeuses  Eaux  Eaux de sources  Eaux minérales
      "product_name": "Finement pétillante"
    },
    {
      "brands": "Ovomaltine",
      "categories": "Boissons, Boissons alcoolisées, Snacks, Snacks sucrés, Vins, Cacao et dér
chocolatées, Alsace Grand Cru, Alsace Grand Cru Saering",
      "product_name": "Ovomaltine - Vitamin & Mineral Bars, 5 ct 100g (3.5oz)"
    },
    {
      "brands": "Danone",
      "categories": "Boissons, Boissons alcoolisées, Vins, Eaux, Vins français, Eaux de source
      "product_name": "Ain Saiss Eau Minerale Naturelle"
    },
    {
      "brands": "Saveurs de nos regions",
      "categories": "Plant-based foods and beverages, Beverages, Plant-based foods, Alcoholic
Fruits based foods, Plant-based spreads, Sweet spreads, Fruit and vegetable preserves, Wines
      "product_name": "Fruchtaufsteich Klementine-Orange"
    },
    {
```

02

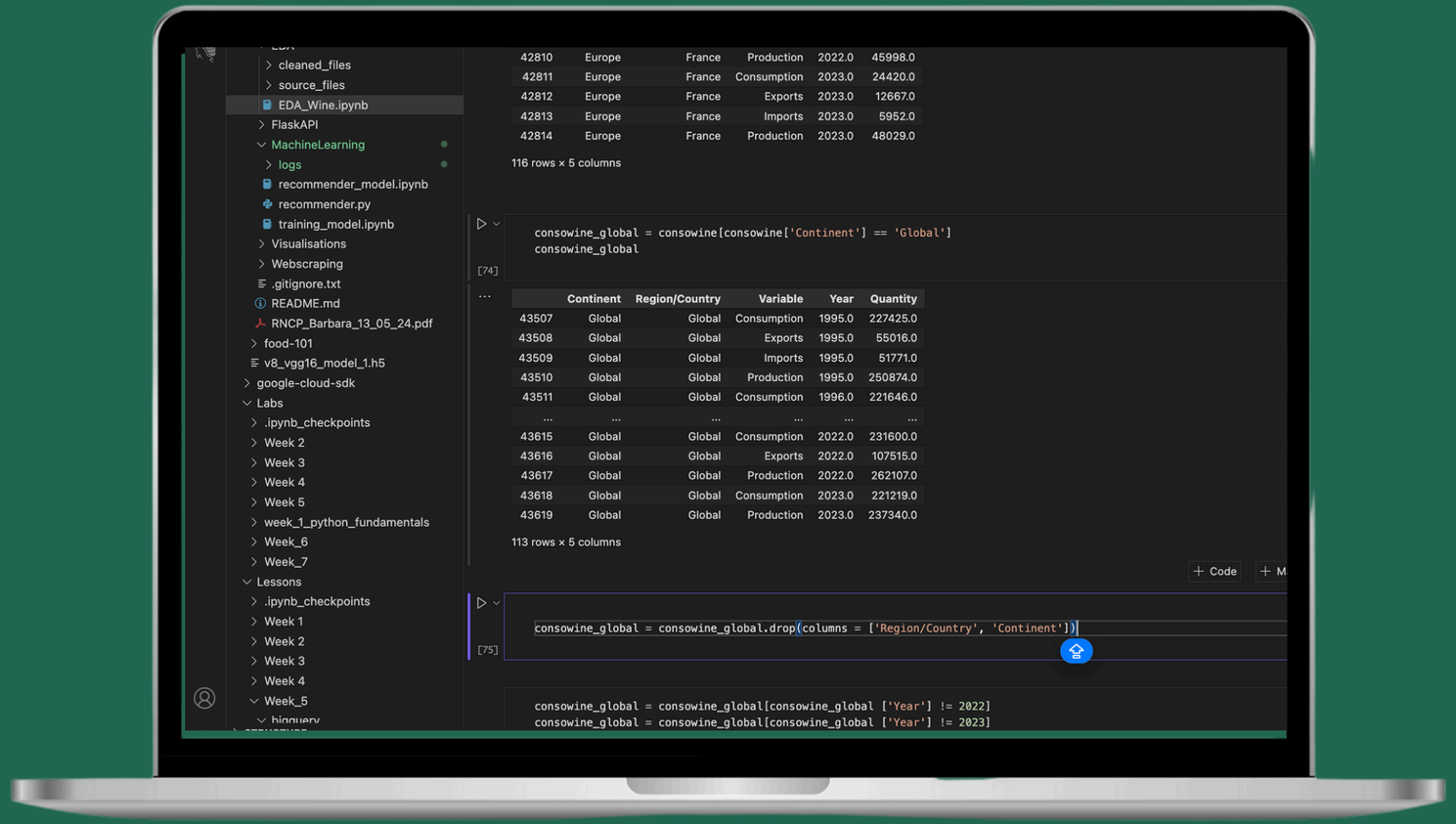
Data Cleaning

The most time-consuming and crucial part was managing data quality. This included handling null values and duplicates, adjusting data types, removing unnecessary columns, reshaping tables, and ultimately saving the cleaned tables to CSV files.

Exemple of cleaning and wrangling OIV table

Data Source: OIV - Organisation Internationale de la Vigne et

- Display the shape of the DataFrame.
- Display the number of missing values, duplicate rows, and data types of each column.
