



README

Application web

2016

VOLKSWAGEN & AUDI

Ralph GAUME

Xavier SIMON

TABLE DES MATIERES

Volkswagen & audi.....	1
Table des matières	2
I. Présentation du projet.....	3
II. Installation du projet.....	3
A. Outils nécessaires.....	3
B. Installation	4
III. L'architecture de l'application.....	6
A. Le dossier src/main/java/*	7
B. Le dossier src/main/resources/*	8

I. Présentation du projet

L'objectif principal de ce projet est de développer une application web qui proposera diverses fonctionnalités mais dont la principale est de fournir un catalogue de véhicules, appartenant aux marques Volkswagen et Audi, qu'un client pourra sélectionner puis personnaliser. À la fin de sa personnalisation, le client pourra sélectionner une date selon sa convenance pour réserver un essai du véhicule au sein de notre concession.

II. Installation du projet

A. Outils nécessaires

a. Service MySQL



Pour stocker et gérer les données de l'application web nous nous sommes servis du système de gestion de base de données relationnelles parmi les plus utilisés au monde, MySQL. Pour lancer et configurer notre base de données, nous avons utilisé plus précisément MAMP.

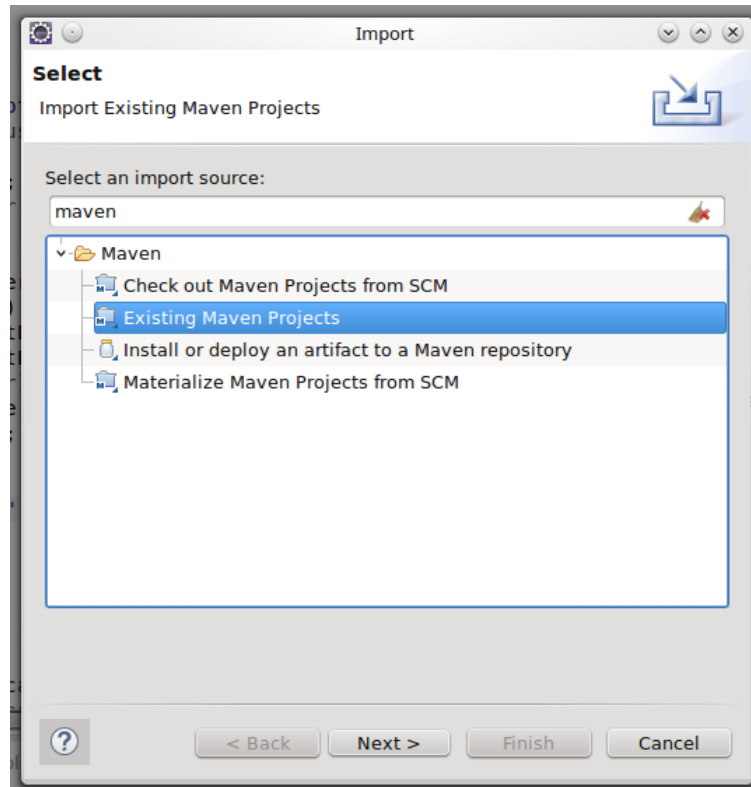
b. Eclipse



Nous avons utilisé Eclipse comme environnement de développement qui à l'avantage de fournir de nombreux outils ou plugins utilisés à la réalisation de notre application web.

B. Installation

- 1) La première étape est d'aller dans Eclipse, puis d'importer un projet Maven existant :



- 2) Renseigner l'archive du projet puis suivre les étapes indiquées par Eclipse. L'archive peut être téléchargée depuis github : <https://github.com/Ralph21/projet-web-Xavier-Ralph/archive/master.zip>. Ou récupérer l'archive depuis le dossier compressé qui contient ce même document.

