SIMPLON.CO

Certification Développeur Data

Rapport projet chef d'œuvre

Présenté le 19 mai 2021 par

Yoann LE VOGUER

Analyse de la tendance du cyclisme à Nantes Métropole entre 2014 et 2021

Table des matières

Introduction 2

1. Analyse de la demande 3

   a) Objectifs et enjeux 3

   b) État de l'art 3

   c) Périmètre du projet 4

   d) Source de données utilisées 4

   e) Règlementation sur l'utilisation des données 6

   f) Schéma fonctionnel 6

   g) Outils technologiques utilisés 7

   h) Description de l'architecture retenue 8

2. Mise en œuvre du projet 9

   a) Planification 9

   b) Nettoyage des données 10

      i. Table Station 10

      ii. Table Mesure 11

      iii. Table Type 14

   c) Développement de la base de données 15

      i. Création base de données et tables 15

      ii. Insertion des données 16

      iii. Vérification des doublons 16

   d) Création du tableau de bord 17

   e) Visualisation du tableau de bord 17

3. Analyse, bilan du projet et améliorations 19

   a) Analyse de la base de données 19

   b) Suggestions pour la suite 21

Conclusion 22

Bibliographie 23

Ressources 23

Introduction

Dans un contexte général de changement climatique, d'augmentation du coût des énergies pour se déplacer et d'embouteillages fréquents dans les grandes villes, la mobilité est devenue l'une des thématiques indispensables et un vrai sujet politique au sein des agglomérations face à la prise de conscience citoyenne. Le vélo constitue aujourd'hui l'une des solutions principales pour répondre à toutes ces problématiques à la fois.

Aujourd'hui, la pratique du vélo est d'ampleur très différente selon les pays. Elle est notamment très fréquente dans les pays en développement. Avec l'urbanisation des villes, couplée avec les problématiques citées précédemment, son utilisation a tendance à augmenter également dans les pays développés. A l'échelle de la France, sa croissance s'est concrétisée en 2019 par une augmentation de +5 % de passages par rapport à 2018 et +19 % par rapport à 2013. Ces indicateurs sont connus grâce à un nombre croissant de compteurs en France, qui permettent de faire ressortir les tendances et d'analyser les comportements des citoyens et leur utilisation du vélo.

Nantes Métropole ne fait pas exception à cette tendance et effectue depuis 2006 des comptages de vélo sur certaines voies cyclables. L'association Place au Vélo, qui a vu le jour en 1991, à l'époque pour obtenir le droit de passage des vélos sur le Pont de Cheviré, et dont objectif principal est de promouvoir l'utilisation du vélo comme moyen de transport au quotidien, réalise des comptages ponctuels depuis 1998. Si les comptages étaient réalisés dans 20 lieux différents depuis 2014, ce sont désormais une cinquantaine de stations au total qui comptabilisent les vélos dans l'agglomération. Ces données, produites par la collectivité, sont disponibles depuis 2014 et ont vocation à être ouvertes sur le portail *data.nantesmetropole.fr*.

Mais ces données peuvent malheureusement comporter des erreurs, à cause de dysfonctionnements tels que des pannes des capteurs ou des perturbations sur un axe de circulation, qui peuvent altérer la fiabilité et l'analyse de ces données. Un algorithme a alors été mis en place afin de détecter automatiquement ces anomalies.

A partir des données considérées comme fiables, Nantes Métropole souhaite aujourd'hui connaître l'évolution de l'utilisation du vélo dans la ville.

1. Analyse de la demande

a) Objectifs et enjeux

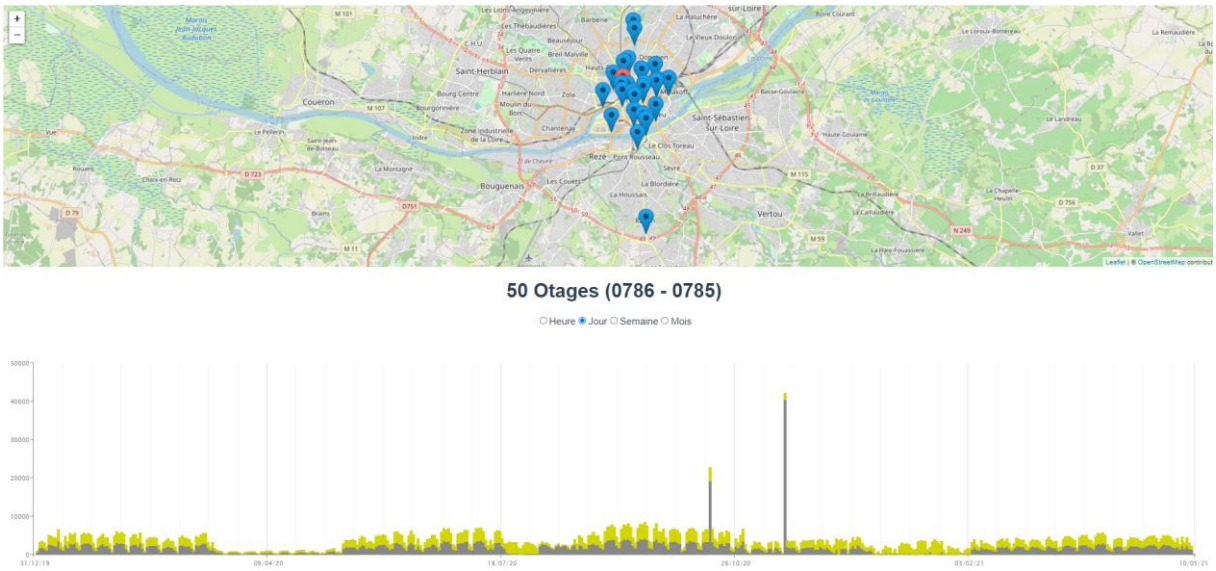
Nantes Métropole souhaite connaître les tendances et évolutions de la pratique du vélo dans la ville ces dernières années, et répondre notamment aux questions suivantes :

- L'utilisation du vélo a-t-elle tendance à augmenter ou à diminuer ?
- L'utilisation du vélo se fait-elle principalement dans le centre-ville ou en périphérie ?
- Le vélo est-il surtout utilisé en semaine ou le weekend ?
- Le confinement a-t-il eu un impact sur la pratique du vélo ?
- Quelles sont les pistes cyclables les plus fréquentées ?
- Quel type de vélos est le plus en expansion ?

b) État de l'art

Concernant l'utilisation des données de comptages vélo de Nantes Métropole, peu d'analyse et d'information sont disponibles pour le grand public. Le portail open data de Nantes Métropole (https://data.nantesmetropole.fr) permet de réaliser quelques analyses graphiques sur les grandes tendances des comptages à travers le temps mais en permet pas d'analyses plus poussées qui permettraient de retirer des conclusions précises.

Le site https://jabby-techs.gitlab.io/comptages-velo/#/ (voir capture d'écran ci-dessous) est un tableau de bord qui permet, en cliquant sur un nom de station, de visualiser l'historique des comptages grâce à un diagramme en barres. Ces informations sont disponibles sur une granularité horaire, journalière, hebdomadaire ou annuelle. De même que le portail de Nantes Métropole, il ne permet pas d'aller plus loin dans l'analyse des données, mais permet uniquement de voir les grandes tendances d'utilisation du vélo pour une station donnée.



En revanche, des travaux ont été réalisés et sont disponibles, tel que le rapport **Analyse des données de fréquentation cyclable 2019**, réalisé par le réseau de collectivités Vélo & Territoires, qui tire des tendances du cyclisme à partir de 903 compteurs à travers le pays durant la dernière décennie.

Pour se rapprocher de ce projet, les apprenants Benoit Gascou, Cynthia Laboureau et Joséphine Vaton ont réalisé dans le cadre de leur formation Data Analyst une étude nommée **Le vélo à Paris, data analyse du trafic cycliste, de septembre 2019 à décembre 2020**. Le travail réalisé prend la forme d'un tableau de bord, associant graphiques et analyses, et se base sur les données Comptage Vélos du portail open data de la Ville de Paris, et de la Base de données accidents de la circulation routières, du portail data du gouvernement. Des conclusions permettent de quantifier l'évolution du trafic cycliste et d'étudier la corrélation avec les accidents. Le tableau de bord réalisé est disponible sur la page suivante : https://share.streamlit.io/benoitgascou/demo_pycycle/main/demo_streamlit.py.