- Remove unnecessary columns
- Display the count of unique values in each column to check for abnormalities
- Exclude the years 2022 and 2023 from rows (due to lack of information)



Python Code

03

Database Creation and ERD

The ‘wine_schema’ database was created in MySQL Workbench to store 11 tables related to the wine market analysis which I collected using flat files, API and web scraping

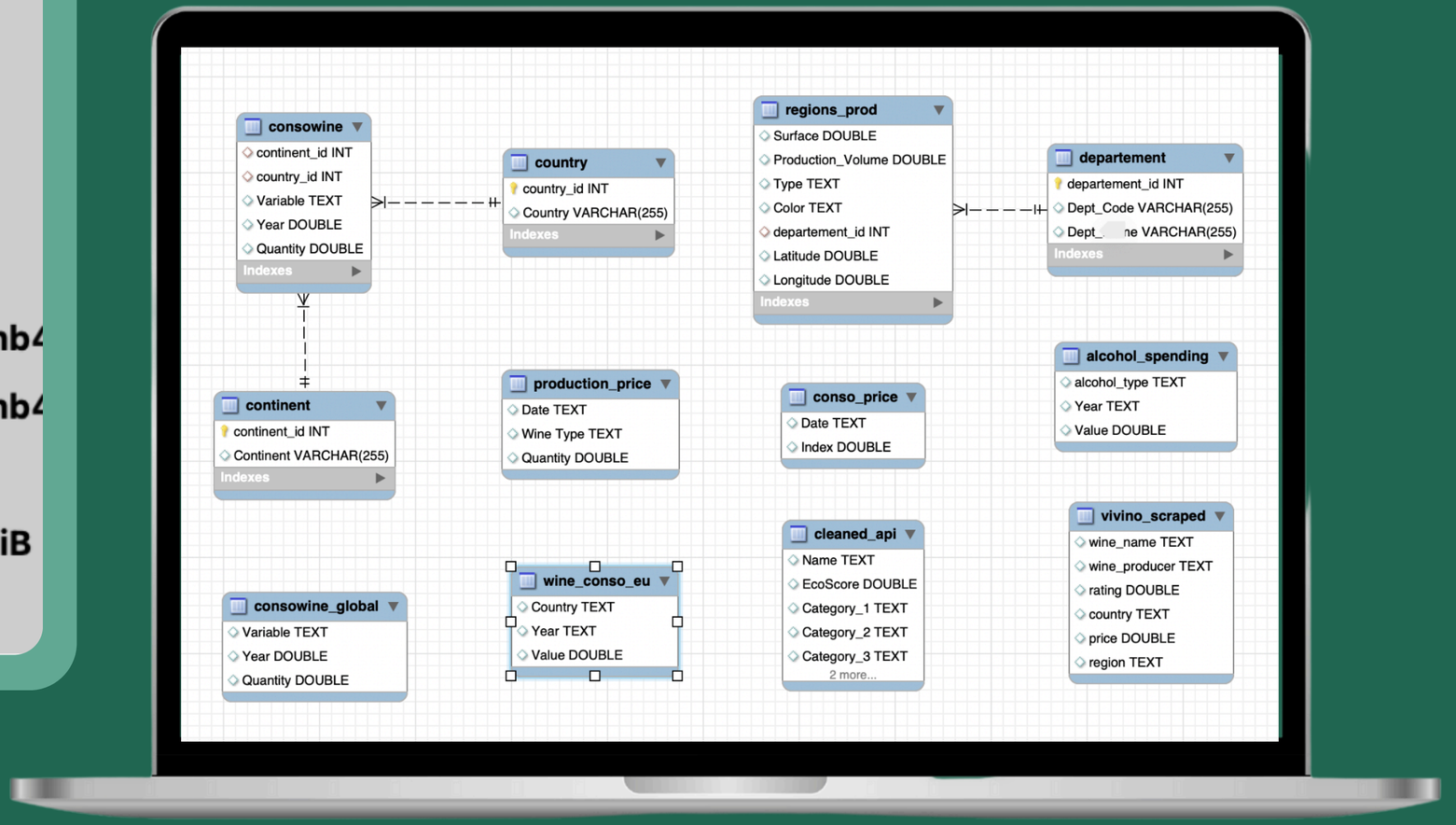
Wine Schema

Ironhack

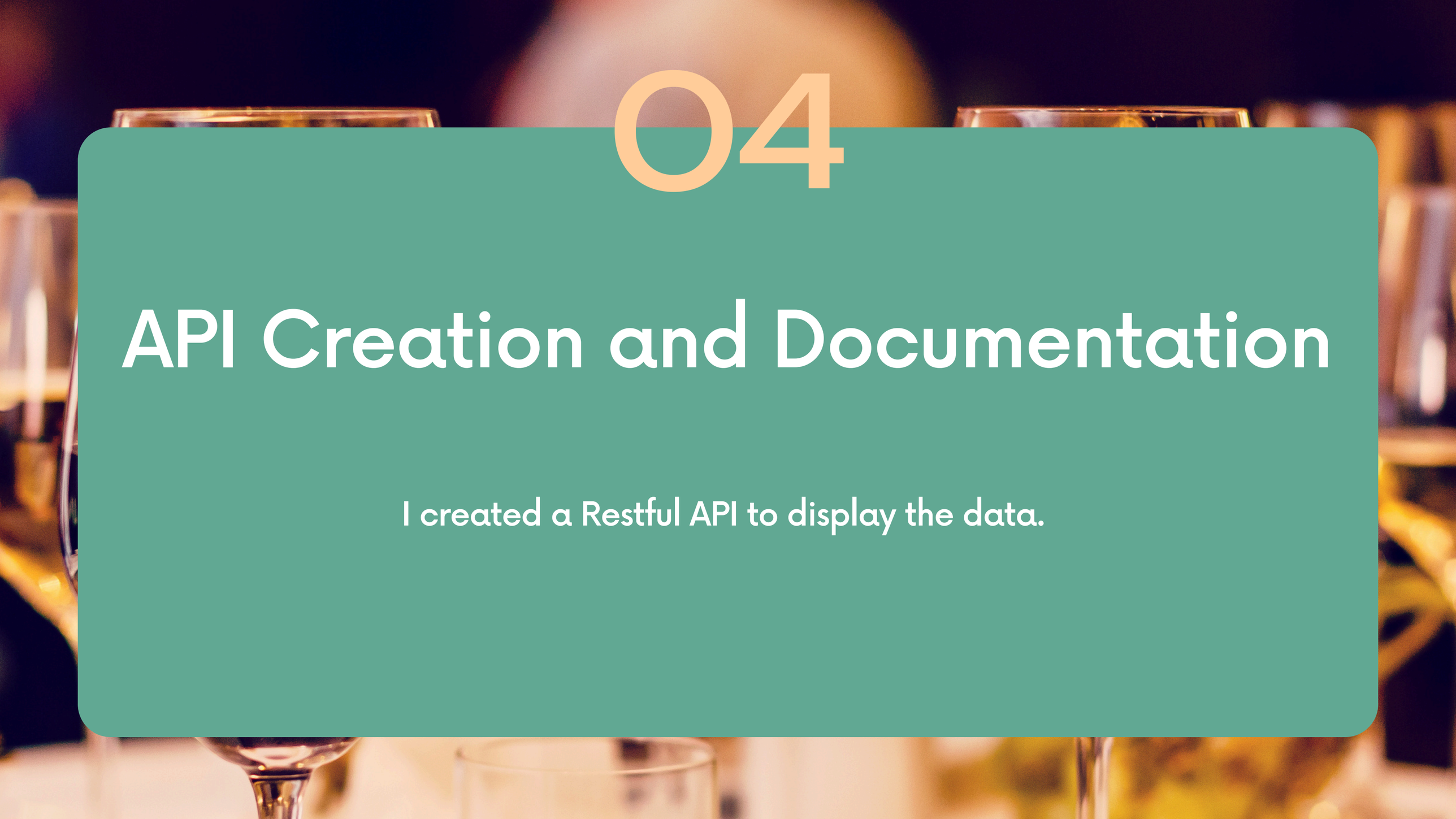
wine_schema

Schema Details

Default collation:	utf8mb4
Default characterset:	utf8mb4
Table count:	11
Database size (rough estimate):	3.8 MiB



MySQL ERD



04

API Creation and Documentation

I created a Restful API to display the data.

