**Rapport Technique** : Application Shiny pour la Gestion des Données de Vélos

**Introduction**

Notre rapport présente une application Shiny développée pour filtrer, afficher et visualiser des données liées aux emplacements et à la disponibilité des vélos vélib dans Lyon. L’application offre une interface utilisateur interactive permettant de mieux connaitre les vélib proche d’une zone géographique sélectionné.

**Composants Principaux**

1. **Info library**

Shiny : C'est la base du programme, elle permet de programmer une application web sous R.

DT : Permet de créer des tableaux de données.

Httr : Permet de faire des appels API.

Jsonlite : Permet de lire des données JSON contenues dans l'appel API. Shinydashboard : Nous permet de créer des tableaux de bord et de les organiser.

Plotly : Permet de créer des graphiques interactifs.

Leaflet : Cette bibliothèque permet de créer des cartes pour l'utilisateur. Geosphere : Effectue des calculs géodésiques sur les données géographiques.

Dplyr : Nous permet de manipuler les données et d'implémenter des filtres.

Tidygeocoder : Permet de récupérer des informations géométriques sur une adresse donnée.

2. **Appel API**

Au début du code (ligne 18-25), un appel à l'API de "jcdecaux" est fait en utilisant "GET" pour récupérer les données qui nous intéressent, "contract=Lyon", pour ne prendre que les données de Lyon.

3. **ui**

(Ligne 33-164) Contient tous les aspects "visuels" de l'application, c'est ce qui sera montré à l'utilisateur.

4. **Server**

(Ligne 167-303) Cette partie contient tous les calculs effectués en fonction des actions de l'utilisateur.

5. **Filtrage des Données**

Le script implémente un filtrage dynamique des données en fonction du statut choisi par l'utilisateur, puis les données sont ajustées en conséquence pour refléter le statut sélectionné, d'où le filtrage dynamique. Filtrage sur le code postal ligne 172.

6. **Tableau Interactif**

Les données filtrées sont présentées dans un tableau interactif de 10 lignes par page, ce qui garantit une présentation claire et concise des informations pour l'utilisateur. Ligne 181.

7. **Rafraîchissement des Donnée**

Notre application intègre un mécanisme de rafraîchissement des données qui, lorsque l'utilisateur appuie sur le bouton de rafraîchissement, recharge et met à jour les données en temps réel. Ligne 83, bouton permettant à l'utilisateur de rafraîchir les données. Ligne 194, la mise à jour des données.

8. **Visualisations Graphiques**

8. Visualisations Graphiques Nous avons intégré dans notre application 3 types de visualisations graphiques pour représenter la partie statistique sur les Vélos à l'utilisateur. Nous avons veillé à ce que les présentations graphiques soient lisibles et compréhensibles pour tous types d'utilisateurs.

• Histogramme des Emplacements : Affiche la répartition des emplacements pour les vélos.

• Histogramme des Vélos Disponibles : Présente la répartition des vélos disponibles.

• Graphique à Secteurs de la Proportion : Illustre visuellement la proportion des vélos disponibles par rapport aux vélos non disponibles.

Ligne 172 à 224.

9. **Fichier des code postaux**

La lecture du jeu de données des codes postaux (un fichier préalablement créé pour permettre une utilisation simple, sachant que le géocodage de ce dernier prend du temps), veuillez prendre soin de modifier le chemin d'accès lié à la lecture du fichier CSV en fonction de votre lieu de stockage (ligne 14)

Pour cela installer le ficher CSV et rentrer la commande suivante :

file\_path <- "C:\\..."

data\_csv <- read.csv(data\_postcode, stringsAsFactors = FALSE).

**Conclusion**

Notre application Shiny développée offre une expérience utilisateur intuitive et informative. Les fonctionnalités de filtrage, de rafraîchissement en temps réel et les visualisations graphiques permettent aux utilisateurs d'analyser efficacement les données liées aux vélos. Cette application peut être un outil précieux pour les personnes ayant besoin de se déplacer dans la ville de Lyon.