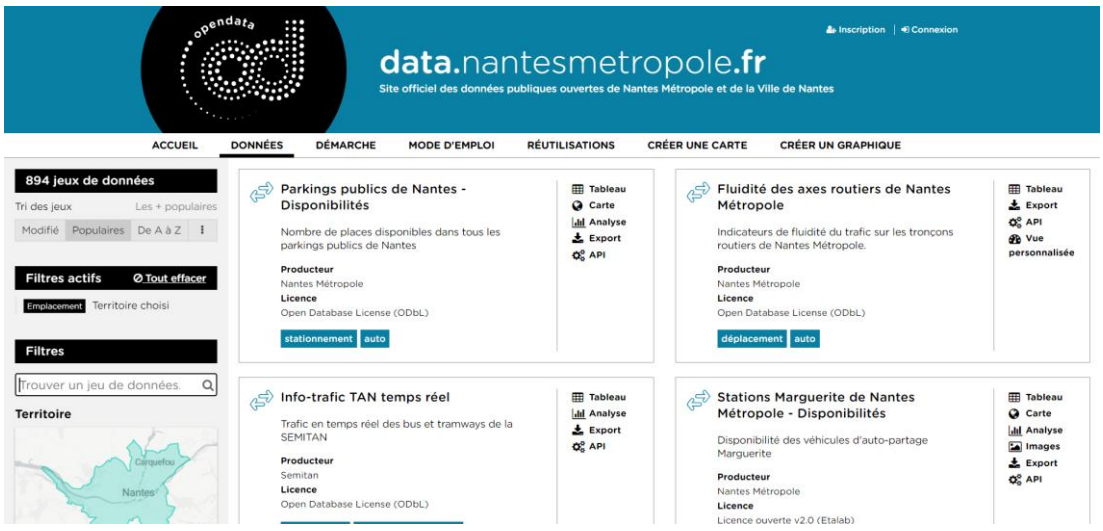
c) Périmètre du projet

Création d'un tableau de bord qui va interroger la base de données et permettre de retirer les tendances du vélo sur Nantes Métropole, en respectant le cahier des charges (voir en annexe à la fin de ce rapport).

d) Source de données utilisées

6 des jeux de données utilisés proviennent du portail de Nantes Métropole : *https://data.nantesmetropole.fr* .

- 4 concernent les données des mesures
- 2 concernent les données des stations



Comptages vélo de Nantes Métropole

- **Producteur :** Nantes Métropole
- **Licence :** Open Database License (ODbL)
- **Description :** Comptages vélo à Nantes Métropole à partir de 2020. Ce jeu de données fournit, pour chaque boucle de comptage fonctionnelle de la Métropole de Nantes, le nombre de passages décomptés heure par heure.

Comptages vélo de Nantes Métropole - Années 2014 à 2019

- **Producteur :** Nantes Métropole
- **Licence :** Open Database License (ODbL)
- **Description :** Historique des comptages vélo : données brutes, anomalies détectées et valeurs ajustées. Ce jeu fournit, pour 20 boucles de comptages dont les données collectées entre 2014 et 2019 sont exploitables, le nombre de passages enregistrés chaque jour. Cela correspond à trois variables : date, compteur et donnée relevée.

Comptages vélo de Nantes Métropole - Boucles de comptage

- **Producteur :** Nantes Métropole
- **Licence :** Licence ouverte v2.0 (Etalab)
- **Description :** Boucles de comptage vélo de Nantes Métropole. Le jeu liste les boucles de comptage vélo en service de Nantes Métropole, et présente leur géolocalisation.

Comptages des VAE et Bicloo par Place au vélo à Nantes - Années 2010 à 2020

- **Producteur :** Place au vélo
- **Licence :** Licence ouverte v2.0 (Etalab)
- **Description :** Comptages des VAE et des Bicloo à Nantes par Place au vélo de 2010 à 2020. Issues des campagnes annuelles de comptages de Place au vélo, les informations sur le nombre de VAE (Vélos à Assistance Électrique) et de Bicloo (Vélos en Libre Service à Nantes Métropole) sont comptabilisées. Les 2 journées de comptage annuelles sont organisées en 2 sessions d'une heure par jour (à 14h et 17h30), sur 5 sites de la ville de Nantes.

Comptages vélo à Nantes par Place au Vélo - Années 1998 à 2020

- **Producteur :** Place au vélo
- **Licence :** Licence ouverte v2.0 (Etalab)
- **Description :** L'association Place au vélo organise chaque année une campagne de comptage sur la ville de Nantes depuis 1998. Les campagnes de comptage de Place au vélo sont réalisées deux fois par mois.

Lieux de comptage de Place au vélo à Nantes

- **Producteur :** Place au vélo
- **Licence :** Licence ouverte v2.0 (Etalab)
- **Description :** Localisation des lieux de comptage de l'association Place au vélo lors de leurs campagnes annuelles de comptage.

Le jeu de données concernant les données météorologiques provient du portail des Pays de la Loire : *https://data.paysdelaloire.fr/* .

Température quotidienne en Pays de la Loire

- **Producteur :** Région des Pays de la Loire
- **Licence :** Licence Ouverte v2.0 (Etalab)
- **Description :** Ce jeu de données présente les températures minimales, maximales et moyennes quotidiennes (en degré celsius), en région Pays de la Loire, du 1er janvier 2016 à aujourd'hui.

e) Règlementation sur l'utilisation des données

Pour la licence ODbL, vous êtes libres de :

- de partager : copier, distribuer et utiliser la base de données.
- de créer : produire des créations à partir de cette base de données.
- d'adapter : modifier, transformer et construire à partir de cette base de données

Pour la licence ouverte / open licence, vous êtes libres de :

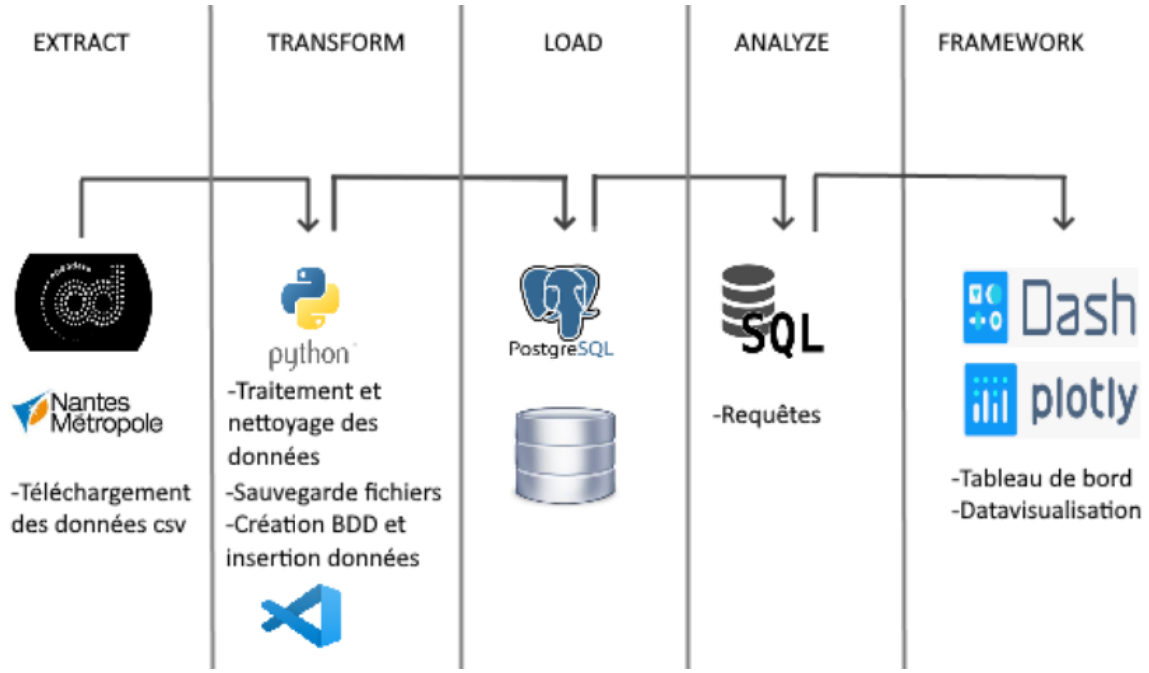
- de reproduire, copier, publier et transmettre « l'information »
- de diffuser et redistribuer « l'information »
- d'adapter, modifier, extraire et transformer à partir de « l'information », notamment pour créer des « informations dérivées »
- d'exploiter « l'information » à titre commercial, par exemple en la combinant avec d'autres « informations », ou en l'incluant dans votre propre produit ou application

Les sources sont en bibliographie pour connaître plus en détail les modalités d'utilisation de ces types de licence.

f) Schéma fonctionnel

Les données d'entrée ont toutes pour format des fichiers .csv, situés dans le répertoire **Data**. Après traitement, nettoyage et mise en forme des données pour ne conserver que celles qui seront intégrées à la base de données, celles-ci sont sauvegardées automatiquement dans le dossier **output**, situé à l'intérieur du répertoire **Data**.