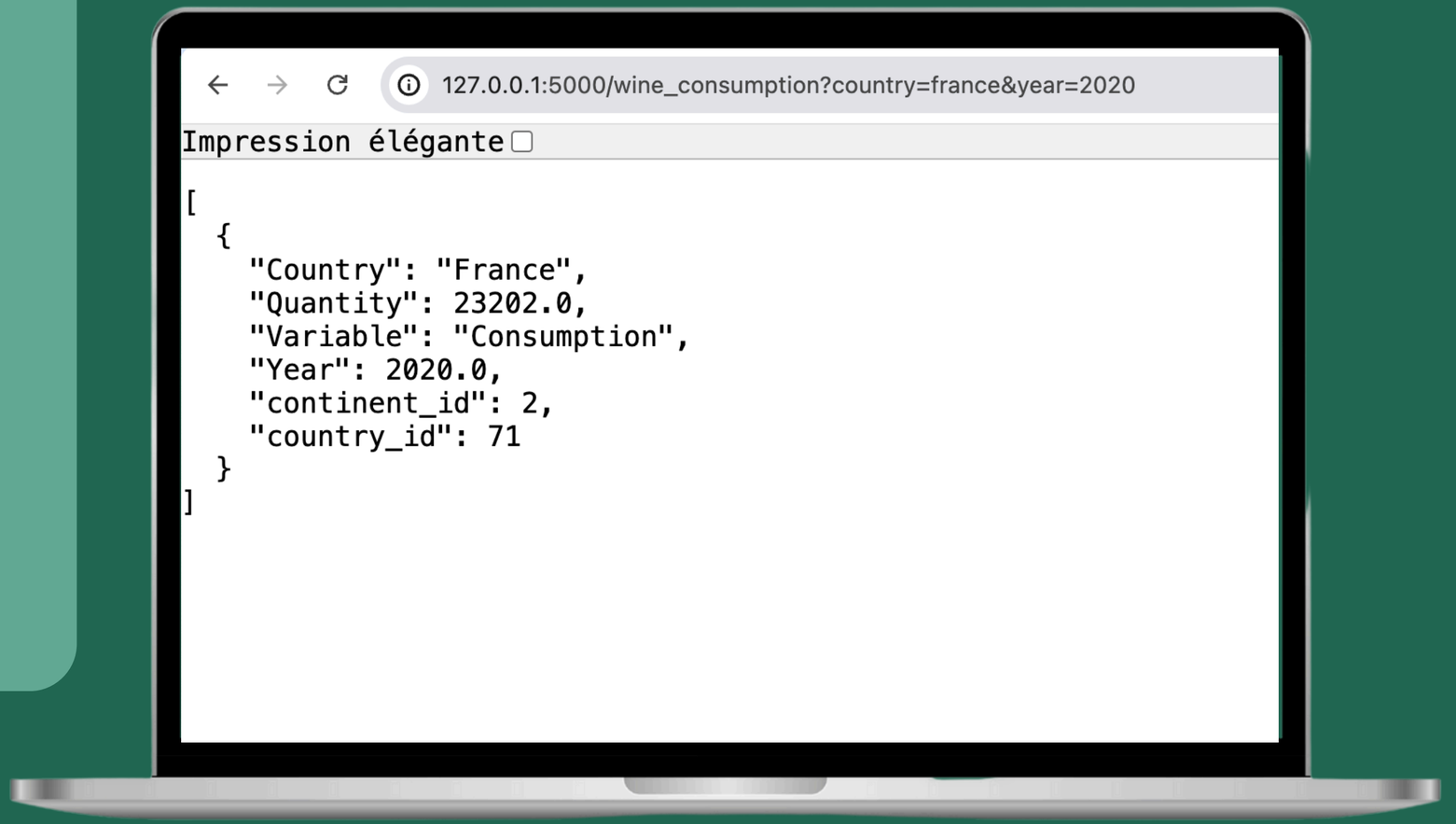
API Creation

Endpoints of my API:

wine_consumption
parameters: country & year

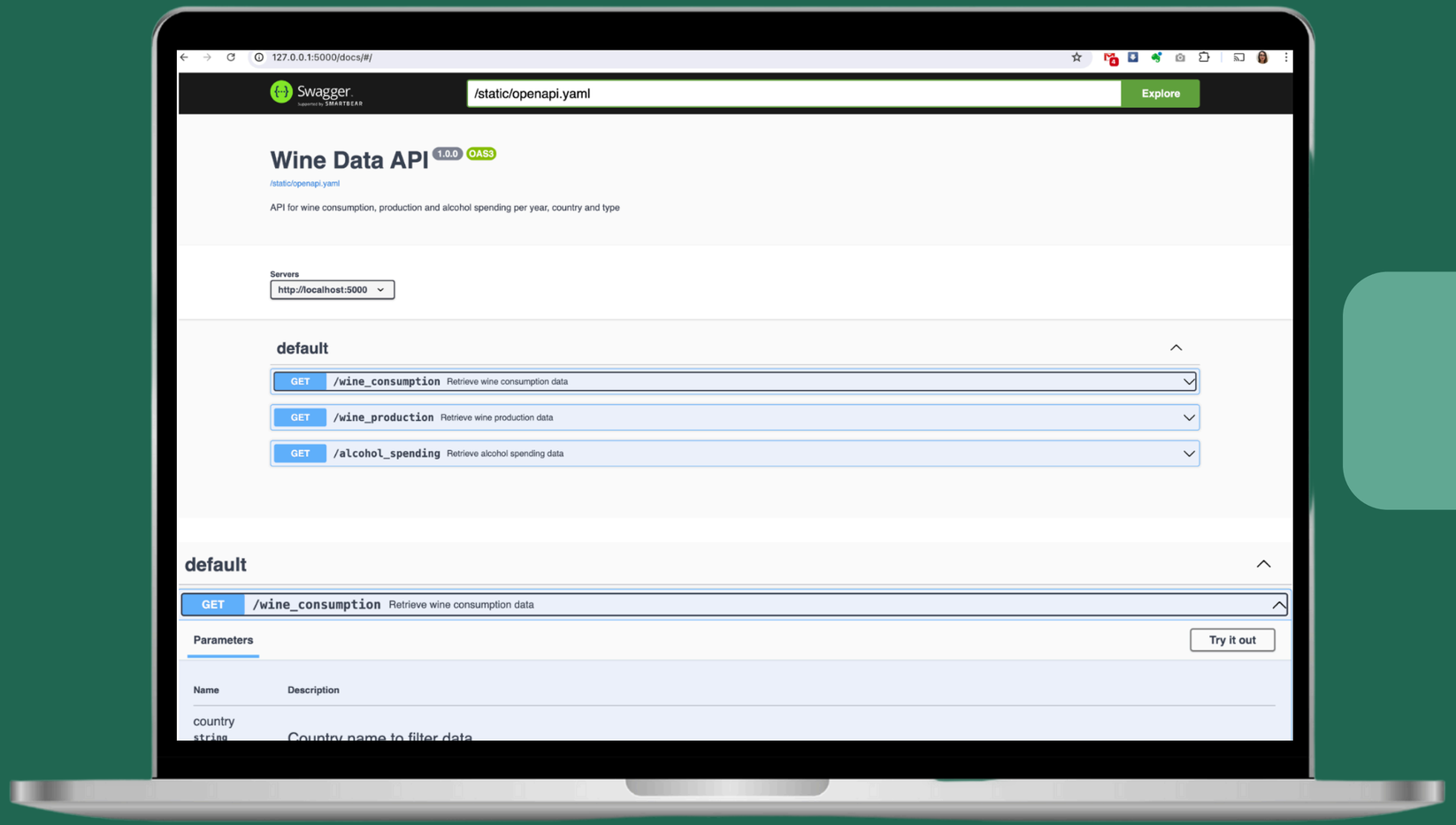
wine_production
parameters: country & year

alcohol_spending
parameters: type & year



API Demonstration

API Documentation



I used Swagger Open API to create a YAML file

SWAGGER Docs

05

EDA and Visualisations

In this section I will present you with some interesting data exploration and visualisations done in Tableau.

