

# JEE Projet Plateforme de gestion de machines à café

Christopher Rousseau

19 décembre 2019

## Contents

<b>Plateforme de gestion de machines à café</b>	<b>2</b>
<b>Modalités</b>	<b>2</b>
<b>Authentification</b>	<b>2</b>
<b>Front</b>	<b>3</b>
Partie administration . . . . .	3
Partie utilisateur . . . . .	4
<b>Travail à rendre</b>	<b>5</b>
Projet . . . . .	5
Rapport . . . . .	5
<b>Annexe</b>	<b>7</b>
Spécifications du service fourni . . . . .	7

## Plateforme de gestion de machines à café

Pour ce projet, vous réaliserez un ensemble de composants visant à aider à la mise en place d'une plateforme de gestion de machines à café, permettant à la fois au personnel de gérer les stocks, mais aussi aux responsables d'accéder à une partie administration ainsi qu'à une partie "dashboard".

La plateforme doit être capable de gérer plusieurs machines à café, possédant une version, et un type.

## Modalités

Afin de réaliser ce projet, vous aurez la possibilité de travailler par groupes de 3 à 4, une partie de ce projet sera donc optionnelle pour les plus petits groupes ainsi que les alternants.

Le principal objectif du projet est de vous apprendre à utiliser et gérer plusieurs ressources / composants et de les interconnecter :

- Back office
- API REST
- Base de données
- Interface utilisateur
- Partie dashboard
- Partie administration
- Services externes (iot / machines à café)

Les technologies autorisées pour ce projet sont :

- Spring Boot
- Spark Java
- Spring security
- Framework Angular

## Authentification

Afin de gérer vos utilisateurs, vous devrez créer un service d'authentification. Ce service devra gérer les rôles utilisateurs (utilisateur/administrateur)

Cette authentification comprendra :

- Une adresse email comme identifiant
- Un mot de passe avec une règle cohérente (affichée sur la page d'authentification)

Prenez garde de ne pas stocker les mots de passe en clair dans la base de données, et à vous protéger d'éventuelles injections SQL.

Le format de l'identifiant devra être une adresse email valide.

Vous pourrez vous aider de ce tutoriel spring : <https://spring.io/guides/gs/securing-web/>

## Front

Dans cette partie, vous devrez utiliser Angular afin de réaliser l'interface utilisateur

Ajoutez dans votre rapport une petite comparaison des technologies suivantes (points forts / points faibles).

- react.js
- angular
- vue.js
- angular.js

## Partie administration

La partie administration devra comprendre les fonctionnalités suivantes :

- Gestion des utilisateurs (Listing + CRUD)
- Gestion des Machines à café (Listing + CRUD)
- Listing des stocks de produits présents dans les machines à café avec un icône d'alerte en cas de rupture de stock sur au moins un produit (dashboard)
- Export (PDF ou json ou XML ou csv) listing utilisateurs
- Export (PDF ou json ou XML ou csv) stocks avec état (état actuel de la plateforme)

Le type d'export est au choix, vous pouvez en implémenter un ou plusieurs.

**Optionnel pour les groupes comprenant des alternants et les groupes de moins de 4**

Le service de machines à café utilisé chez vos clients ne répond plus aux exigences et a besoin d'une mise à niveau. Afin de répondre à de nouvelles exigences vous devrez développer un nouveau service capable de le remplacer.

Ce service devra suivre les règles suivantes :

- Il sera configurable de la même manière que le service d'origine fourni avec un fichier de configuration ou une configuration manuelle par API REST.
- une API REST permettra de gérer ses produits.
- une API REST permettra de consommer ses produits ou de renouveler les stocks.
- Il enverra ses informations de manière régulière à la plateforme configurée, avec une fréquence définie dans le fichier de configuration.
- sa version sera la suivante : 1.1
- sa version sera renvoyée avec ses informations de manière régulière, via le même appel à la plateforme.

## **Partie utilisateur**

La partie utilisateur devra comporter les informations suivantes :

- une page d'accueil accessible en mode connecté et non-connecté
- Un formulaire d'authentification
- Accès et modification des informations de l'utilisateur connecté
- Accès aux informations de configuration des machines à café (fréquence / port / statut si présent)
- Configuration d'une machine à café
- ajout/suppression des produits d'une machine (attention, si le produit est supprimé, il faudra aussi communiquer avec le distributeur de boissons)
- Suppression d'une machine à café (devra supprimer toutes ses informations)

## **Optionnel pour les groupes comprenant des alternants et les groupes de moins de 4**

Pour la page d'accueil, vous rendrez disponibles quelques informations utiles :

- Un "widget" météo intégré sous forme d'iframe `http://www.infoclimat.fr/api-previsions-meteo.html?id=2988507&cntry=FR`
- Un encart presse (type d'implémentation : `http://www.lemurdelapresse.com/`)
- Un encart affichant le resultat d'une requête GET sur le site `https://estcequecestbientotleweekend.fr`

## Travail à rendre

Le travail à rendre prendra la forme d'une archive au format **.zip** et comprendra votre projet ainsi qu'un rapport détaillé comme décrit ci-dessous.

Vous enverrez votre projet au plus tard le Mardi 28 Janvier à l'adresse suivante `c.rousseau@attineos.com`. N'oubliez pas de préciser la composition de votre groupe dans le corps de l'email.

Si la pièce jointe est trop volumineuse, pensez aux solutions d'hébergement de contenu temporaires.

## Projet

L'archive devra comprendre les sources de votre projet, ainsi que un / des fichier(s) **.war** deployables sur un serveur web. Pour les groupes rendant un service de distributeur de boissons, ce service prendra la forme d'un fichier **.jar** ainsi que d'un fichier de configuration. Il sera également présent dans votre **.zip**

## Rapport

Le rapport sera au format PDF et devra comprendre les informations suivantes :

- Vos interfaces de service (vous pourrez les décrire sous forme de tableaux,
- Un diagramme UML
- Un schéma relationnel
- La comparaison des technologies pour la partie interface utilisateur (UI/UX)
- Le détail de l'implémentation de l'authentification
- Un manuel partie Administration
- Un manuel utilisateur

- un manuel comprenant les interfaces et les fonctions du service de distributeur de boissons (pour les groupes de 4)
- Une notice d'exécution / installation

## Annexe

### Spécifications du service fourni

**Service** Le service fourni se comporte comme un simulateur de distributeur. Autrement dit, pour chaque produit créé, il renverra des valeurs aléatoires numériques (positives ou négatives) ou une chaîne de caractères. Les valeurs sont volontairement parfois incohérentes afin de vous permettre de les gérer au mieux dans votre application.

Ce service utilise un fichier de configuration à placer à côté du fichier **.jar** et devra se nommer **agent.cfg**

Ce fichier de propriétés devra contenir un format similaire à ce qui suit :

```
agentPort = 9090
serverPort = 8080
serverURL = http://127.0.0.1
frequency = 15000
```

**agentPort** est le port sur lequel l'API REST du distributeur sera exposé **serverPort** et **serverURL** permettront de configurer l'adresse du service que vous aurez développé dans votre plateforme et qui sera chargée de recueillir les informations de chaque distributeur.

**frequency** est la fréquence à laquelle votre service va envoyer ses informations au serveur, exprimée en ms

**API REST** Les méthodes exposées sont les suivantes :

Route	Méthode	Body
/version	GET	
/products	GET	
/product	POST	{"productName": "café"}
/product	DELETE	{"productName": "café"}
/config	POST	{"port": 8080, "url": "http://127.0.0.1"}
/config	GET	