

Rapport du Projet : Visualisation en Temps Réel des Stations de Vélos

Contexte du Projet

Ce projet vise à développer une application web permettant de visualiser en temps réel les informations des stations de vélos en libre-service dans une ville (exemple : Lille).

L'application exploite les données fournies via une API REST publique, les affiche sur une carte interactive à l'aide de LeafletJS et fournit des détails tels que :

- Le nom de la station
- La commune
- Le nombre de vélos disponibles
- Le nombre de places disponibles

Objectifs

1. Collecter les données des stations via une API REST.
2. Stocker et traiter les données dans une base de données relationnelle.
3. Intégrer les données dans une application web basée sur Flask.
4. Afficher une carte interactive pour visualiser les stations et leurs informations.

Technologies Utilisées

1. **Back-end** : Flask (Python).
2. **Base de données** : MySQL avec connecteur Python.
3. **Frontend** : HTML, CSS, JavaScript.
4. **Bibliothèques cartographiques** : LeafletJS.
5. **API** : API REST pour les données des stations.
6. **Outils de développement** : Docker pour la conteneurisation, Postman pour le test des API.

Architecture du Projet

1. Récupération des Données

L'application consomme une API REST pour obtenir les informations des stations en temps réel. Les données ont été collectées sur trois villes à savoir Lille, Paris et Bordeaux. Ces données incluent :

- Nom de la station
- Coordonnées GPS (latitude, longitude)

- Commune
- Nombre de vélos disponibles
- Nombre de places disponibles

2. Base de Données

Les données sont stockées dans une base MySQL. La structure de la table est la suivante :

Champ	Type	Description
id	INT (PK)	Identifiant unique de la station
nom	VARCHAR(255)	Nom de la station
commune	VARCHAR(255)	Commune de la station
latitude	FLOAT	Latitude de la station
longitude	FLOAT	Longitude de la station
nb_velos_dispo	INT	Nombre de vélos disponibles
nb_places_dispo	INT	Nombre de places disponibles

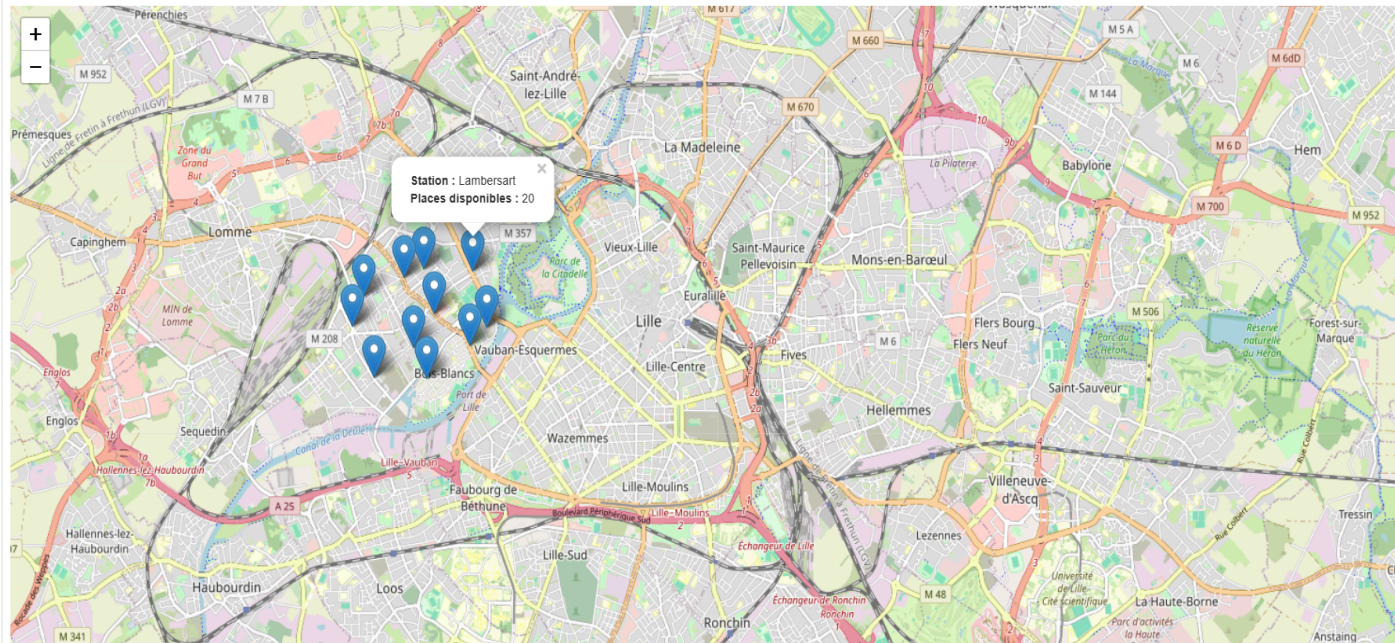
3. Application Web

- **Backend (Flask) :**
 - Point de terminaison pour servir les données des stations.
 - Utilisation de Jinja2 pour rendre les templates HTML dynamiques.
- **Frontend (LeafletJS) :**
 - Affichage d'une carte centrée sur Lille.
 - Ajout de marqueurs pour chaque station, affichant les détails dans une infobulle.

Fonctionnalités Implémentées

1. Carte interactive centrée sur Lille.
2. Marqueurs dynamiques pour chaque station avec infobulles affichant :
 - Commune (Nom de la station)
 - Nombre de places disponibles.

Carte des Stations de Vélo Disponible à proximité



Conclusion

Ce projet a permis de mettre en pratique plusieurs compétences techniques, allant de la manipulation d'API et de bases de données à l'intégration frontend-backend. Bien que des défis aient été rencontrés, ils ont été résolus grâce à une analyse rigoureuse et des bonnes pratiques de développement.