Icon

Description automatically generated

**Data Analytics**

**Visually Similar Pictures Recommender applied to Travel Industry**

*“What if I could choose my next destination based on a picture I like.”*

*A picture containing logo

Description automatically generated*

Final Project, March 2023, delivered by:

**Kevin THAI**

**Table of Contents**

[Introduction 3](#_Toc129561616)

[Plan 4](#_Toc129561617)

[Data Collection 5](#_Toc129561618)

[**1.** **Why did I choose Pexels API to collect my data?** 5](#_Toc129561619)

[**2.** **How did I proceed to collect my data?** 5](#_Toc129561620)

[**a.** **JSON files bulk extraction using a loop.** 5](#_Toc129561621)

[**b.** **Turning the JSON files into a dataframe with selected columns** 6](#_Toc129561622)

[Data Cleaning 7](#_Toc129561623)

# **Introduction**

I believe that successful companies are those who can leverage technology in order to push the boundaries of tailored customer experiences. These companies or service-providers gain significant competitive advantages by always understanding their customers’ changing needs while being able to simplify, enhance and entertain them during the purchase decision journey.

To illustrate such “Customer First” vision, we could think about E-commerce’s success and how they leverage purchase history and browsing behavior data to provide personalized product recommendations. No doubt that these algorithms helped boosting basket value while saving customers’ time! We could also think about how fashion or beauty companies had to re-invent themselves (especially during sanitary restrictions times) to propose virtual try-on experiences for online shoppers leveraging the power of computer vision and augmented reality. Finally, grocery stores without checkout lines or cashiers did make newspaper headlines when they were launched!

Did you ever wonder how e-commerce websites would predict your next purchase or how a computer could recognize an object? As far as I am concerned, I was always intrigued and curious about the technology, data and machine learning techniques behind these enhanced customers experiences. So, I thought that I could explore and learn more on this topic during my last project at Ironhack by picking a project that will mix data collection and analysis, image recognition and clustering techniques.

My final project is called “**Visually Similar Travel Pictures Recommender**”. Here is the pitch: tired of visiting the same popular places and willing to dream further? The promise is simple: the user inputs a travel picture that inspires her/him (ex: beautiful landscape) and the machine returns couple of similar pictures around the world (hopefully)! Isn’t it exciting?

This project consists in two main parts:

1. Leverage Pexels API (Pexels is a website that offers a vast collection of free high-quality photos, videos and illustrations) to gather more than 3,900 pictures from top 50 most visited countries in the world associated with pictures features (ex: dimension, photographer, description) to generate preliminary insights.
2. Apply machine learning techniques to cluster these pictures based on image content similarities and suggest similar pictures based on user’s input.

The purpose of this document is to walk you through the first part of this project (from data collection, cleaning to exploratory data analysis with SQL and Python). The second part is not covered and will be showcased during the presentation!

Ladies and gentlemen, please fasten your seatbelts, we are preparing for take-off!

# **Plan**

The goal of this section is to provide an overview of this project’s key steps. It will hopefully help the reader to get a better understanding of following sections.

1. **Data Collection:**
   1. Small database from Kaggle about Top Visited Countries (49) in the world in 2021.
   2. Pexels API request to collect 80 pictures per country (49) resulting in a database of 3,600+ rows and 8 columns (ex: pictures\_id, dimension and description).
2. **Data Cleaning:**
   1. Mostly dealing with description (string) cleaning to remove special characters and standardize in lowercase in anticipation of word clouds usage.
3. **Exporting to MySQL database:**
   1. Preparing all the tables/entities in Python and exporting to MySQL.
   2. Building relationships between entities (ERD attached).
4. **Exploratory Data Analysis:**
   1. Explore the data with 5 SQL queries.
   2. For each query, follow-up on Python to explore more with Data Visualization.
5. **Machine Learning (not covered in this document)**

# **Data Collection**

## **Why did I choose Pexels API to collect my data?**

I chose Pixels API to build my pictures dataset for two main reasons:

1. Unlike other APIs, Pexels API is very convenient: it enables up to 200 requests per hour which is by far more than enough to build my dataset. Such “high rate-limit” gave me the possibility to do some tests and trials and to make mistakes without being “punished” and having to wait another day to pull more data.
2. A picture containing text, outdoor, way, road

   Description automatically generatedSince, I’m working on image recognition applied to Travel industry, I wanted to find high-quality and inspiring pictures. As you can see below, I found that Pexels top search results were appealing!

## **How did I proceed to collect my data?**

I first use a dataset found on Kaggle containing the 49 most visited countries in the world in 2021. As you can see below, I build a “Country\_travels” column and store all the values into a list. Later, I will iterate through this list to query Pexels API with “France travels” then “Spain travels” etc. as search keywords (using a loop). Note: I have added the keyword “travels” to improve the top search pictures results.

Table

Description automatically generated Graphical user interface, text

Description automatically generated

### **JSON files bulk extraction using a loop.**

First, I retrieved the data for only one country (France) to make sure I understand how did the API work and how does the data look like. Then, I decided to iterate through my previous list of most visited countries in the world defining the below function in Python:

Text

Description automatically generated

It’s worth noticing that I used the time.sleep(3) options within the loop so that I reduced the risk of being banned (even though the risk was very low). Then, I ran my function with following inputs 80 (which corresponds to the maximum number of pictures to retrieve per page) and 1 (meaning first page). After obtaining the aggregated JSON, I decided to save it on my local folder as a first “checkpoint”.

A screenshot of a computer

Description automatically generated with medium confidence

At this stage, I have a JSON aggregated files (a list of dictionaries) that gathers 80 pictures information per country so roughy 3,900 rows. Now, I need to format this file into a dataframe while selecting the columns I would like to keep.

### **Turning the JSON files into a dataframe with selected columns**

From the JSON files, I decided to keep the following columns to continue with my analysis:

['Pictures\_id', 'Width', 'Height', 'Full URL', 'Photographer\_id','Photographer', 'Avg Color', 'Description', 'Country\_id']

While the columns’ content is quite straightforward, please note that Full URL refers to the picture’s URL, Avg Color refers to the dominant color of the picture (Hex format) and Country\_id is an auto-incremental index (starting from 1 and increasing every 80 rows).

Text

Description automatically generated

After calling the function, I obtain my dataframe which looks like:

Graphical user interface, application, website

Description automatically generated

# **Data Cleaning**

### **Checking null values, removing duplicates and string standardization**

Since

Text

Description automatically generated with medium confidence

### **Preparing tables for export**

4 tables