Il est représenté comme ceci :



g) Outils technologiques utilisés

Ci-dessous l'ensemble outils informatiques utilisés pour la mise en œuvre du projet :

- **Gantt Project** : logiciel libre de gestion de projet, ici utilisé pour la planification
- **GitHub/GitLab :** outils de versionning, permettant aux développeurs de stocker et de partager, publiquement ou non, le code qu'ils créent
- **Mocodo :** logiciel de modélisation conceptuelle de données
- **Jupyter :** application web qui permet de créer des notebooks, ici utilisé pour tester les scripts dans un premier temps
- **PostGreSQL :** système de gestion de base de données relationnelle et objet
- **Visual Studio Code :** éditeur de code extensible développé par Microsoft, ici utilisé pour exécuter les scripts Python
- **MongoDB :** système de gestion de base de données orienté documents, ici utilisé pour créer le répertoire de métadonnées
- **DBeaver :** logiciel permettant l'administration et le requêtage de base de données
- **Python :** langage de programmation interprété, multi-paradigme et multiplateformes
- **SQL :** Structured Query Language - langage informatique normalisé servant à exploiter des bases de données relationnelles
- **Dash :** framework Python qui permettant de développer des applications web analytiques

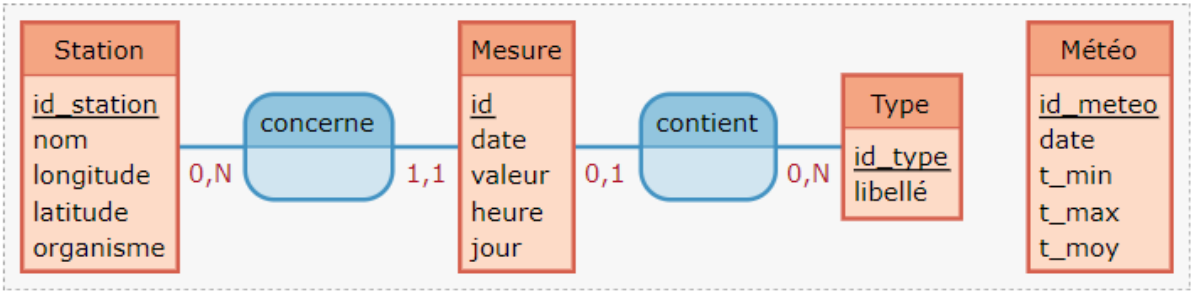
h) Description de l'architecture retenue

L'architecture retenue sera une base de données SQL.

La relation entre la table **Station** et la table **Mesure** sera réalisée grâce à l'attribut *id_station*, clé primaire *id* de la table **Station**. C'est une relation one-to-many.

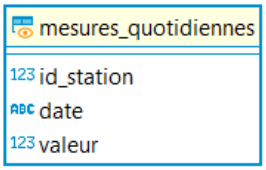
La relation entre la table **Type** et la table **Mesure** sera réalisée grâce à l'attribut *type*, clé primaire *id* de la table **Type**. C'est une relation one-to-many.

La table **Meteo** n'a pas de relation directe avec d'autres tables, bien qu'une jointure de son attribut *date* pourra être relié à l'attribut *date* de la table **Mesure** lors des requêtes SQL. Il n'y a pas de liaison directe car aucun des 2 attributs n'est une clé primaire et qu'une date peut figurer dans la table **Mesure** et non dans la table **Meteo** ou inversement, ce qui posera des problèmes lors de l'insertion des données.



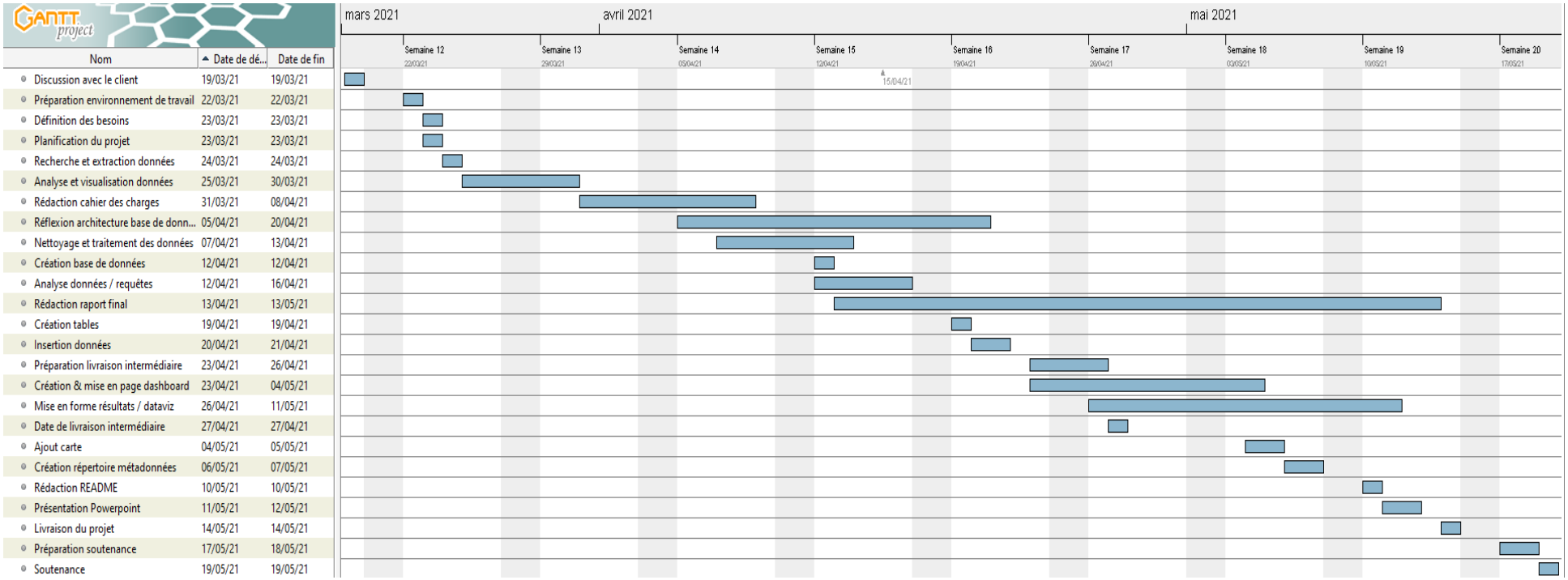
STATION (id_station, nom, longitude, latitude, organisme)
MESURE (id, date, valeur, heure, jour, *id_type*, *id_station*)
TYPE (id_type, libellé)
MÉTÉO (id_meteo, date, t_min, t_max, t_moy)

Une vue **Mesures_quotidiennes** a également été créée. Elle permet, à partir de l'identifiant de la station et de la date, de connaître le nombre de vélos passés par jour, ce qui permet d'avoir une vue d'ensemble plus générale, sans rentrer dans les détails. L'attribut *valeur* a l'avantage de posséder des données homogènes (nombre de vélos par jour et par station), contrairement à l'attribut valeur de la table **Mesure**, qui détaille parfois l'heure ou le type, ce qui simplifiera les requêtes SQL par la suite.



