Webscraping – DIA I

BOUCHIBA Emine
CHENIK Yassine

RAPPORT – PROJET

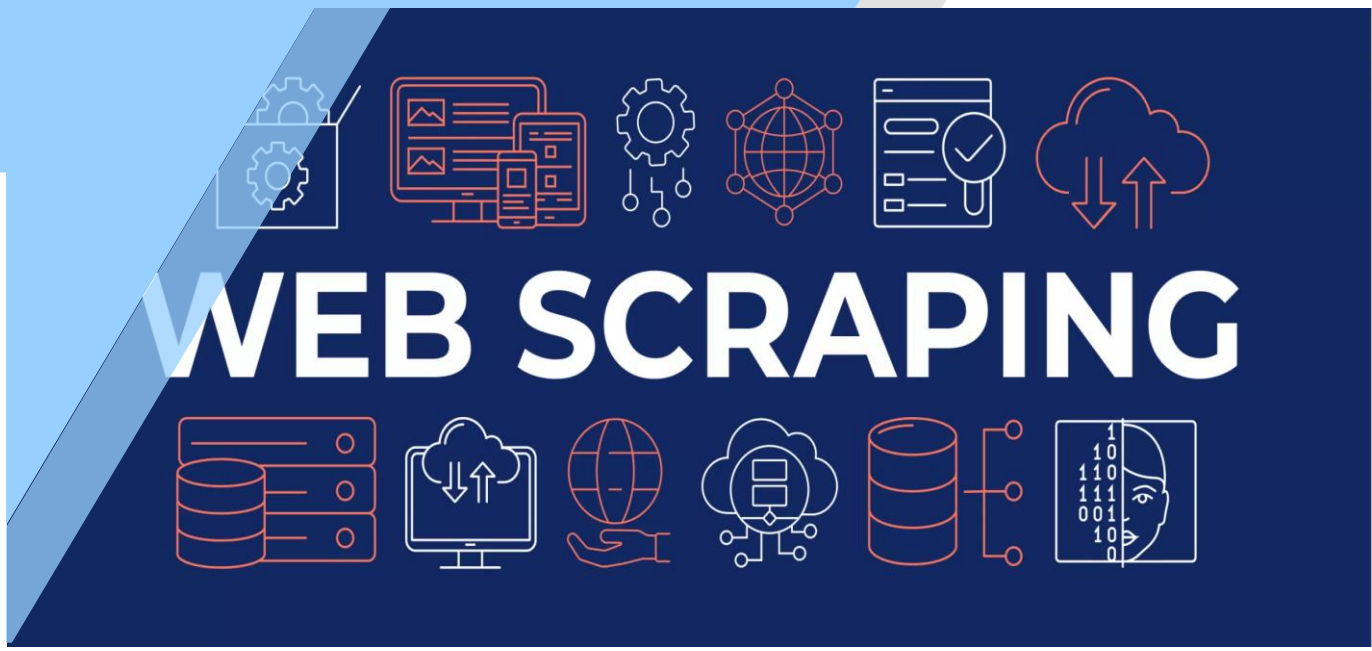


TABLE DES MATIERES

Project Definition 2-3
Webscraping Part 4-5
With Excel 4
With beautifulSoup and Selenium 5
Web Interface 6-9
Getting the startups locations 6-7
Generation of the RSE Scores 8
Streamlit Website Generation 9
Demo Video 10

Project Definition

Our goal :

- ✓ We sincerely believe that there can be no change in society unless we start small, namely with startups. If we change the way we invest, namely by investing in promising startups that are also respectful of RSE standards, it is only then that there can be real effective change. That's why our goal through this project is to build an interface that can provide information about medical startups and, more than that, an interface that can help us choose the startups that are most respectful of RSE.

