**1. Introduction**

Cette documentation décrit l'architecture, l'installation, et les composants nécessaires pour exécuter l'application Shiny permettant l'analyse des données énergétiques des logements existants et neufs. L'application permet de visualiser des indicateurs de performance (KPI), de réaliser des analyses bivariées, et d'afficher des graphiques interactifs en fonction de différents filtres comme le code postal et le type de logement.

**2. Schéma de l'Architecture**

Une image contenant texte, capture d’écran, diagramme, ligne

Description générée automatiquement

L'application se base sur une architecture simple et modulable. Elle se divise en trois parties principales :

* **Interface utilisateur** : Permet à l'utilisateur de choisir les options de filtrage (code postal et type de logement) et de visualiser les KPI, les statistiques bivariées, et les graphiques.
* **Chargement des données** : Les fichiers CSV contenant les données sur les logements sont importés et manipulés pour afficher les résultats demandés.
* **Visualisation** : L'application génère des graphiques interactifs pour visualiser les analyses demandées par l'utilisateur.

**3. Installation de l'Application**

Pour installer et exécuter l'application sur votre poste, suivez les étapes ci-dessous :

**3.1. Prérequis**

* **R** : Vous devez avoir installé R sur votre machine. Téléchargez R depuis CRAN.
* **RStudio** (optionnel) : Vous pouvez utiliser RStudio comme IDE pour R, téléchargeable depuis [RStudio](https://www.rstudio.com/).

**3.2. Installer les packages nécessaires**

Les bibliothèques R nécessaires pour l'exécution de cette application sont les suivantes :

* shiny : Permet de créer l'application interactive.
* ggplot2 : Utilisée pour la génération des graphiques.
* dplyr : Pour la manipulation des données.
* readr : Pour l'import des fichiers CSV.
* kableExtra : Pour améliorer la mise en forme des tableaux dans le rendu.
* Leaflet : Carte interactive
* Shinydashboard : shinydashboard est un package qui permet de créer des tableaux de bord avec une mise en page élégante pour les applications Shiny
* Ggpubr : ggpubr est un package qui facilite la publication de graphiques créés avec ggplot2. Il fournit des fonctions pour personnaliser les graphiques, ajouter des annotations et effectuer des tests statistiques.
* DT : DT est un package qui permet de créer des tableaux interactifs dans R

Pour installer ces packages, exécutez le code suivant dans R :

install.packages(c("shiny", "ggplot2", "dplyr", "readr", "kableExtra",”leaflet”,”shinydashboard”,ggpubbr”,”DT”))

**3.3. Installation des données**

Les fichiers CSV nécessaires (existants\_41.csv et neufs\_41.csv) doivent être placés dans le répertoire de travail de l'application. Si les fichiers sont dans un autre dossier, veillez à modifier le chemin d'accès lors de l'import des données dans le script.

**3.4. Exécuter l'application**

1. Ouvrez RStudio ou votre environnement R.
2. Placez le script de l'application Shiny (app.R) dans votre répertoire de travail.
3. Dans la console R, exécutez le script en lançant la commande suivante :

shiny::runApp("app.R")

L'application sera accessible via un navigateur web à l'adresse locale par défaut.

**4. Packages Nécessaires**

|  |  |
| --- | --- |
| shiny | Framework pour créer des applications web interactives en R. |
| ggplot2 | Création de graphiques élégants et personnalisables. |
| dplyr | Manipulation efficace des données en R. |
| readr | Lecture rapide des fichiers CSV. |
| kableExtra | Mise en forme avancée des tableaux pour les rapports HTML et PDF. |
| Leaflet | Carte interactive |
| ggpubr | ggpubr est un package qui facilite la publication de graphiques créés avec ggplot2. |
| Shinydashboard | shinydashboard est un package qui permet de créer des tableaux de bord avec une mise en page élégante pour les applications Shiny |
| DT | DT est un package qui permet de créer des tableaux interactifs dans R |

**5. Structure du Code**

L'application Shiny est divisée en deux parties principales :

* **Interface utilisateur (ui)** : Contient les éléments de sélection et les sorties graphiques (graphiques, tables, etc.).
* **Serveur (server)** : Gère les filtres, les calculs de KPI, et la génération des visualisations.

**Exemple d'architecture dans le fichier app.R :**

ui <- dashboardPage(

dashboardHeader(title = "ENEDIS"),

dashboardSidebar(

selectInput("codePostal", "Sélectionnez un code postal", choices = unique(df$Code\_postal)),

selectInput("typeLogement", "Type de logement", choices = c("Tous", "Ancien", "Neuf"))

),

dashboardBody(

# Output des graphiques et tableaux

tabItems(

tabItem(tabName = "visualisation",

plotOutput("scatterPlot"),

valueBoxOutput("avgCostBox"))

)

)

)

server <- function(input, output, session) {

# Filtrage des données et génération des KPI et graphiques

output$scatterPlot <- renderPlot({

# Graphique de dispersion avec régression

ggplot(df\_filtered(), aes(x = Surface\_habitable\_logement, y = Coût\_chauffage)) +

geom\_point() + geom\_smooth(method = "lm")

})

}

shinyApp(ui, server)

**6. Conclusion**

Ce document présente une vue d'ensemble de l'application Shiny pour l'analyse des données de logements. Elle est basée sur des technologies open source comme R et Shiny, ce qui la rend facilement déployable sur tout poste disposant d'un environnement R configuré.