Explications projets :

2 API -> Traitement en Python -> Bucket GCP (Staging) -> Table BigQuery (DataWareHouse) -> DataViz

Sécurité :

* Google cloud storage + Big query + serveur sur paris
* GitGuardian pour le projet et éviter les leaks de mdp

Looker studio : utilisé actuellement par les équipes de Good air. Néanmoins, il n’y a pas de MAJ sur les graphiques (Cf Axel), ducoup on à préféré partir sur un autre outil de dataviz : [ICI]

Villes choisies pour le MVP :

* Lille : 50.633333, 3.066667
* Nice : 43.700000, 7.250000
* Strasbourg : 48.583328, 7.75
* Paris : 48.866667, 2.333333
* Rennes : 48.083328, -1.68333
* Nantes : 47.216671, -1.55
* Marseille : 43.300000, 5.400000
* Toulouse : 43.599998, 1.43333
* Lyon : 45.750000, 4.850000
* Bordeaux : 44.833328, -0.56667

Chaque heure, l’algorithme effectue pour chaque ville un premier call API sur l’air quality, cela retourne une première partie des données, puis la deuxième call API sur WeatherMap est fait, permettant d’avoir le reste des données comme la température.

On souhaitera normaliser ensuite les données, passant tout en français, en choisissant les noms de tables, colonnes etc. Et en traitant la donnée, changeant les types et les traitant les valeurs nulles.

On va devoir rédiger un rapport pour expliquer nos réflexions, comment on va créer le pipeline, RGPD, sécurité. Quels KPI etc.

Une image contenant texte, capture d’écran, Police, nombre

Le contenu généré par l’IA peut être incorrect.

Intégration de données externes : quand on recup les données, on determine si on les stocke sur le cloud, que la BDD soit située en europe ou France. S’assurer que les données soient au bon format, pour ne pas fausser les calculs. Ne pas surcharger le dashboard, faire attention à la charte graphique.

Idée pour le machine learning :

Régression linéaire pour montrer l’augmentation linéaire de la température.

https://api.openweathermap.org/data/3.0/onecall/day\_summary? lat=48.856614&lon=2.3522219&date=2003-08-03&appid=2221e16dbf152832c33fa22db1f090a8

Table Air Quality :

* CITY
  + ID
  + Name
  + Latitude
  + Longitude
  + Location ? (vide sur le test sur Paris)

pip install requests pandas python-dotenv google-cloud-storage