2. Mise en œuvre du projet

a) Planification



Nom	Date de dé...	Date de fin
Discussion avec le client	19/03/21	19/03/21
Préparation environnement de travail	22/03/21	22/03/21
Définition des besoins	23/03/21	23/03/21
Planification du projet	23/03/21	23/03/21
Recherche et extraction données	24/03/21	24/03/21
Analyse et visualisation données	25/03/21	30/03/21
Rédaction cahier des charges	31/03/21	08/04/21
Réflexion architecture base de donn...	05/04/21	20/04/21
Nettoyage et traitement des données	07/04/21	13/04/21
Création base de données	12/04/21	12/04/21
Analyse données / requêtes	12/04/21	16/04/21
Rédaction raport final	13/04/21	13/05/21
Création tables	19/04/21	19/04/21
Insertion données	20/04/21	21/04/21
Préparation livraison intermédiaire	23/04/21	26/04/21
Création & mise en page dashboard	23/04/21	04/05/21
Mise en forme résultats / dataviz	26/04/21	11/05/21
Date de livraison intermédiaire	27/04/21	27/04/21
Ajout carte	04/05/21	05/05/21
Création répertoire métadonnées	06/05/21	07/05/21
Rédaction README	10/05/21	10/05/21
Présentation Powerpoint	11/05/21	12/05/21
Livraison du projet	14/05/21	14/05/21
Préparation soutenance	17/05/21	18/05/21
Soutenance	19/05/21	19/05/21

b) Nettoyage des données

Après collecte des données (voir 1-d : Source de données utilisées),

i. Table Station

Les données proviennent de 2 fichiers différents : les stations appartenant à Nantes Métropole et ceux appartenant à Place au Vélo. La première étape est d'importer les données.

```
lieux1 = pd.read_csv("C:/Users/levog/Simplon/projet-chef-d-oeuvre/Data/Lieux_comptages_velo_nantes_metropole.csv", delimiter=';')
```

```
lieux1
```

	Numéro	Libellé	Observations	Géolocalisation
0	997.0	Van Iseghem vers Sud	NaN	47.2326801845,-1.54881176633
1	996.0	Van Iseghem vers Nord	NaN	47.2326672074,-1.5487986994
2	995.0	Ceineray vers Ouest	NaN	47.2211938621,-1.55203851068
3	994.0	Ceineray vers Est	NaN	47.2211809025,-1.55205574517
4	989.0	Kennedy vers Ouest	NaN	47.2170822199,-1.54476993041
...	...	...	...	...
59	665.0	Bonduelle vers Nord	NaN	47.2116093885,-1.54325998832
60	664.0	Bonduelle vers sud	NaN	47.2115623077,-1.54340196493
61	89.0	Coteaux vers Ouest	Releve manuel	47.2050895879,-1.76738871373
62	NaN	La Chapelle sur Erdre	Releve manuel	47.3150926414,-1.54316810118
63	NaN	Saint Léger les Vignes	Releve manuel	47.1408039248,-1.71322007326

64 rows × 4 columns

```
lieux2 = pd.read_csv("C:/Users/levog/Simplon/projet-chef-d-oeuvre/Data/Lieux_comptages_place_au_velo.csv"
```

```
lieux2
```

	identifiant	Lieu de comptage	Géométrie	geo_point_2d
0	1000	Quai de la Fosse	{"type": "Point", "coordinates": [-1.56418, 47...	47.20977,-1.56418
1	1003	Pont Audibert rive Nord	{"type": "Point", "coordinates": [-1.54995, 47...	47.20893,-1.54995
2	1002	Intersection de boulevards Michelet, Orrion et...	{"type": "Point", "coordinates": [-1.55673, 47...	47.23311,-1.55673
3	1001	Bd Kennedy devant le Chateau	{"type": "Point", "coordinates": [-1.54952, 47...	47.21538,-1.54952
4	1004	Cours des 50 otages à l'intersection de la rue...	{"type": "Point", "coordinates": [-1.5563, 47....	47.2158,-1.5563

Les variables **lieux1** et **lieux2** correspondent à des dataframes lisant les données recueillies dans les fichiers .csv. Ils subiront différentes étapes de traitement et de nettoyage. La 1ère modification de **lieux1** est de splitter la colonne Géolocalisation.

lieux1[['Longitude','Latitude']] = lieux1.Géolocalisation.str.split(",",expand=True)

Les colonnes Observations et Géolocalisation, inutiles pour la suite du projet (Géolocalisation ferait doublon aux 2 colonnes créées), sont supprimées.

lieux1 = lieux1.drop(columns=['Observations', 'Géolocalisation'])

Deux stations ne possèdent pas d'identifiant, nous allons leur en attribuer un : 100 pour 'Saint Léger les Vignes', 101 pour 'La Chapelle sur Erdre'.

lieux1['Numéro'].where(lieux1['Libellé']!='Saint Léger les Vignes','100',inplace=True)

lieux1['Numéro'].where(lieux1['Libellé']!='La Chapelle sur Erdre','101',inplace=True)

On convertir ensuite l'identifiant station (initialement en format texte) en Integer.

lieux1['Numéro'] = lieux1['Numéro'].astype(int)

Puis on ajoute une colonne Organisme qui permettra par la suite de connaître le propriétaire du compteur, ici Nantes Métropole.

lieux1['Organisme'] = 'Nantes Métropole'

Nous renommons ensuite les colonnes d'un nom d'attribut explicite qui nous permettra par la suite de les standardiser avec les futures données à ajouter.

lieux1 = lieux1.rename(columns = {'Numéro': 'Id', 'Libellé': 'Nom'})

Le traitement des données concernant les stations de comptage de Nantes Métropole est maintenant achevé, nous réalisons approximativement les mêmes étapes avec le dataframe **lieux2**, avant de concaténer les 2 dataframes :

```
df_data_stations = pd.concat([lieux1, lieux2])
```

Le résultat final est celui-ci, nous pouvons maintenant l'exporter au format csv :

	Id	Nom	Longitude	Latitude	Organisme
0	997	Van Iseghem vers Sud	47.2326801845	-1.54881176633	Nantes Métropole
1	996	Van Iseghem vers Nord	47.2326672074	-1.5487986994	Nantes Métropole
2	995	Ceineray vers Ouest	47.2211938621	-1.55203851068	Nantes Métropole
3	994	Ceineray vers Est	47.2211809025	-1.55205574517	Nantes Métropole
4	989	Kennedy vers Ouest	47.2170822199	-1.54476993041	Nantes Métropole
...	...	...	...	...	...
0	1000	Quai de la Fosse	47.20977	-1.56418	Place au vélo
1	1003	Pont Audibert rive Nord	47.20893	-1.54995	Place au vélo
2	1002	Intersection de boulevards Michelet, Orrion et...	47.23311	-1.55673	Place au vélo
3	1001	Bd Kennedy devant le Chateau	47.21538	-1.54952	Place au vélo
4	1004	Cours des 50 otages à l'intersection de la rue...	47.2158	-1.5563	Place au vélo

69 rows × 5 columns

ii. Table Mesure

Le traitement de table Mesure a demandé un travail plus complexe car il nécessitait un travail sur 4 fichiers .csv différents et non standardisés, les colonnes et la structure des fichiers étant très hétérogènes. Lorsque les étapes seront similaires pour les 4 fichiers, seul le code du premier fichier sera démontré.