Project Roadmap :

- ✓ Initially, we planned to scrape data from LinkedIn to gather information about medical startups. However, we soon realized that LinkedIn's data was insufficient for our interface, focusing more on business-related details rather than practical information.
- ✓ Consequently, we shifted our approach to constructing our own list of medical startups. We explored alternative data scraping techniques beyond BeautifulSoup and Selenium, opting to use Excel. This method involved scraping data from various medical websites known for their extensive startup information, such as Biocat, Biopartner, Bionow, Atlantapole, ci3, and others.
- ✓ Having obtained data from these websites, we then employed a combination of BeautifulSoup and Selenium to scrape URL links from our previously created Excel sheet. This enabled us to extract information directly from the startups' websites.
- ✓ With this approach, we successfully compiled a comprehensive dataset containing all the necessary information. Our next focus was on developing the web interface.
- ✓ For the interface, we chose to use Streamlit, diverging from our usual tools to explore new methods. Streamlit proved to be an efficient choice, requiring only a single Python script instead of separate HTML and CSS files. This streamlined the process of creating faster, more customizable websites.

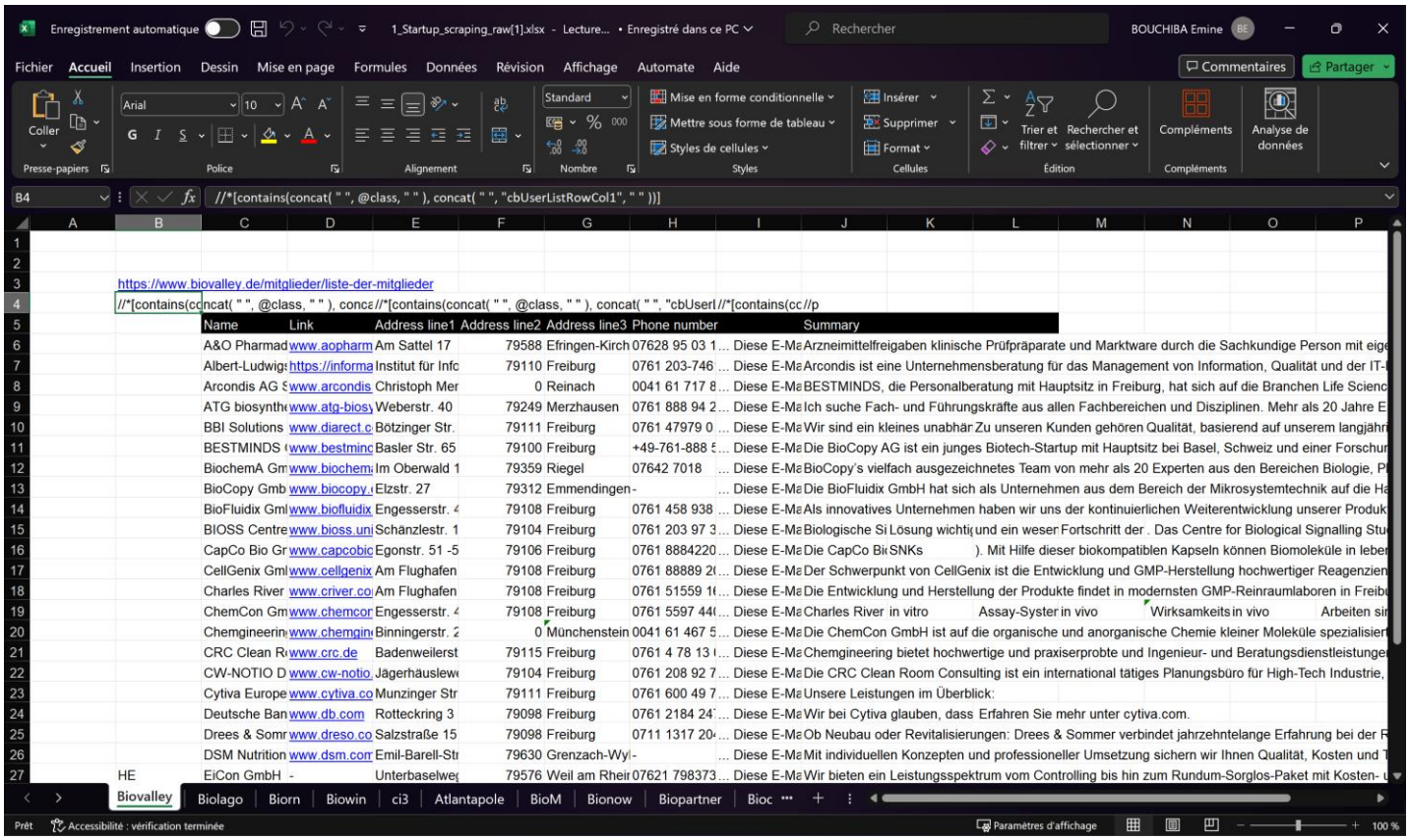
Project Definition

- ✓ With the web interface solution decided, we then considered which features to include :
  - **1st feature :** Displaying information about all startups, filterable by the « sector » column.
  - **2nd feature :** Visualizing startup locations using the « geopy » library to obtain « latitude » and « longitude » data from string inputs.
  - **3rd feature :** Presenting each startup's RSE score, filtered by the 'sector' column. This involved using NLP models from Hugging Face to identify words in the « specialization » column that closely match the RSE lexicon.

Webscraping Part

With Excel :

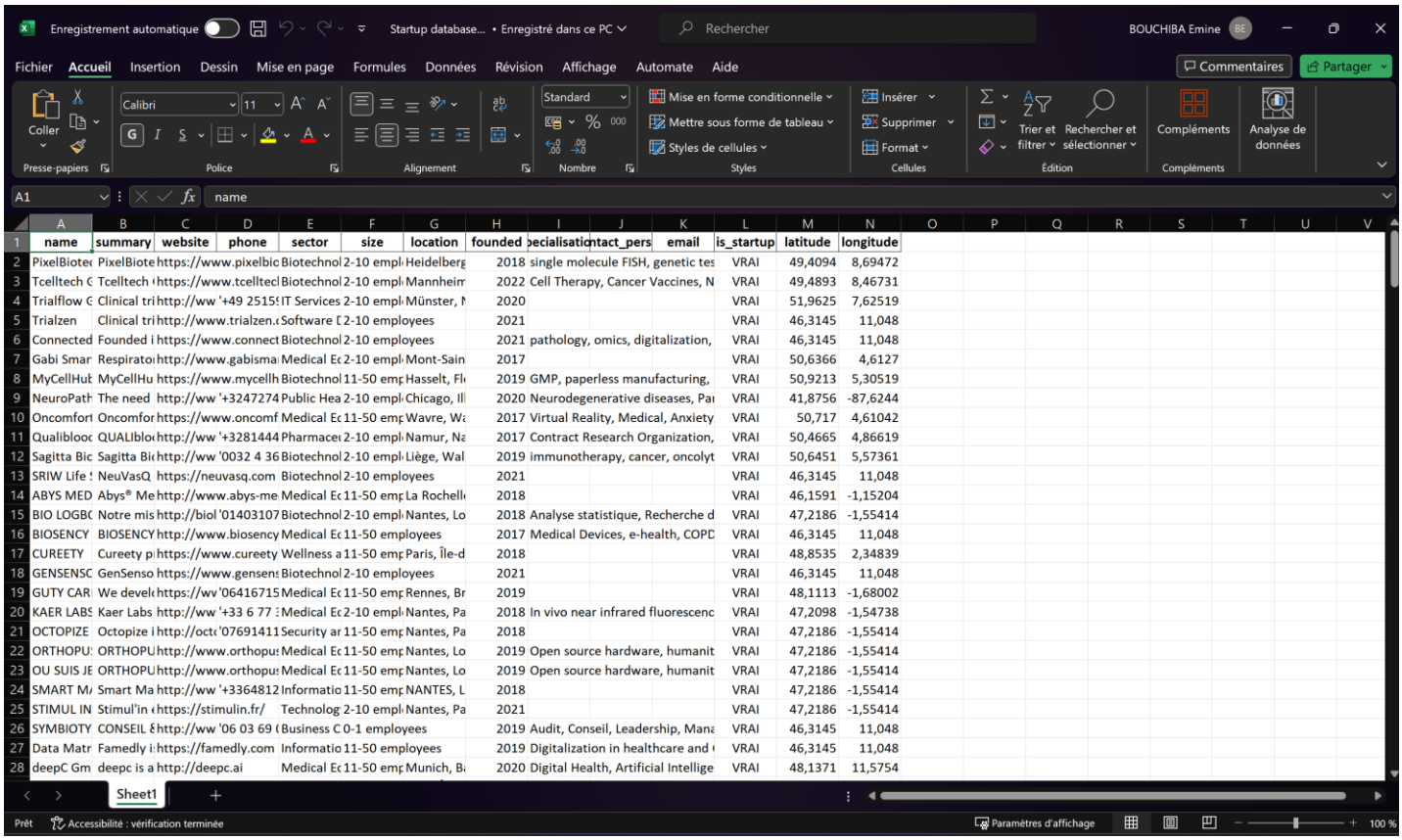
- ✓ **We used this function in order to scrap the data :**