Wine in France

796 K

Hectares of vineyards (Second to Spain)

4.69 B

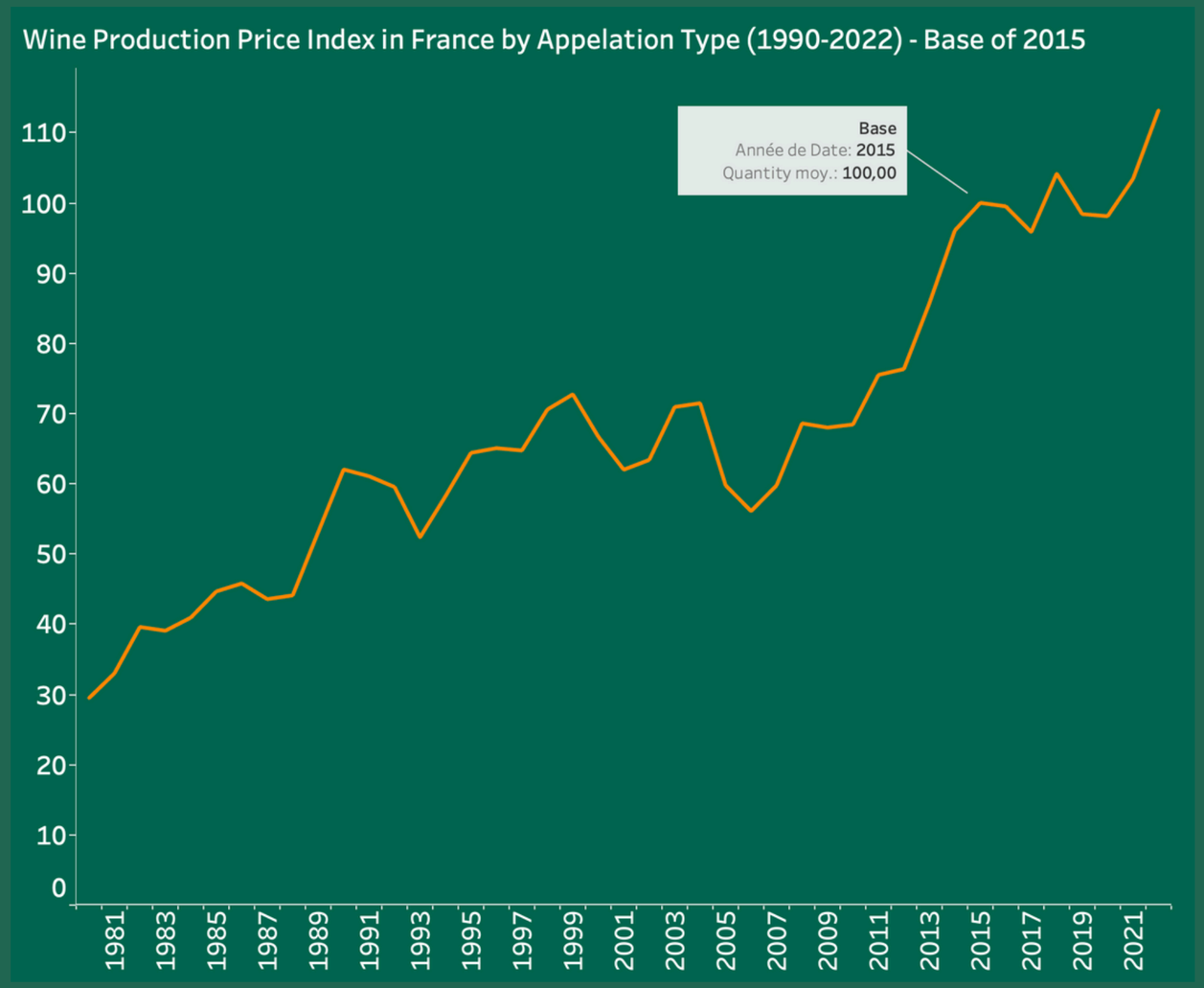
Liters of wine produced (Second to Italy)

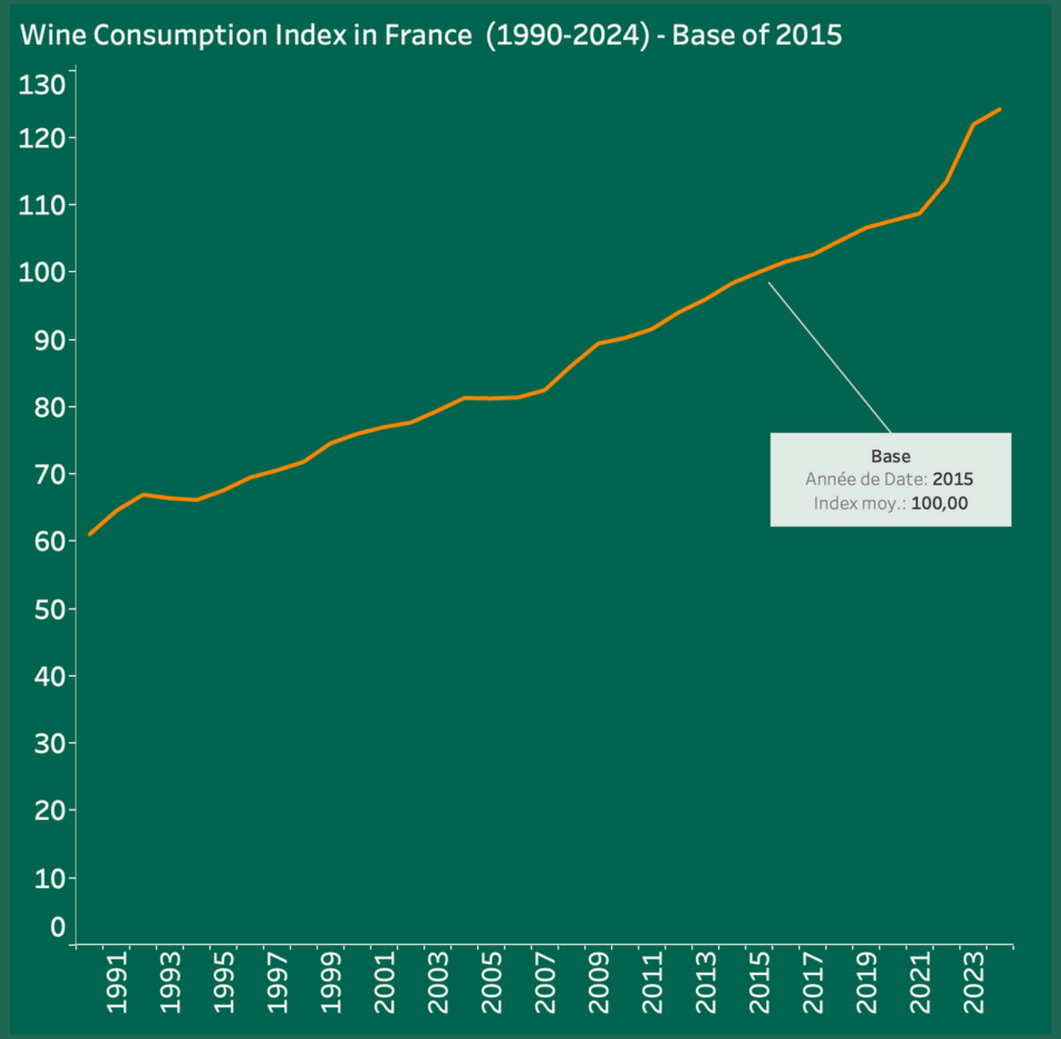
1.3 B

Liters of wine exported (Third to Italy and Spain)

Price Trends in France