La 1ère étape est d'importer les 4 fichiers sous forme de liste, voici un exemple du fichier comprenant les mesures des compteurs de Nantes Métropole entre 2014 et 2019 :

```
mesures1 = 'C:/Users/levog/Simplon/projet-chef-d-oeuvre/Data/Comptages_nantes_metropole_2014-2019.csv'

data_mesures1 = []

with open(mesures1, newline="",encoding='utf-8') as csvfile:
  reader = csv.DictReader(csvfile, delimiter=';')
  for row in reader:
    data_mesures1.append(row)
```

Pour les listes **data_mesures1** ou **data_mesures2**, une suppression des valeurs qui possèdent des anomalies est effectuée, afin de ne conserver que les données jugées fiables, puis on les insère dans une nouvelle liste.

```
Data_mesures1 = []

for e in data_mesures1:

  if e['Valeur modélisée'] != 'NA' or ['Valeur modélisée'] != '':

    Data_mesures1.append(e)
```

On convertit ensuite les listes en dataframes pour travailler dessus :

```
df_data_mesures1 = pd.DataFrame(Data_mesures1)
```

Voici désormais les rendus de chaque dataframe, très hétérogènes comme nous pouvons le constater, le **df_data_mesures2** possédant comme information supplémentaire une granularité horaire et le **df_data_mesures4** une information par type de vélo :

df_data_mesures1

	Identifiant du compteur	Jour	Nom du compteur	Comptage relevé	Anomalie	Valeur modélisée
0	745	2016-03-24	Calvaire_vers_Ouest	98	NA	154
1	745	2016-03-27	Calvaire_vers_Ouest	13	NA	29
2	745	2016-04-06	Calvaire_vers_Ouest	56	NA	150
3	745	2016-04-11	Calvaire_vers_Ouest	48	NA	127

df_data_mesures2

	Numéro de boucle	Libellé	Jour	00	01	02	03	04	05	06	...	16	17	18	19	20	21	22	23	Probabilité de présence d'anomalies	Jour de la semaine
0	0665	Bonduelle vers Nord	2021-05-01	1	4	2	1	0	4	2	...	39	56	49	40	18	15	8	3		6
1	0680	Stalingrad vers ouest	2021-05-01	4	2	2	3	3	4	3	...	53	41	53	32	19	12	6	6		6
2	0681	Stalingrad vers est	2021-05-01	6	2	0	0	1	2	1	...	41	48	62	41	18	7	8	10		6

df_data_mesures3

	Années de comptage	Identifiant du lieu de comptage	Nom du lieu de comptage	Date complète	Date	Heure	Jour de la semaine	Comptage relevé
0	2018	1000	Quai de la Fosse vers l'ouest	2018-09-27T17:30:00+02:00	2018-09-27	17:30:00	jeudi	315
1	2016	1000	Quai de la Fosse vers l'ouest	2016-09-29T14:00:00+02:00	2016-09-29	14:00:00	jeudi	94
2	2015	1000	Quai de la Fosse vers l'ouest	2015-09-17T14:00:00+02:00	2015-09-17	14:00:00	jeudi	71

df_data_mesures4

	Années de comptage	Identifiant	Nom du lieu de comptage	Date complète	Date	Heure	Jour de la semaine	VAE	bicloo	Autres vélos	Total
0	2011	1003	Pont Audibert rive Nord	2011-09-29T14:00:00+02:00	2011-09-29	14:00:00	jeudi	0	17	108	125
1	2010	1000	Quai de la Fosse	2010-09-22T17:30:00+02:00	2010-09-22	17:30:00	mercredi	0	13	204	217
2	2013	1002	Intersection de boulevards Michelet, Orrion et...	2013-10-01T14:00:00+02:00	2013-10-01	14:00:00	mardi	1	2	41	44

On attribue de nouveau un identifiant aux deux stations qui n'en possédaient pas, en reprenant les mêmes numéros que pour la table Station.

```
df_data_mesures1['Identifiant du compteur'].where(df_data_mesures1['Nom du compteur']!='Saint Léger les Vignes','100',inplace=True)

df_data_mesures1['Identifiant du compteur'].where(df_data_mesures1['Nom du compteur']!='La Chapelle sur Erdre','101',inplace=True)
```

On supprime maintenant les enregistrements dont aucun comptage n'a été relevé.

```
df_data_mesures1.drop(df_data_mesures1.loc[df_data_mesures1['Comptage_relevé']=='NA'].index, inplace=True)
```

Les colonnes qui ne nous serviront pas pour la suite ou qui représentent un doublon sont supprimés, nous avons alors le même nombre de colonnes pour chaque dataframe.

```
df_data_mesures1 = df_data_mesures1.drop(columns=['Nom du compteur', 'Comptage relevé', 'Anomalie'])

df_data_mesures2 = df_data_mesures2.drop(columns=["Probabilité de présence d'anomalies", "Jour de la semaine", "Libellé"])

df_data_mesures3 = df_data_mesures3.drop(columns=["Années de comptage", "Nom du lieu de comptage", "Date complète", "Jour de la semaine"])

df_data_mesures4 = df_data_mesures4.drop(columns=["Années de comptage", "Nom du lieu de comptage", "Date complète", "Jour de la semaine", "Total"])
```

Nous avons vu dans les captures d'écran ci-dessus que les informations par horaire et par type sont représentées en autant de colonnes que de précision d'information (heure et type). La fonction *melt* va nous permettre de réaliser un pivot de ces informations, c'est-à-dire qu'au lieu d'y avoir autant de colonnes en une seule ligne, un enregistrement sera créé par information (ex : une ligne par jour, heure et station).

```
df_data_mesures2 = df_data_mesures2.melt(id_vars = ['Numéro de boucle', 'Jour'], var_name = 'heure', value_name = 'valeur')

df_data_mesures4 = df_data_mesures4.melt(id_vars = ['Identifiant', 'Date', 'Heure'], var_name = 'type', value_name = 'total')
```

Etant donné qu'une table **Type** sera créé par la suite (explication dans la partie suivante), nous remplaçons les libellés des types de vélo par un identifiant.

```
df_data_mesures4 = df_data_mesures4.replace(['VAE', 'bicloo', 'Autres vélos'], ['1', '2', '3'])
```

Toutes les colonnes des différents dataframes sont ensuite renommés afin d'être standardisés les uns avec les autres.

```
df_data_mesures1 = df_data_mesures1.rename(columns = {'Identifiant du compteur': 'Id_station', 'Jour': 'Date', 'Valeur modélisée': 'Valeur'})

df_data_mesures2 = df_data_mesures2.rename(columns = {'Numéro de boucle': 'Id_station', 'Jour': 'Date', 'heure': 'Heure', 'valeur': 'Valeur'})

df_data_mesures3 = df_data_mesures3.rename(columns = {'Identifiant du lieu de comptage': 'Id_station','Comptage relevé': 'Valeur'})

df_data_mesures4 = df_data_mesures4.rename(columns = {'Identifiant': 'Id_station', 'type': 'Type', 'total': 'Valeur'})
```

Les colonnes Heure sont ensuite modifiées, d'une valeur Integer *13* à un format *13:00:00* afin de pouvoir par la suite les exploiter en temps que Datetime.

df_data_mesures2['Heure'] = df_data_mesures2['Heure'].replace(['00'],'00:00:00')

Nous pouvons à présent concaténer les 4 dataframes.

Le format des colonnes *Date* et *Id_station* peuvent être passés en numérique.

df_data_mesures['Date'] = pd.to_datetime(df_data_mesures['Date'])

df_data_mesures['Id_station'] = pd.to_numeric(df_data_mesures['Id_station'])

Puis une colonne *Jour* est ajoutée afin de connaître le jour de la semaine de chaque date grâce à la fonction *dt.day_name*.

df_data_mesures['Jour'] = df_data_mesures['Date'].dt.day_name()

La dernière étape sera de rechercher la liste des stations enregistrées dans le fichier consacré aux stations que nous avons exportés tout à l'heure. Ainsi, les quelques identifiants de stations inconnus ne seront pas enregistrés dans le fichier de sortie.

liste_stations = pd.read_csv("C:/Users/Public/Data/Stations_velos.csv", delimiter=',')

liste = liste_stations['Id']

liste_stations = list(liste)

df_data_mesures = df_data_mesures[df_data_mesures.Id_station.isin(liste_stations)]

Les données contenues dans **df_data_mesures** peuvent à présent être exportées.

	Id_station	Date	Valeur	Heure	Type	Jour
0	745	2016-03-24	154	NaN	NaN	Thursday
1	745	2016-03-27	29	NaN	NaN	Sunday
2	745	2016-04-06	150	NaN	NaN	Wednesday
3	745	2016-04-11	127	NaN	NaN	Monday
4	745	2016-04-13	148	NaN	NaN	Wednesday
...	...	...	...	...	...	...
655	1004	2019-09-26	491	17:30:00	3	Thursday
656	1000	2015-09-24	105	14:00:00	3	Thursday
657	1003	2015-09-24	145	14:00:00	3	Thursday
658	1004	2015-09-24	234	14:00:00	3	Thursday
659	1001	2014-09-18	279	17:30:00	3	Thursday

634925 rows × 6 columns

iii. Table Type

Une table **Type** est créée, afin de l'associer à la colonne Type de la table **Mesure**. Ce système permettra de mettre un simple identifiant si des ajouts manuels devaient être fait dans la table **Mesure**. Ainsi, un identifiant est plus simple à rentrer qu'un libellé et permet entre autres d'éviter les fautes de frappe ou les saisies différentes.