3) Pour la partie MySQL :

- a. Créer une base (nom par défaut « basedetest ») et y affecter un utilisateur avec les droits sur cette base (veuillez vous renseigner à la documentation officielle de MySQL)
- b. Modifier SecurityConfig.java (src/main/java/com.miage.config) pour y renseigner les valeurs d'accès à la base :

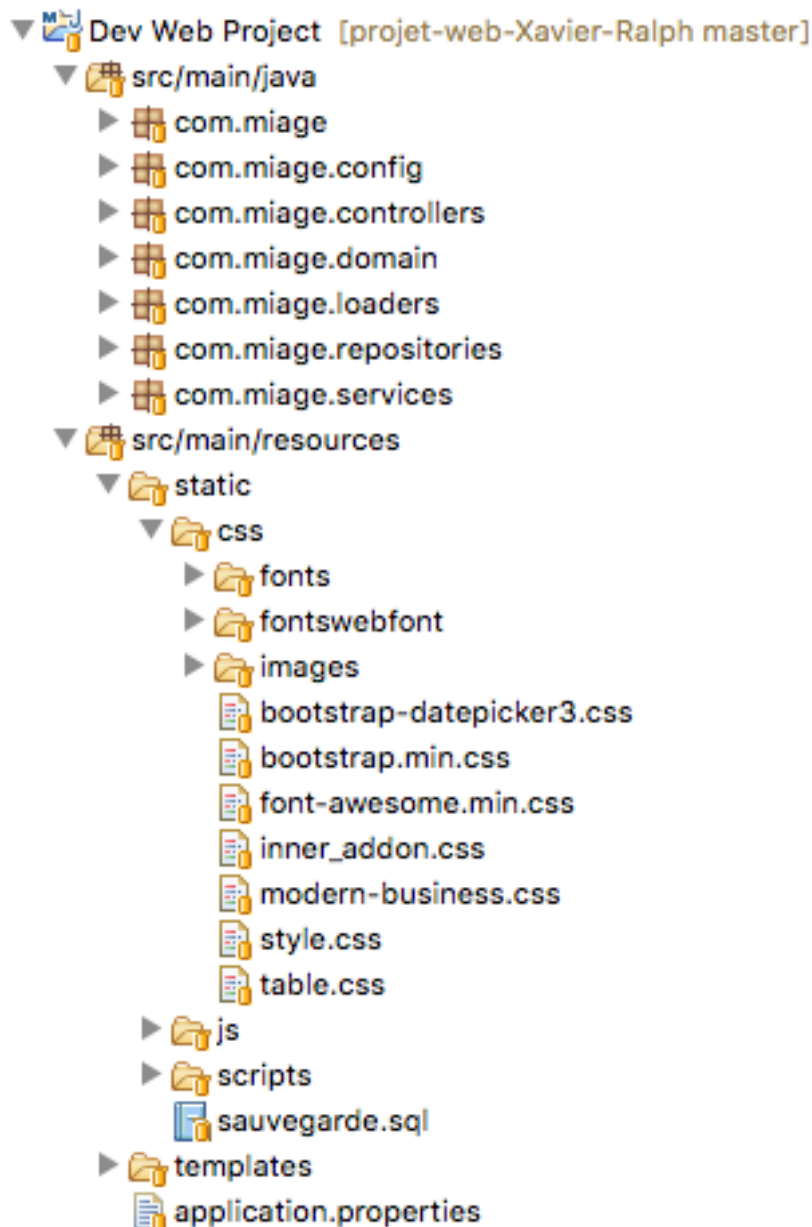
```
@Bean(name = "dataSource")
public DriverManagerDataSource dataSource() {
    DriverManagerDataSource driverManagerDataSource = new DriverManagerDataSource();
    driverManagerDataSource.setDriverClassName("com.mysql.jdbc.Driver");
    driverManagerDataSource.setUrl("jdbc:mysql://localhost:3306/basedetest");
    driverManagerDataSource.setUsername("root");
    driverManagerDataSource.setPassword("root");
    return driverManagerDataSource;
}
```

- 4) Puis lancer l'application via le fichier Application.java (src/main/java/com.miage).

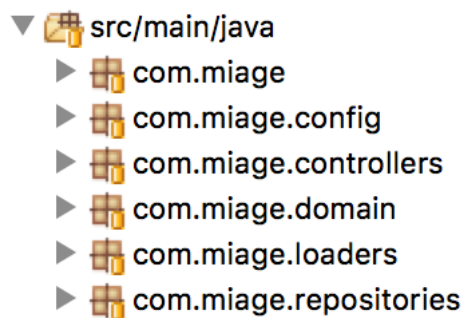
III. L'architecture de l'application

Le projet se décompose en deux parties, la première qui gère les contrôleurs (le mapping des requêtes), la sécurité et les données (base de données) et la seconde qui gère les ressources web (fichiers HTML, fichiers CSS et JavaScript ou encore les images).