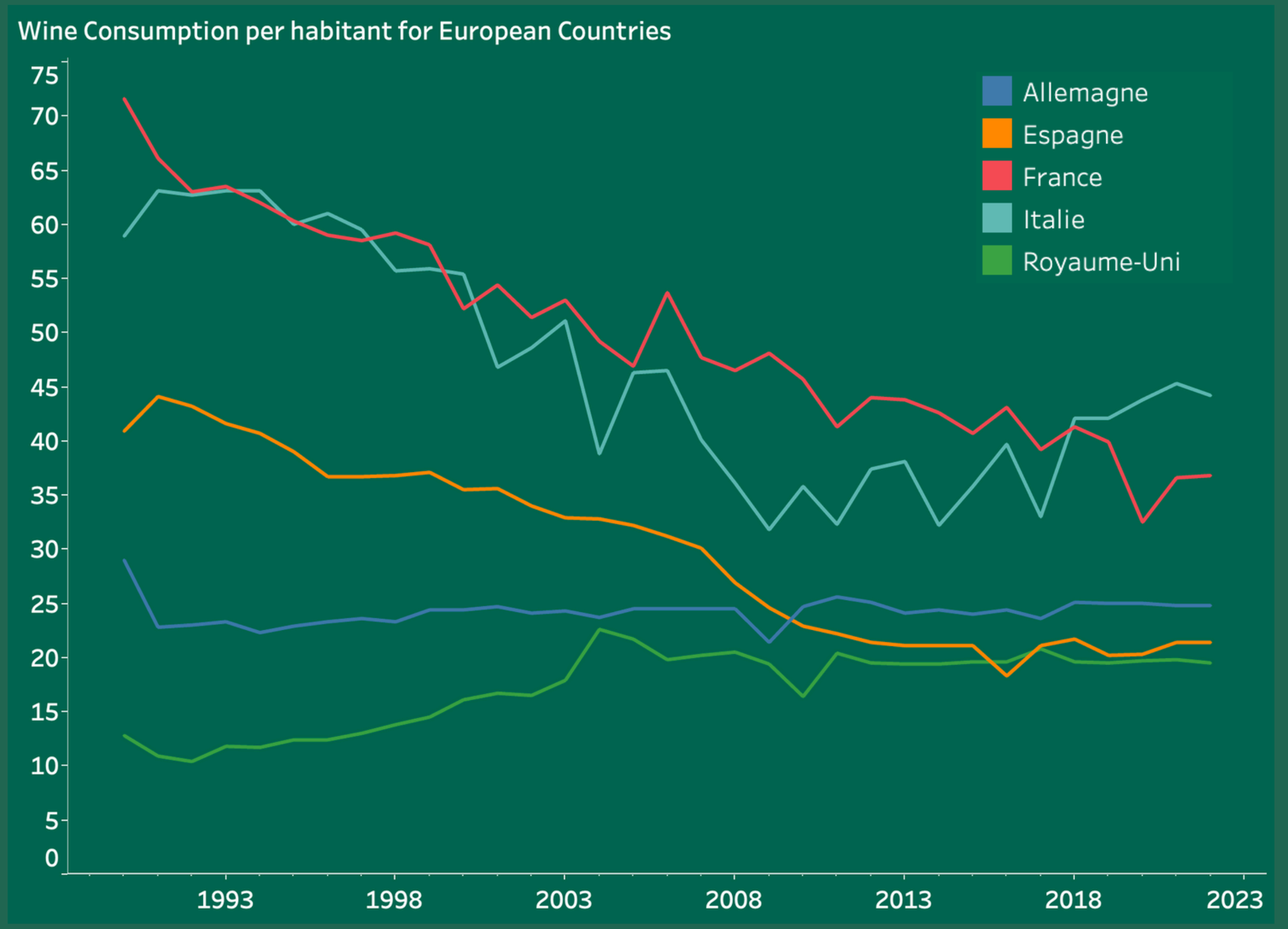
We see clear increase in production and comsumption prices



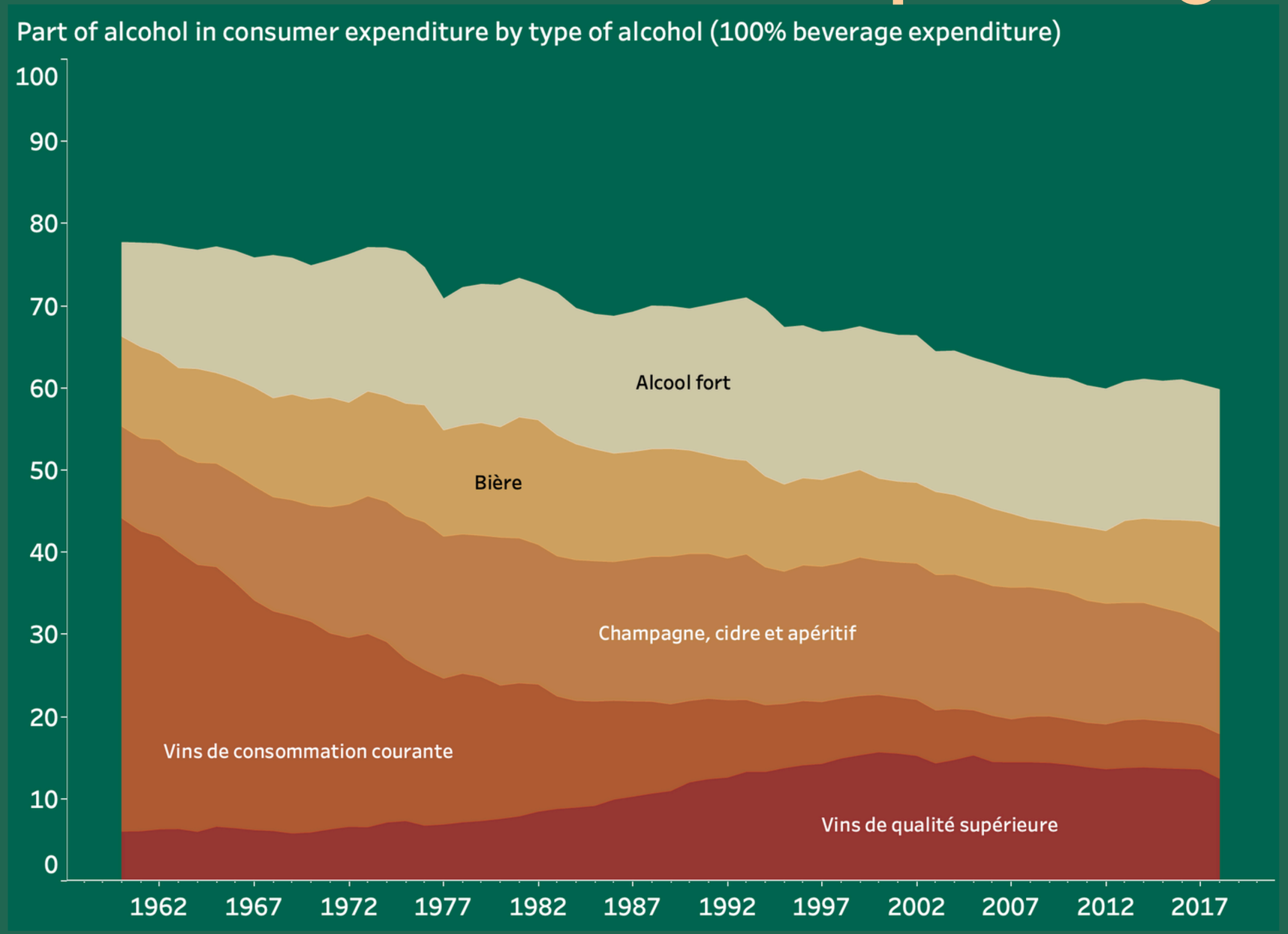


Consumption per habitant

We observe that consumption in France is clearly decreasing compared to other countries.



