

Pré-requis

Version de Java :

Ce projet a été développé sous la **JDK 19**.
Cette version est présentement [disponible sur cette page](#).

La commande `java --version` permet de vérifier rapidement la version.

Base de donnée :

La base de données utilisée est un serveur **MariaDB version 10.4**.
Par souci de simplicité, on peut l'installer via [XAMPP](#) ce qui permet d'utiliser phpMyAdmin.

La commande `mariadb --version` permet de vérifier la version.
Il est aussi possible de consulter cette information sur la page d'accueil de phpMyAdmin.

Bibliothèques :

Dans un premier temps, nous procéderons au téléchargement des librairies.
Ensuite, nous traiterons leur installation.

Bibliothèques fournies sur le dépôt :

- JUnit 5.8.1: *junit-platform-console-standalone-1.9.1.jar*
- Gson 2.10: *gson-2.10.jar*
- Argon2-jvm 2.11: *Argon2-jvm-2.11.jar*, *Argon2-jvm-nolib-2.11.jar*
- JNA version 5.12.1: *jna-5.12.1.jar* utilisée par Argon2-jvm 2.11
- Connecteur MYSQL version 8.0.13: *mysql-connector-java-8.0.13.jar*

Pour les retrouver, veuillez consulter le dossier **/librairies** disponible à la racine du dépôt.

Bibliothèques non-fournies :

JavaFX étant dépendante du système d'exploitation, nous avons choisi de ne pas la fournir.
Notez que le projet a été développé sous la **version 19 de JAVA**FX [disponible sur cette page](#).

Installation du projet :

Clonage du dépôt :

- 1/ Sélectionnez la SDK 19 comme JRE.
- 2/ Importez le dépôt : `https://github.com/g4bey/Gestionnaire-Distribution-Agricoles.git`

Attention! : Utilisez **VSCode**, **IntelliJ** ou la ligne de commande preference. En effet, il est difficile d'importer le projet correctement sur eclipse qui veut absolument un package autre que celui par défaut. Nous avons essayé de créer un package par défaut pour eclipse... cependant lors de l'import, il créait un nouveau package ce qui posait des problèmes. Si refactor pour accommoder le changement, il faudra modifier les liens dans les vues FXML qui ne seront plus à jour, et potentiellement les chemins relatifs dans les contrôleurs.

Ajout des bibliothèques :

Bibliothèques fournies :

- 1/ Copiez les jar fournies dans le dossier `/librairies/` dans un dossier `/lib/` à la racine.
- 2/ Ajoutez ces jars au buildpath.
- 3/ Vérifiez qu'elles apparaissent bien dans le dossier `Referenced Libraries`.

Installation de JavaFX 19 :

- 1/ Décompressez le dossier `javafx-sdk-19` dans le dossier `/lib/`
- 2/ Ajoutez les jars dans `/lib/javafx-sdk-19/lib/` au buildpath.

Configuration du projet :

Lancement du projet JAVAfX :

- 1/ Créez une nouvelle configuration de type **JAVA Application**.
- 2/ Vérifiez que la JRE est bien la **JDK 19**.
- 3/ Sélectionnez `src/GDA.java` comme classe principale.
- 4/ Ajoutez les VM arguments suivants :

```
--module-path <CHEMIN>/lib/javafx-sdk-19/lib/  
--add-modules=javafx.base,javafx.graphics,javafx.controls,javafx.fxml,javafx.web,javafx.media
```

<CHEMIN> est le chemin absolu vers le dossier `/lib/javafx-sdk-19/lib/`.

Lancement des tests JUnit :

- 1/ Créez une nouvelle configuration de type **JUnit**.
- 2/ Vérifiez que la JRE est bien la **JDK 19**.
- 3/ Sélectionnez `src/tests/` comme dossier contenant les tests.

Mise en route des bases de données :

- 1/ Lancez MariaDB version 10.4 sur le port 3306.
- 2/ Exécutez les scripts `deploy-test.sql` et `deploy-prod.sql` afin de créer les bases **GDATest** et **GDAProd**.
- 3/ Exécutez ensuite le script `dump-production.sql`.

Configuration par défaut :

Vérifiez que les informations dans `src/ressources/config.properties` correspondent à la configuration suivante.

GDATest:

```
db.testing.username=GDATest  
db.testing.password=1234  
db.testing.url=jdbc:mysql://localhost/GDATest
```

GDAProd:

```
db.production.username=GDA  
db.production.password=1234  
db.production.url=jdbc:mysql://localhost/GDAProd
```

API OpenRouteService `ORS_KEY=5b3ce3597851110001cf6248b0564520d72840508742e806dd03e7a1`