Webscraping Part

With BeautifulSoup and Selenium :

- ✓ **The code is in the notebook, however this is the given .xlsx file :**



name	summary	website	phone	sector	size	location	founded	specialisation	contact_pers	email	is_startup	latitude	longitude
PixelBiotec	PixelBiote	https://www.pixelbic		Biotechnol	2-10 empl	Heidelberg	2018	single molecule FISH, genetic tes			VRAI	49,4094	8,69472
Tcelltech C	Tcelltech (	https://www.tcelltecl		Biotechnol	2-10 empl	Mannheim	2022	Cell Therapy, Cancer Vaccines, N			VRAI	49,4893	8,46731
Trialflow C	Clinical tri	http://ww '+49 2515		IT Services	2-10 empl	Münster, N	2020				VRAI	51,9625	7,62519
Trialzen	Clinical tri	http://www.trialzen.(		Software [	2-10 employees		2021				VRAI	46,3145	11,048
Connected	Founded i	https://www.connect		Biotechnol	2-10 employees		2021	pathology, omics, digitalization,			VRAI	46,3145	11,048
Gabi Smar	Respirato	http://www.gabisma		Medical Ec	2-10 empl	Mont-Sain	2017				VRAI	50,6366	4,6127
MyCellHub	MyCellHu	https://www.mycellh		Biotechnol	11-50 emp	Hasselt, Fl	2019	GMP, paperless manufacturing,			VRAI	50,9213	5,30519
NeuroPath	The need	http://ww '+3247274		Public Hea	2-10 empl	Chicago, Il	2020	Neurodegenerative diseases, Pa			VRAI	41,8756	-87,6244
Oncomfort	Oncomfor	https://www.oncomf		Medical Ec	11-50 emp	Wavre, Wa	2017	Virtual Reality, Medical, Anxiety			VRAI	50,717	4,61042
Qualiblooc	QUALIblo	http://ww '+3281444		Pharmace	2-10 empl	Namur, Na	2017	Contract Research Organization,			VRAI	50,4665	4,86619
Sagitta Bic	Sagitta Bi	http://ww '0032 4 36		Biotechnol	2-10 empl	Liège, Wal	2019	immunotherapy, cancer, oncolyt			VRAI	50,6451	5,57361
SRIW Life S	NeuVasQ	https://neuvasq.com		Biotechnol	2-10 employees		2021				VRAI	46,3145	11,048
ABYS MED	Abys® Me	http://www.abys-me		Medical Ec	11-50 emp	La Rochell	2018				VRAI	46,1591	-1,15204
BIO LOGBC	Notre mis	http://biol '01403107		Biotechnol	2-10 empl	Nantes, Lo	2018	Analyse statistique, Recherche d			VRAI	47,2186	-1,55414
BIOSENCY	BIOSENCY	http://www.biosency		Medical Ec	11-50 employees		2017	Medical Devices, e-health, COPD			VRAI	46,3145	11,048
CUREETY	Cureety p	https://www.cureety		Wellness a	11-50 emp	Paris, Île-d	2018				VRAI	48,8535	2,34839
GENSENSC	GenSenso	https://www.gensens		Biotechnol	2-10 employees		2021				VRAI	46,3145	11,048
GUTY CAR	We devel	https://ww '06416715		Medical Ec	11-50 emp	Rennes, Br	2019				VRAI	48,1113	-1,68002
KAER LABS	Kaer Labs	http://ww '+33 6 77		Medical Ec	2-10 empl	Nantes, Pa	2018	In vivo near infrared fluorescenc			VRAI	47,2098	-1,54738
OCTOPIZE	Octopize i	http://octo '07691411		Security ar	11-50 emp	Nantes, Pa	2018				VRAI	47,2186	-1,55414
ORTHOPU	ORTHOPU	http://www.orthopu		Medical Ec	11-50 emp	Nantes, Lo	2019	Open source hardware, humanit			VRAI	47,2186	-1,55414
OU SUIS JE	ORTHOPU	http://www.orthopu		Medical Ec	11-50 emp	Nantes, Lo	2019	Open source hardware, humanit			VRAI	47,2186	-1,55414
SMART MA	Smart Ma	http://ww '+3364812		Informatio	11-50 emp	NANTES, L	2018				VRAI	47,2186	-1,55414
STIMUL IN	Stimul'in (	https://stimulin.fr/		Technolog	2-10 empl	Nantes, Pa	2021				VRAI	47,2186	-1,55414
SYMBIOTY	CONSEIL &	http://ww '06 03 69 (		Business C	0-1 employees		2019	Audit, Conseil, Leadership, Mana			VRAI	46,3145	11,048
Data Matr	Famedly i	https://famedly.com		Informatio	11-50 employees		2019	Digitalization in healthcare and (			VRAI	46,3145	11,048
deepC Gm	deepc is a	http://deepc.ai		Medical Ec	11-50 emp	Munich, Ba	2020	Digital Health, Artificial Intellige			VRAI	48,1371	11,5754