La structure de l'application est donc la suivante :



A. Le dossier src/main/java/*

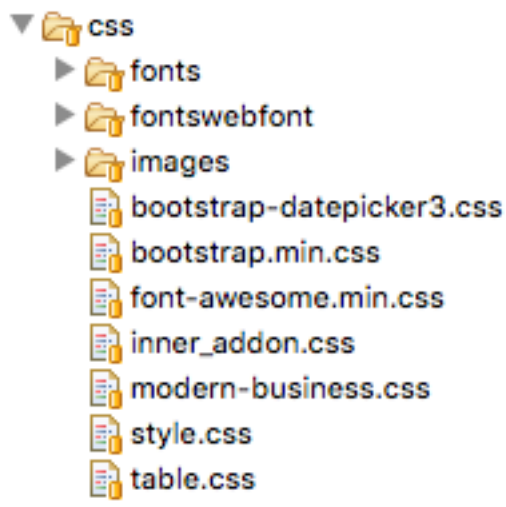


Ce dossier contient différents fichiers java permettant de gérer divers éléments de l'application web.

- *com.miage*, qui contient le fichier qui permet de lancer l'application web.
- *com.miage.config*, qui contient les fichiers de configuration de la sécurité et qui gère la connexion avec la base de données.
- *com.miage.controllers*, contient les fichiers qui définissent et gèrent tous les requests mappings de l'application.
- *com.miage.domain*, contient les classes (beans) qui vont être persistées en base et utilisées pour modéliser cette dernière.
- *com.miage.repositories*, contient l'ensemble des repositories qui permet de se connecter à la base de données et de l'interroger par le biais de ces beans pour récupérer, traiter et sauvegarder l'information.
- *com.miage.loaders*, le dossier n'est désormais plus utile.

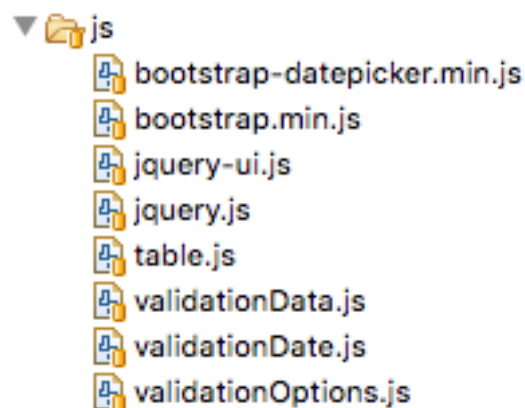
B. Le dossier src/main/resources/*

Le dossier *src/main/resources* contient toutes les ressources nécessaires à l'application web. Tout d'abord le dossier *templates* qui contient les pages ***.html**. Quant au dossier *static*, il contient notamment les fichiers ***.css** et ***.js** mais également les images et diverses icônes télécharger depuis bootstrap.



Le dossier *css* contient les fichiers ***.css**, qui sont pour la plupart des standards bootstrap, mais également les images et icônes (dossiers *fonts* et *fontswefont*) nécessaires à l'application web.

Le dossier *js* contient les fichiers JavaScript utilisés dans le site pour gérer diverses choses. La plupart sont des standard bootstrap ou jQuery mais il y a également les fichiers **validation*.js** qui contrôlent les informations saisies par l'utilisateur.



Le fichier **validationData.js** contrôle que l'utilisateur rentre les bonnes informations lors de son inscription. Les règles sont les suivantes :

- Il doit sélectionner une civilité (homme ou femme)
- Le nom et prénom sont obligatoire et doivent contenir entre 2 et 30 caractères
- L'âge est obligatoire et doit être contenir que des chiffres, la personne doit être également majeur
- L'e-mail est obligatoire (sera utilisé comme identifiant)
- Le mot de passe doit contenir :
 - Au moins 8 caractères
 - Au moins une minuscule
 - Au moins une majuscule
 - Au moins un chiffre.

Et doit être saisie une seconde fois et la fonction vérifie que les deux mots de passes saisie par l'utilisateur sont bien identiques.

Les fichiers **validationDate.js** et **validationOptions.js** vérifient que l'utilisateur lors de la personnalisation de son véhicule et de la réservation de l'essai en concession rentre bien toutes les informations nécessaires.

Volkswagen & Audi