Afin de réaliser ce travail, j’ai travaillé de la manière suivante.  
Dans un premier temps, j’ai d’abord suivis un cours en ligne qui expliquait ce qu’est Spring Boot et comment créer une première application. Ce cours concernait principalement les bases de ce framework ainsi que le modèle MVC et son implémentation.

Voici un lien vers ce cours : <https://www.youtube.com/watch?v=9SGDpanrc8U&ab_channel=Amigoscode>

Ensuite, j’ai suivis deuxième cours qui expliquait comment créer différents micro-services et les faire communiquer entre eux.

Voici le lien vers cette playlist de vidéos : <https://www.youtube.com/playlist?list=PLqq-6Pq4lTTZSKAFG6aCDVDP86Qx4lNas>

Enfin, j’ai suivis un troisième cours concernant cette fois-ci le principe de sécurité avec Spring Boot. Ce cous d’envire 5 heures explique la limitation d’accès aux url, les différentes technique d’authentification (Basic Auth, Form Auth, JSON Web Tokens) ainsi que l’implémentation de celle-ci avec une base de données.

Voici un lien vers ce cours : <https://www.youtube.com/watch?v=her_7pa0vrg&ab_channel=Amigoscode>

Ayant méticuleusement suivis ces différents tutoriels, je n’ai pas rencontré beaucoup de problème en travaillant sur ce dossier. Voici cependant deux points qui m’ont posé problème.

1. *Insertion de la date de la commande dans la base de données*

N’ayant jamais travaillé avec JPA/Hibernate et PostgreSQL avant, je me suis retrouvé face à un problème lorsque j’ai voulu enregistrer une commande (plus particulièrement sa date) dans la base de données. Après différents essais infructueux, je me suis simplement orienté vers l’attribution d’une valeur par défaut (la date de création de l’objet commande) pour la date de la commande. JPA et Hibernate vont ensuite se charger de faire l’insertion dans la DB. Voici ce que cela donne dans le code :



Bien que je sois parvenu à cette solution par moi-même, voici tout de même les ressources consultées :

• <https://www.postgresqltutorial.com/postgresql-now/>

• <https://www.postgresql.org/docs/8.2/functions-datetime.html>

• <https://stackoverflow.com/questions/38245025/how-to-insert-current-datetime-in-postgresql-insert-query>

1. *Problème de versions entre les différentes dépendances dans le Pom.xml*

Une fois arrivé à la fin du développement, il était temps de connecter l’application principale au serveur Eureka et plus particulièrement au service TVA et qui est un projet séparé su reste qui vient se connecter au serveur Eureka pour se rendre disponible. Lors de la mise en place du Load Balancing coté client, je me suis retrouvé face à cette erreur lorsque que j’essayais de build le projet principal : « org.springframework.cloud:spring-cloud-starter-netflix-eureka-client:unknown ».

Après quelque recherches, j’ai rapidement compris que le problème venait de la compatibilité des différentes versions que j’utilisais avec mes dépendances dans le Pom.xml. Cependant, comprendre d’où venait le problème, ne m’avançait pas sur comment le résoudre.

Je suis alors retourné voir dans le projet de test que j’avais fait en regardant le cours sur les micro-services. Afin d’avoir une idée des versions compatibles entre-elles. Après avoir essayé (beaucoup) de versions et changement dans le Pom.xml , ça ne fonctionnait toujours pas. J’ai alors décidé de downgrade ma version de Spring et , magie, tous les problèmes étaient résolus. A la base, je travaillais avec la dernière version du framework et il semblerait que celle-ci ne soit pas (encore) compatible avec tout. Une fois le problème résolu, j’ai enfin pu terminer le développement.

Voici deux des ressources consultées :

• <https://stackoverflow.com/questions/63693570/cannot-resolve-org-springframework-cloudspring-cloud-starter-netflix-eureka-cli>

• <https://www.youtube.com/watch?v=jhvJg4tDrpw&list=PLqq-6Pq4lTTZSKAFG6aCDVDP86Qx4lNas&index=21&ab_channel=JavaBrains>