Web Interface

Getting the startups locations :

- ✓ **Using the GeoPy Library in order to get the startups exact location from an str input :**

jupyter Getting_StartupsGeoLocalisation Last Checkpoint: jeudi dernier à 17:03 (autosaved)

Logout

File Edit View Insert Cell Kernel Widgets Help Trusted Python 3 (ipykernel)

Run Code

In [2]:
```
!pip install folium
!pip install geopy
```

```
Requirement already satisfied: folium in c:\users\emine\anaconda3\lib\site-packages (0.15.1)
Requirement already satisfied: requests in c:\users\emine\anaconda3\lib\site-packages (from folium) (2.28.1)
Requirement already satisfied: branca>=0.6.0 in c:\users\emine\anaconda3\lib\site-packages (from folium) (0.7.0)
Requirement already satisfied: jinja2>=2.9 in c:\users\emine\anaconda3\lib\site-packages (from folium) (3.1.2)
Requirement already satisfied: numpy in c:\users\emine\anaconda3\lib\site-packages (from folium) (1.23.5)
Requirement already satisfied: xyzservices in c:\users\emine\anaconda3\lib\site-packages (from folium) (2023.10.1)
Requirement already satisfied: MarkupSafe>=2.0 in c:\users\emine\anaconda3\lib\site-packages (from jinja2>=2.9->folium) (2.
1.1)
Requirement already satisfied: certifi>=2017.4.17 in c:\users\emine\anaconda3\lib\site-packages (from requests->folium) (202
2.12.7)
Requirement already satisfied: charset-normalizer<3,>=2 in c:\users\emine\anaconda3\lib\site-packages (from requests->foliu
m) (2.0.4)
Requirement already satisfied: idna<4,>=2.5 in c:\users\emine\anaconda3\lib\site-packages (from requests->folium) (3.4)
Requirement already satisfied: urllib3<1.27,>=1.21.1 in c:\users\emine\anaconda3\lib\site-packages (from requests->folium)
(1.26.14)
Requirement already satisfied: geopy in c:\users\emine\anaconda3\lib\site-packages (2.4.1)
Requirement already satisfied: geographiclib<3,>=1.52 in c:\users\emine\anaconda3\lib\site-packages (from geopy) (2.0)
```