Alcohol Spending



We can observe that customers shifted from consuming table wine to consuming higher quality wine.

06

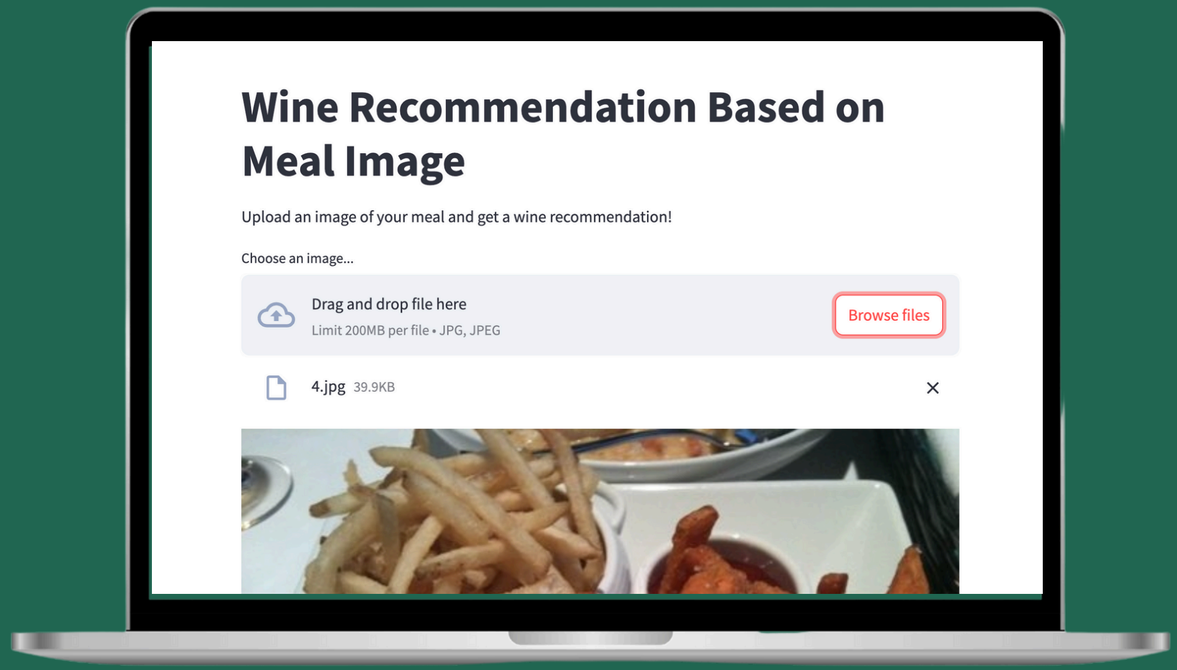
Machine Learning

This project aims to develop a Streamlit application that integrates a food recognition system with a wine recommendation engine.

Overview of ML

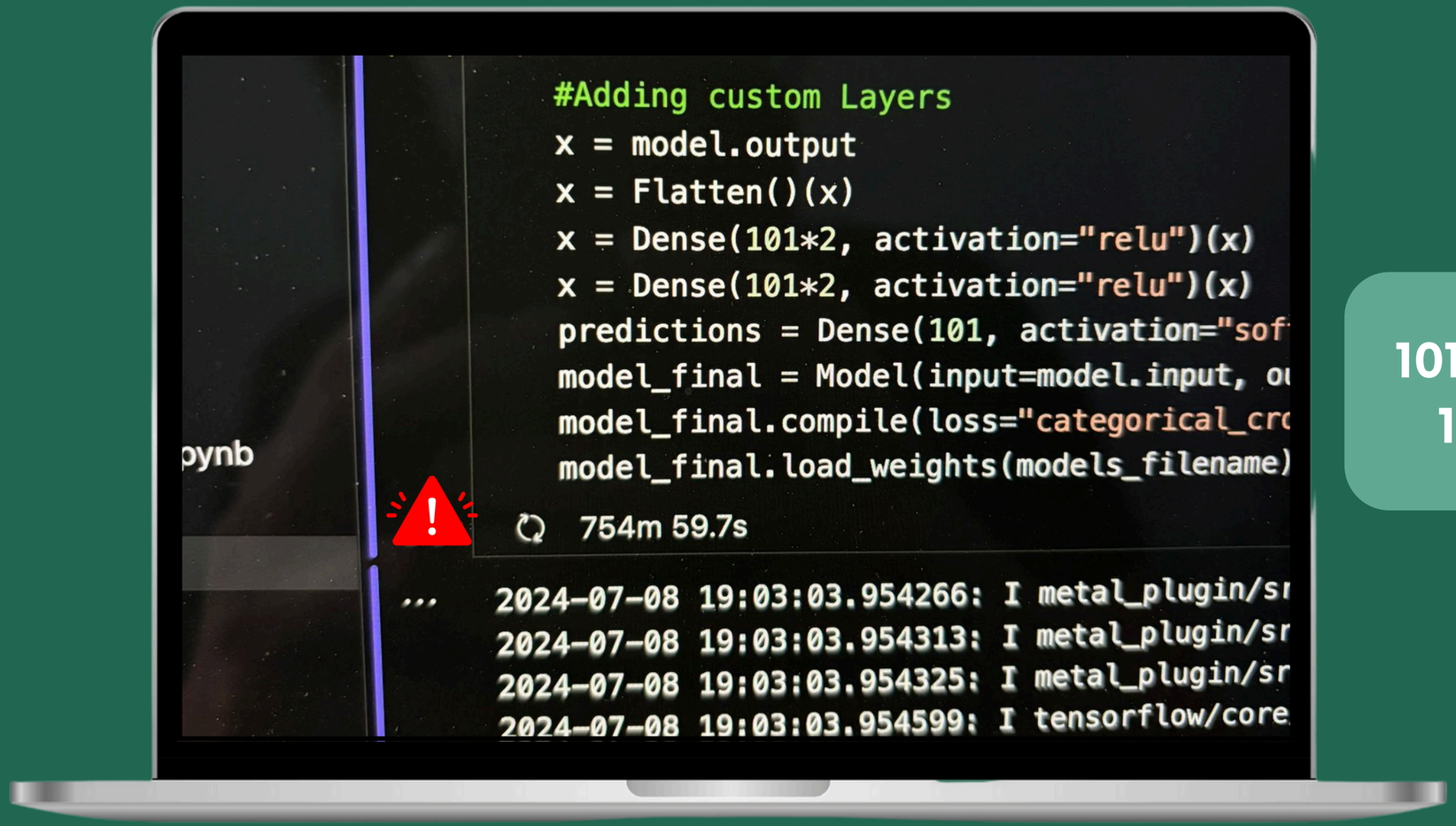
This project aims to develop a Streamlit application that provides users with personalised wine recommendations based on images of their meals by leveraging machine learning and deep learning techniques.