Les identifiants correspondent aux mêmes libellées que ceux entrés dans la table **Mesure** :

1. VAE
2. Bicloo
3. Autres vélos

Voici le code de création du dataframe puis de l'export des données en fichier .csv.

```
types = pd.DataFrame({'id': ['1', '2', '3'],
                      'type': ['VAE', 'Bicloo', 'Autres vélos']})
types.to_csv('C:/Users/levog/Simplon/projet-chef-d-oeuvre/Data/output/Types.csv', index=False)
```

c) Développement de la base de données

i. Création base de données et tables

Dans un 1er temps, la base de données est créée sur PostGreSQL grâce à ce script :

```
sql = '''CREATE database vélos_nantesmetropole''';
cursor.execute(sql)
```

Après connexion à cette base de données, les tables sont créées grâce au script suivant. Tout d'abord, les tables **Station**, **Meteo** et **Type** dont les attributs ne sont dépendants d'aucune autre table. Le *SERIAL* de la clé primaire *id* de la table **Meteo** correspond à la création automatiquement d'un index qui correspondra à l'identifiant de l'enregistrement. La table **Mesure** contient 2 clés étrangères : *id_station* et *type*.

```
create_tables_sql = '''CREATE TABLE IF NOT EXISTS Station (
  id INTEGER PRIMARY KEY,
  nom VARCHAR,
  longitude FLOAT,
  latitude FLOAT,
  organisme VARCHAR);

  CREATE TABLE IF NOT EXISTS Meteo (
  id SERIAL PRIMARY KEY,
  date VARCHAR,
  t_min FLOAT,
  t_max FLOAT,
  t_moy FLOAT);

  CREATE TABLE IF NOT EXISTS Type (
  id INTEGER PRIMARY KEY,
  libellé VARCHAR);
```

```
  CREATE TABLE IF NOT EXISTS Mesure (
  id SERIAL PRIMARY KEY,
  date VARCHAR,
  id_station INTEGER,
  valeur FLOAT,
  heure TIME,
  type INTEGER,
  jour VARCHAR,
  FOREIGN KEY (id_station) REFERENCES Station(id),
  FOREIGN KEY (type) REFERENCES Type(id));''';

cursor.execute(create_tables_sql)
```

ii. Insertion des données

Nous commençons par insérer les données de la table **Station**. La méthode étant identique pour les 4 tables, nous ne verrons que le script de celle-ci, à partir du fichier .csv contenant les données que nous avons exportées après l'étape de nettoyage.

```
sql_inserer_stations = """
  COPY Station (id, Nom, Longitude, Latitude, Organisme)
  FROM 'C:/Users/Public/Data/Stations_velos.csv'
  WITH csv header ENCODING 'UTF-8' DELIMITER ',' QUOTE '"';
"""

cursor.execute(sql_inserer_stations)
```

Création de la vue **Mesures_quotidiennes**.

```
sql_creation_mesures_jours = '''CREATE VIEW Mesures_quotidiennes (
    Id_station,
    Date,
    Valeur)
  AS
        SELECT id_station,
               date,
               sum(valeur)
        FROM Mesure
        GROUP BY id_station, date'''

cursor.execute(sql_creation_mesures_jours)
```

iii. Vérification des doublons

Il est possible de vérifier si des doublons existent qu'il ne devrait pas y en avoir lieu. Deux simples requêtes SQL permettent de vérifier, pour la première, si deux stations différentes ne portent pas le même nom, et pour la deuxième, s'il y a bien un enregistrement par date et par identifiant de station dans la vue **Mesures_quotidiennes**.

Les requêtes sont les suivantes :

```
SELECT   COUNT(*) AS nbr_doublon, nom
FROM     station
GROUP BY nom
HAVING   COUNT(*) > 1
```

```
SELECT   COUNT(*) AS nbr_doublon, date,
id_station
FROM     mesures_quotidiennes
GROUP BY date, id_station
HAVING   COUNT(*) > 1
```

Dans les 2 cas, aucun enregistrement n'est sorti, ce qui démontre 0 doublon.

d) Création du tableau de bord

La gestion et l'exécution des scripts Python permettront la création du tableau de bord. Voici les différents scripts dans leur ordre d'utilisation et leurs différents rôles :

- **Meta.py** : Création du répertoire de métadonnées des fichiers sources.
- **Nettoyage_stations.py :** Traitement et export des données sur les stations.
- **Création_fichier_types.py :** Création des données sur les types de vélo.
- **Nettoyage_mesures.py :** Traitement et export des données sur les mesures.
- **Nettoyage_meteo.py :** Traitement et export des données météorologiques.
- **Carto.py :** Création et export de la cartographie des stations.
- **Create_bdd.py :** Création de la BDD et des tables, insertion des données.
- **BackupBDD.py** : Sauvegarde des données de la BDD en l'état
- **Data.py :** Stockage des requêtes et des données à partir de la BDD.
- **App.py :** Fichier de création et présentation du tableau de bord.

e) Visualisation du tableau de bord

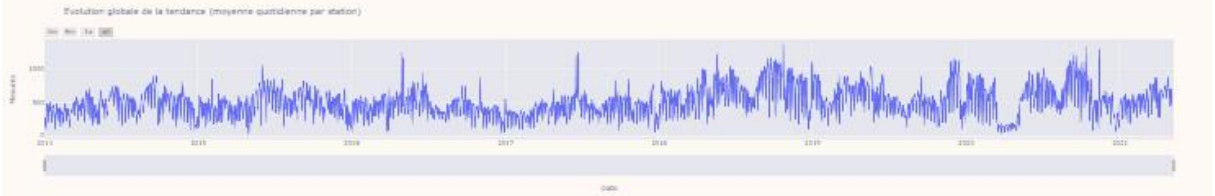
Le tableau de bord final contient dans l'ordre une photo de couverture, un titre, une brève introduction, une cartographie représentant les différentes stations de comptage, ainsi que différents graphiques, séparés en mini-parties, et accompagnés d'une brève explication pour analyser les données produites et répondre aux questions que se posait le client. Une conclusion est disponible à la fin du tableau de bord pour résumer les différentes analyses produites et les croiser afin d'en tirer des enseignements.

Un extrait de ce tableau de bord est disponible à la page suivante.

Analyse de la tendance du cyclisme à Nantes Métropole

Ce tableau de bord décrit les principaux enseignements tirés de l'exploitation de la base de données 'vélos_nantesmetropole.db', qui analyse l'évolution du cyclisme dans la métropole de Nantes à partir des données des stations de comptage placées à différents lieux de la ville. Ce travail a été fourni par Yoann Le Voguer dans le cadre du projet chef d'oeuvre pour la certification Développeur Data à l'organisme Simplon.co Grand Ouest. Nous nous contenterons de commencer dans un premier temps de façon brute les données sous chaque graphique, puis d'en tirer des conclusions à la fin.

I. Evolution globale du cyclisme

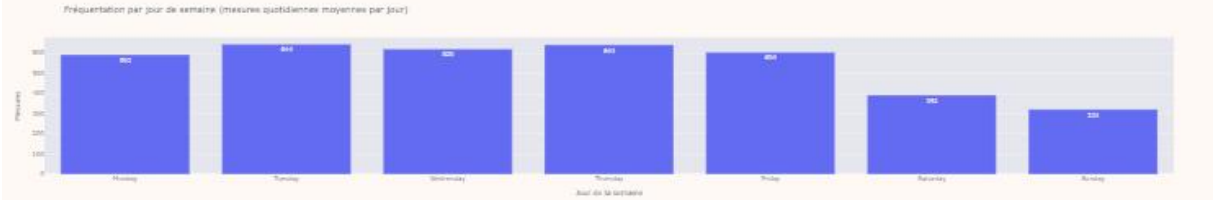


Comme nous pouvons le constater, la fréquence d'utilisation du vélo possède des cycles chaque année, avec une légère augmentation continue entre janvier et juin, avant une diminution en été (juillet-août) avant de regartir à la hausse jusqu'en octobre puis de redimiuer. Globalement on peut donc constater que l'utilisation du vélo est la plus élevée hors vacances scolaires et lors des journées chaudes. De 2014 à 2019, les mesures avaient tendance a augmenté de façon régulière chaque année.