In [5]:
```python
import pandas as pd
import folium
from geopy.geocoders import Nominatim
from geopy.exc import GeocoderTimedOut, GeocoderQuotaExceeded
import time

def load_data():
    path = "C:/Users/emine/Documents/Ecole/A5/Webscraping & Data Processing/Project/data/Startup database test.xlsx"
    data = pd.read_excel(path)
    return data

def get_location(address):
    geolocator = Nominatim(user_agent="geoapiforWebScraping")
    try:
        location = geolocator.geocode(address)
        if location:
            return location.latitude, location.longitude
    except (GeocoderTimedOut, GeocoderQuotaExceeded):
        return None, None
    return None, None

df = load_data()

map = folium.Map(location=[20, 0], zoom_start=2)

for index, row in df.iterrows():
    lat, lon = get_location(row['location'])
    if lat and lon:
        folium.Marker([lat, lon], popup=row['name']).add_to(map)
        # Ajouter directement les coordonnées au DataFrame
        df.at[index, 'latitude'] = lat
        df.at[index, 'longitude'] = lon
    time.sleep(1)

# Afficher la carte
map

# Sauvegarder le DataFrame avec les coordonnées
output_path = 'C:/Users/emine/Documents/Ecole/A5/Webscraping & Data Processing/Project/data/Startup database.xlsx'
df.to_excel(output_path, index=False)
```

Web Interface

Displaying the startups locations :

✓ **Using the folium library from python :**

jupyter Displaying_StartupsGeoLocalisation Last Checkpoint: jeudi dernier à 18:00 (autosaved)

File Edit View Insert Cell Kernel Widgets Help

In [1]:

```
import pandas as pd
import folium

# Fonction pour charger les données depuis le fichier Excel
def load_coordinates():
    path = "C:\\Users\\emine\\Documents\\Ecole\\A5\\Webscraping & Data Processing\\Project\\data\\startup_coordinates.xlsx"
    data = pd.read_excel(path)
    return data

# Charger les données de coordonnées
df_coordinates = load_coordinates()

# Créer la carte
map = folium.Map(location=[20, 0], zoom_start=2)

# Ajouter des marqueurs pour chaque startup
for _, row in df_coordinates.iterrows():
    folium.Marker([row['latitude'], row['longitude']], popup=row['name']).add_to(map)

# Afficher la carte avec Streamlit
map
```

Out[1]:

In [ ]:

```
import pandas as pd
import folium

def load_data():
    path = "C:/Users/emine/Documents/Ecole/A5/Webscraping & Data Processing/Project/data/Startup database final.xlsx"
    data = pd.read_excel(path)
    return data

# Obtenir les coordonnées géographiques
def get_location(address):
    # Ici, vous pouvez ajouter votre logique de géocodage
    # ou utiliser une méthode pour extraire les coordonnées si elles sont déjà présentes dans les données
    pass

# Appel de la fonction de chargement de données
df = load_data()

# Sélectionner un secteur via un menu déroulant
sector = st.selectbox('Choisissez un secteur', df['sector'].unique())

# Filtrer les données par secteur
filtered_data = df[df['sector'] == sector]

# Afficher les données filtrées
st.subheader(f'Données pour le secteur : {sector}')
st.write(filtered_data[['name', 'phone', 'size', 'website', 'founded', 'location']])

# Création de la carte pour le secteur sélectionné
map = folium.Map(location=[20, 0], zoom_start=2)
for _, row in filtered_data.iterrows():
    lat, lon = get_location(row['location'])
    if lat and lon:
        folium.Marker([lat, lon], popup=row['name']).add_to(map)


# Fonction pour charger les données depuis le fichier Excel
def load_coordinates():
    path = "C:\\Users\\emine\\Documents\\Ecole\\A5\\Webscraping & Data Processing\\Project\\data\\startup_coordinates.xlsx"
    data = pd.read_excel(path)
    return data

# Charger les données de coordonnées
df_coordinates = load_coordinates()

# Créer la carte
map = folium.Map(location=[20, 0], zoom_start=2)

# Ajouter des marqueurs pour chaque startup
for _, row in df_coordinates.iterrows():
    folium.Marker([row['latitude'], row['longitude']], popup=row['name']).add_to(map)
```