The project uses the FOOD 11 image database to train a convolutional neural network (CNN) model for accurate food recognition.



First attempt on the app

Food 101 Fail



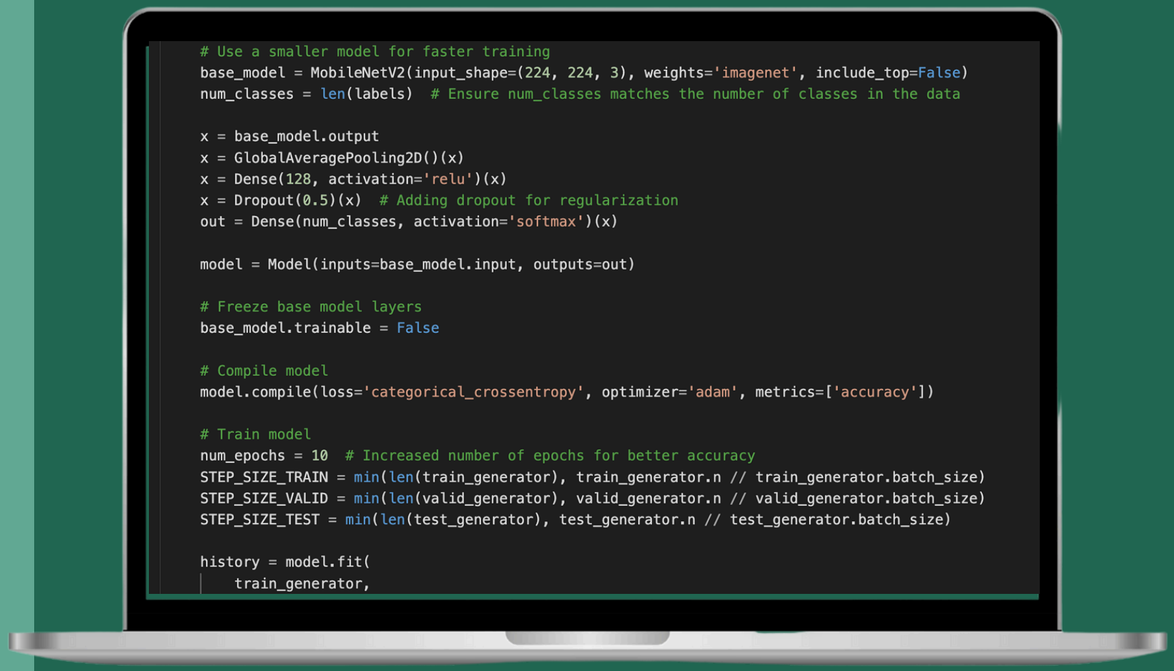
101000 images in 101 classes...
13 hours and still running...

Tuning the parameters to reduce time

Food 11 Dataset
16 643 food images grouped in 11 major food categories:
Bread, Dairy product, Dessert, Egg, Fried food, Meat, Noodles-Pasta, Rice, Seafood, Soup, Vegetable-Fruit

I tried two already trained lightweight convolutional neural network (CNN)
MobileNetV2
VGG16

30 epochs



```
# Use a smaller model for faster training
base_model = MobileNetV2(input_shape=(224, 224, 3), weights='imagenet', include_top=False)
num_classes = len(labels)  # Ensure num_classes matches the number of classes in the data

x = base_model.output
x = GlobalAveragePooling2D()(x)
x = Dense(128, activation='relu')(x)
x = Dropout(0.5)(x)  # Adding dropout for regularization
out = Dense(num_classes, activation='softmax')(x)

model = Model(inputs=base_model.input, outputs=out)

# Freeze base model layers
base_model.trainable = False

# Compile model
model.compile(loss='categorical_crossentropy', optimizer='adam', metrics=['accuracy'])