II. Fréquence par station

Ce diagramme en barres nous indique le nombre moyen de vélos comptabilisés par stations. Sans grande surprise, nous nous apercevons que les pistes les plus fréquentées se situent au centre de la ville, notamment 50 Otages.

III. Fréquence par jour de la semaine



Comme nous pouvons le constater, la fréquence d'utilisation du vélo possède des cycles chaque année, avec une légère augmentation continue entre janvier et juin, avant une diminution en été (juillet-août) avant de regartir à la hausse jusqu'en octobre puis de redimiuer. Globalement on peut donc constater que l'utilisation du vélo est la plus élevée hors vacances scolaires et lors des journées chaudes. De 2014 à 2019, les mesures avaient tendance a augmenté de façon régulière chaque année.

IV. Fréquence par type de vélo

Nous possédons grâce à ces graphiques 2 types d'informations différentes. Tout d'abord, le nombre de vélos comptés sur des sessions d'une heure par jour est systématiquement plus élevé à 17h30 qu'à 14h, du double environ. Ensuite, nous constatons une importante augmentation du nombre de vélos entre 2019 et 2021, quelque soit le type de vélo. Les vélos classiques de particuliers constituent la grande majorité des mesures observés, mais si l'on observe dans un même temps la légère augmentation du nombre de Bicloo (vélos libre-service), le nombre de VLE (vélo à assistance électrique) a explosé en une décennie.

3. Analyse, bilan du projet et améliorations

a) Analyse de la base de données

Grâce à des requêtes SQL, nous allons maintenant répondre au maximum aux questions posées par le client. Nous allons ici illustrer quelques requêtes comme exemples, l'ensemble des résultats étant disponibles sur le tableau de bord.

Fréquence par station

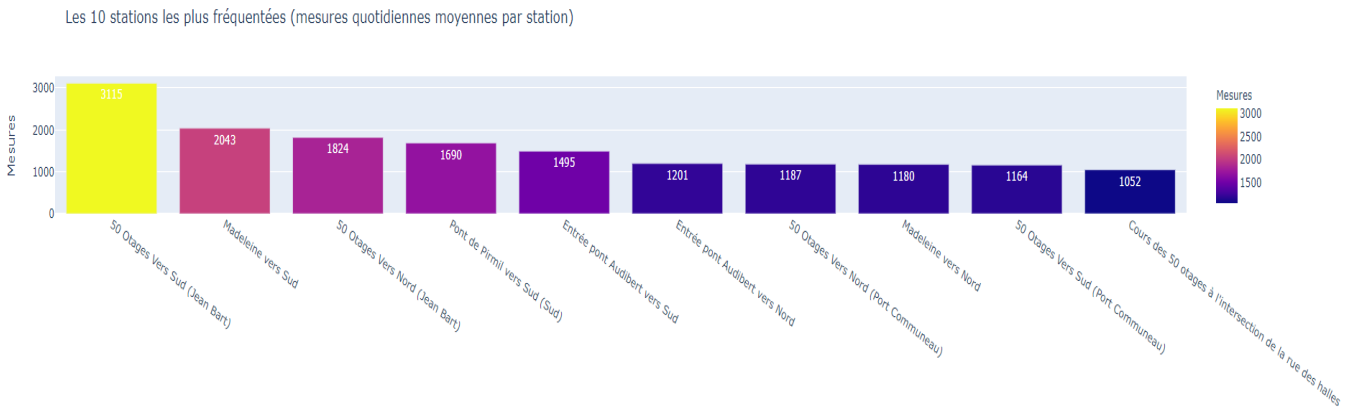
Ce premier exemple, grâce à la jointure entre la vue **Mesures_quotidiennes** et la table **Station**, permet de connaitre les 10 stations dont la moyenne journalière de vélos comptabilisés est la plus élevée.

```
SELECT nom, CAST(ROUND(AVG(CAST(valeur AS DECIMAL))) AS INTEGER) AS moyenne, id
FROM station
INNER JOIN mesures_quotidiennes ON station.id=mesures_quotidiennes.id_station
GROUP BY station.id
ORDER BY moyenne desc
LIMIT 10
```

Voici les résultats de la requête :

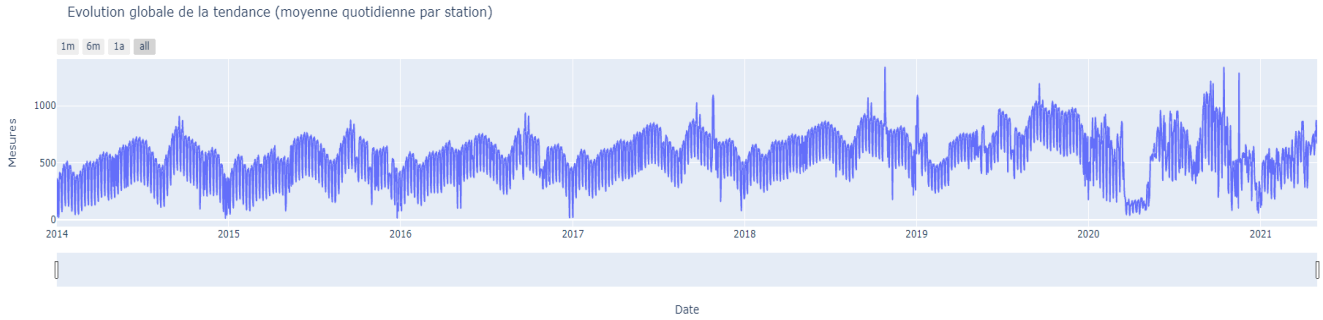
	nom	moyenne	id
1	50 Otages Vers Sud (Jean Bart)	3 115	785
2	Madeleine vers Sud	2 043	881
3	50 Otages Vers Nord (Jean Bart)	1 824	786
4	Pont de Pirmil vers Sud (Sud)	1 690	890
5	Entrée pont Audibert vers Sud	1 495	847
6	Entrée pont Audibert vers Nord	1 201	667
7	50 Otages Vers Nord (Port Communeau)	1 187	788
8	Madeleine vers Nord	1 180	880
9	50 Otages Vers Sud (Port Communeau)	1 164	787
10	Cours des 50 otages à l'intersection de la rue des halles	1 052	1 004

Puis la visualisation réalisée par un graphique en barres avec la librairie **Pyplot** :



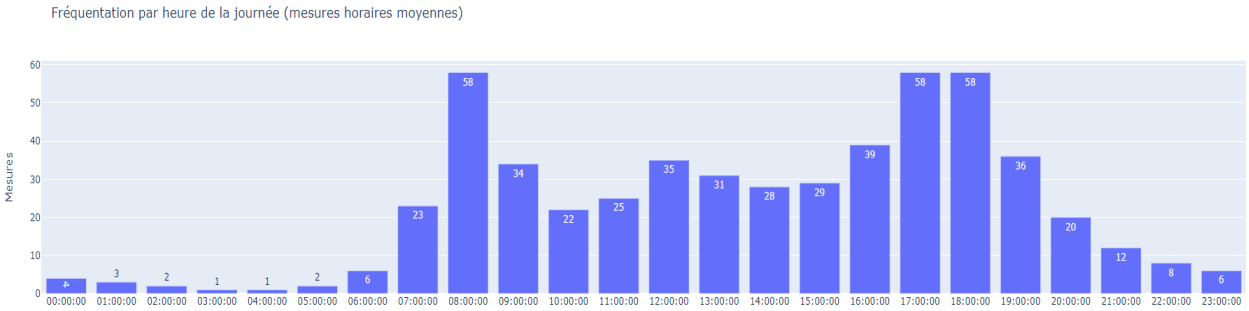
Ce diagramme en barres nous indique le nombre moyen de vélos comptabilisés par stations. Sans grande surprise, nous nous apercevons que les pistes les plus fréquentées se situent au centre de la ville, notamment 50 Otages.