Web Interface

Generation of the RSE Scores :

- ✓ **Using the transformers library created by Hugging Face for NLP integration in python :**

jupyter RSE_Score Last Checkpoint: il y a 15 heures (autosaved) Logout

File Edit View Insert Cell Kernel Widgets Help Not Trusted Python 3 (ipykernel)

Run Code

In [1]:
```
!pip install sentence_transformers
import pandas as pd

# Charger les données
def load_data():
    path = "C:/Users/emine/Documents/Ecole/A5/Webscraping & Data Processing/Project/data/Startup database.xlsx"
    return pd.read_excel(path)

df = load_data()

# Extraire les mots de la colonne 'specialization'
all_words = set()
for specialization in df['specialisation'].dropna():
    words = specialization.split()
    all_words.update(words)

# Convertir l'ensemble en liste pour faciliter l'utilisation
unique_words = list(all_words)

# Afficher les mots uniques
print(unique_words)
```

```
Requirement already satisfied: sentence_transformers in c:\users\emine\anaconda3\lib\site-packages (2.2.2)
Requirement already satisfied: torch>=1.6.0 in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers)
(2.1.2)
Requirement already satisfied: sentencepiece in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers)
(0.1.99)
Requirement already satisfied: huggingface-hub>=0.4.0 in c:\users\emine\anaconda3\lib\site-packages (from sentence_transf
ormers) (0.10.1)
Requirement already satisfied: scipy in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers) (1.10.0)
Requirement already satisfied: scikit-learn in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers)
(1.2.1)
Requirement already satisfied: torchvision in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers) (0.
16.2)
Requirement already satisfied: transformers<5.0.0,>=4.6.0 in c:\users\emine\anaconda3\lib\site-packages (from sentence_tr
ansformers) (4.24.0)
Requirement already satisfied: nltk in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers) (3.7)
Requirement already satisfied: tqdm in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers) (4.64.1)
Requirement already satisfied: numpy in c:\users\emine\anaconda3\lib\site-packages (from sentence_transformers) (1.23.5)
Requirement already satisfied: typing-extensions>=3.7.4.3 in c:\users\emine\anaconda3\lib\site-packages (from huggingface
-hub>=0.4.0->sentence_transformers) (4.9.0)
```

In [2]:
```
from transformers import AutoTokenizer, AutoModelForSequenceClassification
from scipy.spatial.distance import cosine
from sentence_transformers import SentenceTransformer
import numpy as np

# Initialisation du modèle de similarité sémantique
model = SentenceTransformer('sentence-transformers/all-MiniLM-L6-v2')

# Liste de mots-clés RSE
rse_keywords = ["sustainability", "ethical", "social", "environment", "health", "well-being", "community",
                "green", "renewable", "eco-friendly", "inclusive", "diversity", "equality", "charity", "volunteer"]


# Convertir les listes de mots en embeddings
rse_embeddings = model.encode(rse_keywords)
input_embeddings = model.encode(unique_words)

# Trouver les mots les plus similaires
threshold = 0.65  # Seuil de similarité, ajustable selon les besoins
rse_related_words = []

for word, word_embedding in zip(unique_words, input_embeddings):
    similarities = [1 - cosine(word_embedding, rse_embedding) for rse_embedding in rse_embeddings]
    if max(similarities) > threshold:
        rse_related_words.append(word)

print(rse_related_words)
```

```
['Wellness,', 'wellbeing,', 'Health,', 'Well-being,', 'health,', 'Wellbeing,', 'Community', 'Medical', 'healthcare,', 'Commu
nities,', 'wellness', 'Health', 'Environmental', 'sustainability,', 'Medicine', 'healthcare', 'sustainable', 'Wellness', 'we
llness,', 'medical', 'Healthcare,', 'medicine', 'Fundraising,', 'wellbeing', 'Healthcare', 'Communities', 'health']
```