# Train model
num_epochs = 10  # Increased number of epochs for better accuracy
STEP_SIZE_TRAIN = min(len(train_generator), train_generator.n // train_generator.batch_size)
STEP_SIZE_VALID = min(len(valid_generator), valid_generator.n // valid_generator.batch_size)
STEP_SIZE_TEST = min(len(test_generator), test_generator.n // test_generator.batch_size)

history = model.fit(
    train_generator,
```

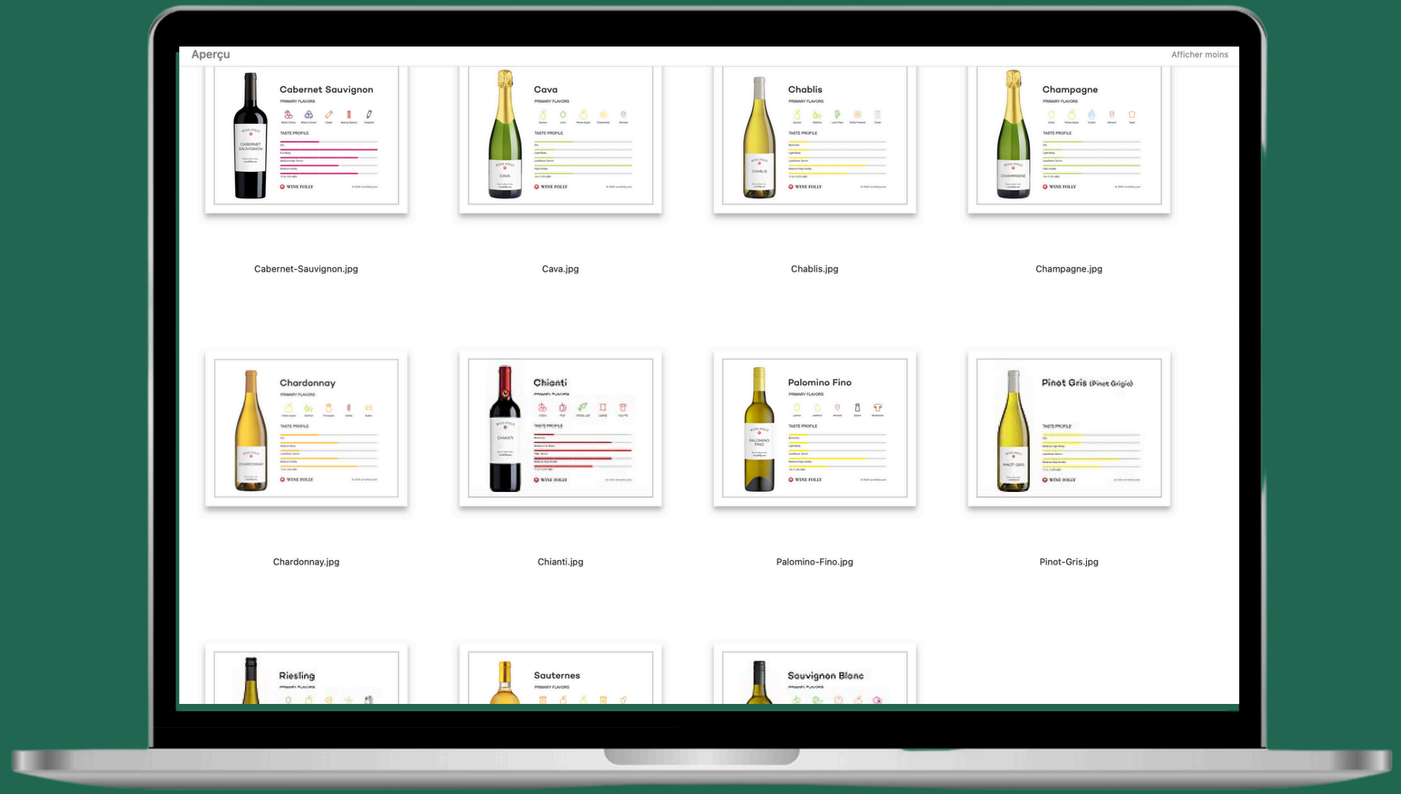
Python Code

0,77

Accuracy Score

(and possibility to do better with more time and processing power)

Creating a simple recommendation dataframe

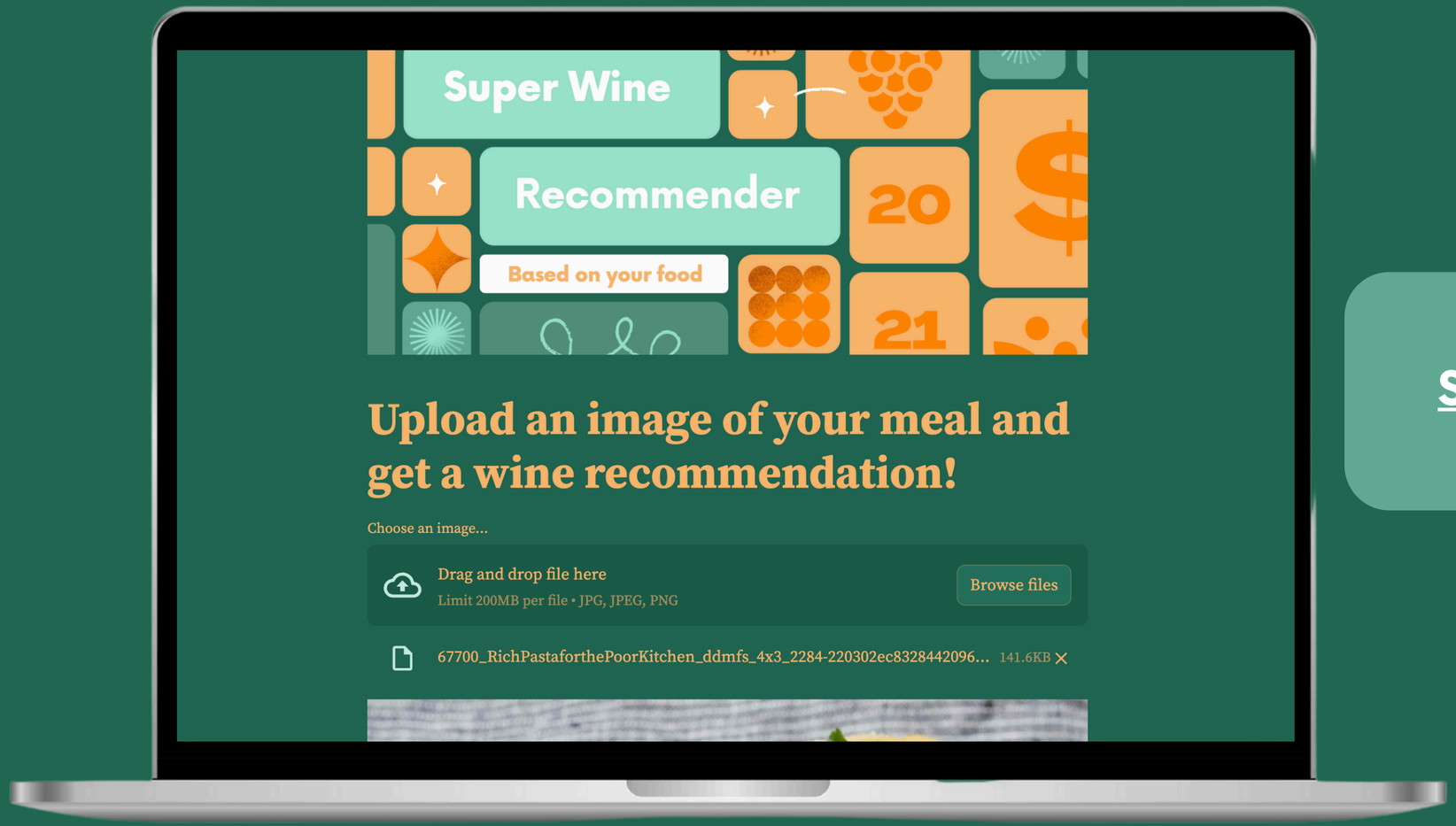


Wine Cards

Dataframe contains:
Food Class
Wine recommendation
Wine Card with more information

I linked it to the recognition model for my Streamlit application

Demo



[Super Wine Recommender]

Challenges

These are the areas where I faced the most difficulties.

01

Finding relevant data on internet. Cleaning and making it usable.

02

Web-Scraping with Selenium due to high protections of the page

03

Understanding the food recognition models and making it work

Next Steps

01 Train the model on more images to increase accuracy

02 Develop a more sophisticated dababase for wine recommendation

03 Improve the application UX design

04 Commercialize it :)

Thank You