Evolution globale du cyclisme



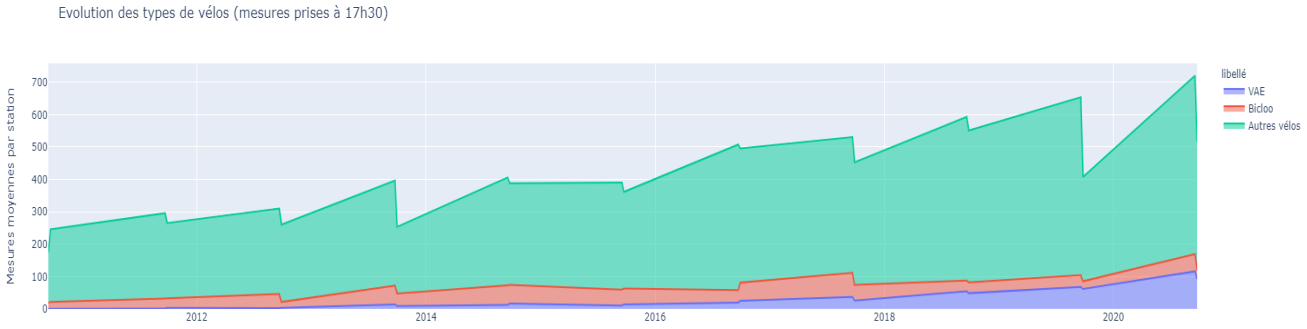
Comme nous pouvons le constater, la fréquence d'utilisation du vélo possède des cycles chaque année, avec une légère augmentation continue entre janvier et juin, avant une diminution en été (juillet-août) avant de repartir à la hausse jusqu'en octobre puis de rediminuer. Globalement on peut donc constater que l'utilisation du vélo est la plus élevée hors vacances scolaires et lors des journées chaudes. De 2014 à 2019, les mesures avaient tendance a augmenté de façon régulière chaque année.

Fréquence par heure de la journée



Ce graphique montre l'évolution du nombre de vélos au fur et à mesure de la journée grâce à une moyenne horaire. On constate un premier pic entre 8h et 9h, ainsi qu'un second entre 17h et 19h. Un léger regain est observé à l'heure de midi. L'utilisation correspondant aux horaires de travail/étude classiques, nous pouvons donc supposer que le vélo est principalement utilisé pour des raisons professionnelles.

Fréquence par type de vélo



Nous possédons grâce à ce graphique le nombre et le type de vélos comptabilisés à 17h30.

Tout d'abord, le nombre de vélos comptés sur des sessions d'une heure par jour est systématiquement plus élevé à 17h30 qu'à 14h (voir le second graphique sur le tableau de bord), du double environ. Ensuite, nous constatons une importante augmentation du nombre de vélos entre 2010 et 2021, quelque soit le type de vélo. Les vélos classiques de particuliers constituent la grande majorité des mesures observés, mais si l'on observe dans un même temps la légère augmentation du nombre de Bicloo (vélos libre-service), le nombre de VAE (vélo à assistance électrique) a explosé en une décennie.

b) Suggestions pour la suite

Une amélioration possible pour la suite de ce projet sera de récupérer les données source en API, cela permettrait de ne pas avoir besoin de télécharger les fichiers sur le portail de Nantes Métropole, les données s'actualiseraient directement à l'exécution du script app.py. Rentre les graphiques interactifs, par exemple pour choisir de connaître les données d'une station en particulier, aurait également pu apporter une plus-value.

Une autre amélioration aurait été de ne travailler qu'avec des stations similaires avant et après 2020. En effet, de nouvelles stations ayant été mises en place après le 1 janvier 2020, les moyennes des données pré et post-2020 ont été comparées alors que les stations ne sont pas spécialement situées dans les mêmes zones de la ville. Par exemple, de nouvelles stations ont pu être installées en périphérie, ce qui ferait potentiellement diminuer la moyenne globale par rapport à une situation où des stations ne sont présentes qu'en centre-ville. Cela ne devrait pas modifier les grandes tendances, mais fait légèrement perdre en précision.

Enfin, la possibilité de modéliser l'ensemble des données, qu'elles soient fiables ou non, pour les rapprocher de ce qui serait la réalité. L'algorithme étant écrit en langage R, et ne connaissant pas ce langage, seules les données jugées fiables ont été retenues puis analysées, mais ce détecteur d'anomalies a probablement donné des faux positifs (fiables) par endroit.

Conclusion

L'exploitation de la base de données a permis de retirer des enseignements sur l'utilisation du vélo à Nantes. Cette tendance est globalement à la hausse sur la dernière décennie, malgré une diminution durant l'année 2020, très marquée durant les confinements, et qui est imputable à la situation sanitaire, donc qui ne laisse pas présager d'une baisse du vélo dans le temps.

Nous en savons également davantage sur les habitudes, et par conséquent les motivations des cyclistes. Deux des analyses que nous avons retirées sont la fréquentation dans l'année et par heure. Cette première information nous apprend que le vélo est plus fréquemment utilisé au printemps et à l'automne et connaît une diminution en été, la seconde nous montre que les pistes cyclables sont principalement fréquentées aux alentours de 8h à 9h, puis de 16h à 18h, avec un léger regain à l'heure de midi. En croisant ces analyses, on peut supposer que le vélo est davantage utilisé pour réaliser le trajet domicile-travail que pour les loisirs. La diminution en hiver peut, elle, être expliquée par les conditions météorologiques plus froides et moins adaptées à la pratique du vélo.

Les stations situées en centre-ville sont, sans grande surprise, plus fréquentées que celle en périphérie. Plusieurs hypothèses peuvent expliquer ce phénomène : plus forte densité de population, plus d'entreprises et d'établissements scolaires présents, meilleur aménagement au niveau des pistes cyclables…

Pour toutes les conclusions qui ont été tirées de ces analyses, si les chiffres permettent de tirer des enseignements, leurs explications restent hypothétiques et il faudrait dépasser les données chiffrées, autrement dit quantitatives, qui est le seul type de données que nous possédons et que nous analysons dans ce type de projet, pour extrapoler et expliquer de façon plus qualitative toutes ces données, en tenant en compte d'autres paramètres.

C'est ce qui a été fait en tentant d'observer le lien entre utilisation du vélo et météo, ce qui a permis de se rendre compte qu'il n'y avait pas de corrélation entre ces deux données, et même une corrélation n'aurait pas permis de mettre en avant une causalité sur la seule base de ces éléments. D'autres paramètres pourraient être pris en compte, tels que la politique de mobilité et d'aménagement cyclable dans la métropole, les jours de grève, les vacances scolaires, les évènements particuliers… tout en pondérant ces données, ce qui démontre qu'il reste potentiellement du travail pour approfondir le sujet !

Bibliographie

Benoit Gascou, C. L. (2020, Novembre). *Le vélo à Paris, data analyse du trafic cyclistede septembre 2019 à décembre 2020*. Récupéré sur https://share.streamlit.io/benoitgascou/demo_pycycle/main/demo_streamlit.py

Etalab. (2017, Avril). *Licence ouverte v2.0 - Open data.* Récupéré sur https://www.etalab.gouv.fr/wp-content/uploads/2017/04/ETALAB-Licence-Ouverte-v2.0.pdf

Kyaagba, S. (2018, Juillet 26). *Integrating Folium with Dash*. Récupéré sur https://medium.com/@shachiakyaagba_41915/integrating-folium-with-dash-5338604e7c56

Métropole, Nantes. (s.d.). *Mode d'emploi - Open Data Nantes Métropole*. Récupéré sur https://data.nantesmetropole.fr/pages/mode-emploi/

Rédaction EP&S. (2015, Avril-mai). Revue EP&S. *Se déplacer en vélo, un enjeu pour mieux vivre en ville*.

Vélos & Territoires. (Mai 2020). *Analyse des données de fréquentation cyclable 2019.*

Ressources

https://plotly.com/

https://openclassrooms.com/forum/

https://forum.clubic.com/

https://forums.commentcamarche.net/forum/

https://stackoverflow.com/

https://python.developpez.com/