Table des matières

[Source code 1](#_Toc151017059)

[Schéma de la base de données 2](#_Toc151017060)

[Procédure de création de la base de données 2](#_Toc151017061)

[Démarrage de l’application Springboot 2](#_Toc151017062)

[Liste des contrôleurs 2](#_Toc151017063)

[B2CController 2](#_Toc151017064)

[Liste de tous les clients particuliers : 3](#_Toc151017065)

[Recherche par référence EKL 4](#_Toc151017066)

[Recherche par référence inexistant 5](#_Toc151017067)

[B2BController 5](#_Toc151017068)

[Liste de tous les clients professionnels : 6](#_Toc151017069)

[Recherche par référence EKL : 7](#_Toc151017070)

[Recherche par référence inexistant : 7](#_Toc151017071)

[BillingController 8](#_Toc151017072)

[Lister toutes les factures pour un client : 8](#_Toc151017073)

[Rechercher une facture client par mois : 9](#_Toc151017074)

[Axes d’améliorations : 9](#_Toc151017075)

# Source code

<https://github.com/tarekTabib/energie-api>

# Schéma de la base de données

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

Description générée automatiquement

# Procédure de création de la base de données

Exécuter le script energie-db.sql

# <https://github.com/tarekTabib/energie-api/blob/master/src/main/resources/sql/energie-db.sql>

# Démarrage de l’application Springboot

java -jar ou bien lancer l’application BillingApplication dans un IDE (eclipse, intellij,…)

# Liste des contrôleurs

## B2CController

Permet de lister les clients particuliers ou de rechercher un particulier par référence : (EKL + 9 caractères numériques)

### Liste de tous les clients particuliers :

Une image contenant texte, capture d’écran, logiciel, Logiciel multimédia

Description générée automatiquement

### Recherche par référence EKL

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

### Recherche par référence inexistante

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

# B2BController

Permet de lister les clients professionnels ou de rechercher un professionnel par référence : (EKL + 9 caractères numériques)

## Liste de tous les clients professionnels :

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

## Recherche par référence EKL :

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

## Recherche par référence inexistante :

Une image contenant texte, capture d’écran, logiciel, Icône d’ordinateur

Description générée automatiquement

# BillingController

Permet de lister toutes les factures pour un client donné et de chercher une facture client pour un mois donné

## Lister toutes les factures pour un client :

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

## Rechercher une facture client par mois :

Une image contenant texte, capture d’écran, logiciel, Page web

Description générée automatiquement

# Axes d’améliorations :

Ajout d’APIs de création, modification, suppression…

Gestion des erreurs (404, 500…)

Sécurité (Token JWT, …)

Gestion des privilèges (Spring security)

Création de tests unitaires avec Mockito