Web Interface

Streamlit Website Generation :

✓ **Installation of the necessary libraries :**

```
pip install streamlit pandas openpyxl matplotlib seaborn folium streamlit-folium geopy streamlit_option_menu scipy sentence_transformers
```

✓ **Running Streamlit locally :**

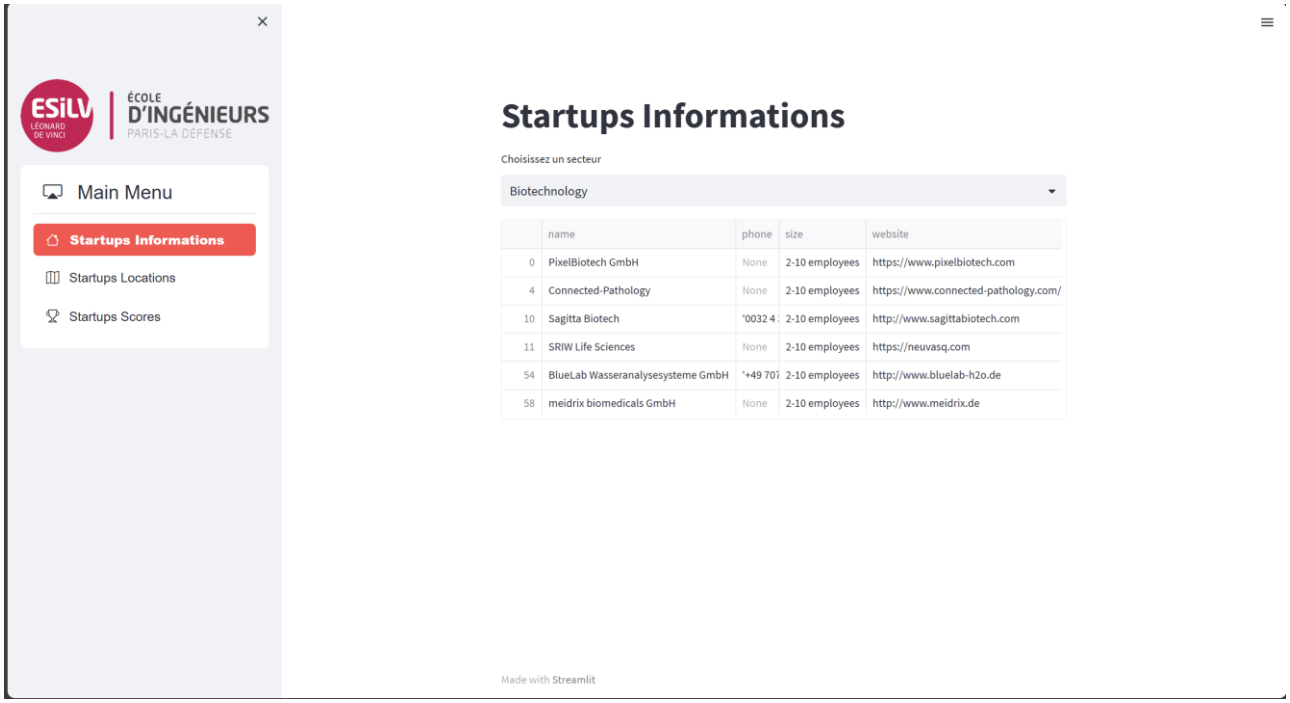
```
Invite de commandes - strean
Microsoft Windows [version 10.0.22621.2861]
(c) Microsoft Corporation. Tous droits réservés.

C:\Users\emine>streamlit run "C:\Users\emine\Documents\Ecole\A5\Webscraping & Data Processing\Project\code\Project_WebInterface.py"

  You can now view your Streamlit app in your browser.

  Local URL: http://localhost:8501
  Network URL: http://192.168.1.26:8501
```

✓ **Having the web interface launched and ready to be used :**



Demo Video

Streamlit Website Demo :

- ✓ Demo Video