

NOM	BOUCLET
Prénom	Florian
Date de naissance	07/01/1994

## Copie à rendre

### Graduate Développeur

(Android, Angular, Flutter, Front End, Full Stack, IOS, PHP/Symfony)

## Documents à compléter et à rendre

## Objectif & Livrables

### Livrables :

Le lien du (ou des) dépôt(s) Github public, où sera présent le code de vos applications

Le lien de la version en ligne de l'application web

Le lien vers votre logiciel de gestion de projet pour cet ECF (Trello, Jira, etc...)

Devra aussi se trouver:

- Un fichier readme.md contenant la démarche à suivre pour l'exécution en local
  - Ne pas oublier dans ce readme l'explication de la création d'un administrateur pour le back-office de l'application web
- Les fichiers de création et d'alimentation de la base de données de votre application web (migrations, fixtures ou scripts SQL à la main)
  - **Attention:** Générer du SQL par le biais d'une interface graphique (ex: PHPMYAdmin), et donc sans avoir à coder, conduira à la non-validation d'une des compétences du titre professionnel!
- Une documentation technique au format .pdf avec vos réflexions initiales sur le projet (technologies choisies, configuration de l'environnement de travail, diagramme de classe ou Méthode MERISE, diagramme de cas d'utilisation, diagramme de séquence...)
  - **Attention:** Chaque diagramme précisé dans l'annexe doit être obligatoirement présent! Ils sont en lien direct avec une des compétences du référentiel.
- Une charte graphique au format .pdf regroupant:
  - La palette de couleurs
  - La palette des polices d'écriture choisies
  - L'export des maquettes attendues
    - wireframe et/ou mockup pour la version mobile ET desktop

**Attention!** Pour respecter les contraintes de temps et valider toutes les compétences du référentiel, il est vivement conseillé de ne pas passer trop de temps sur le design de vos applications. S'il est minimaliste, mais cohérent, alors c'est suffisant.

## Livrables :

Tous les documents sont disponibles dans GitHub dans le dossier ECF.

### **A. Liens**

Dépôts GitHub :

- Application front end :
  - ✓ <https://github.com/Florian-Bclt/Quai-antique>
- Application back end :
  - ✓ <https://github.com/Florian-Bclt/quai-antiqueapi>

Version des applications en ligne :

- <https://quai-antique.netlify.app/>
- <http://app-8a84a013-ae4a-4e4f-9176-5e1c62e8a561.cleverapps.io/>

### **B. Fichier ReadMe**

Les fichiers ReadMe sont présents à la racine de chacune des applications sur GitHub, elles décrivent les étapes à suivre pour l'installation.

### **C. Fichiers de base de données**

- Lien de l'image Docker à copier dans le terminal : `docker pull boucletf/postgres`

A la racine du projet vous pourrez retrouver le fichier `schema.gql` contenant l'ensemble des données nécessaire au fonctionnement de l'application.

***Ne souhaitant pas partager mes données sur le repository GitHub je mets en annexe la démarche à suivre pour créer un administrateur.***

### **D. Documentation technique**

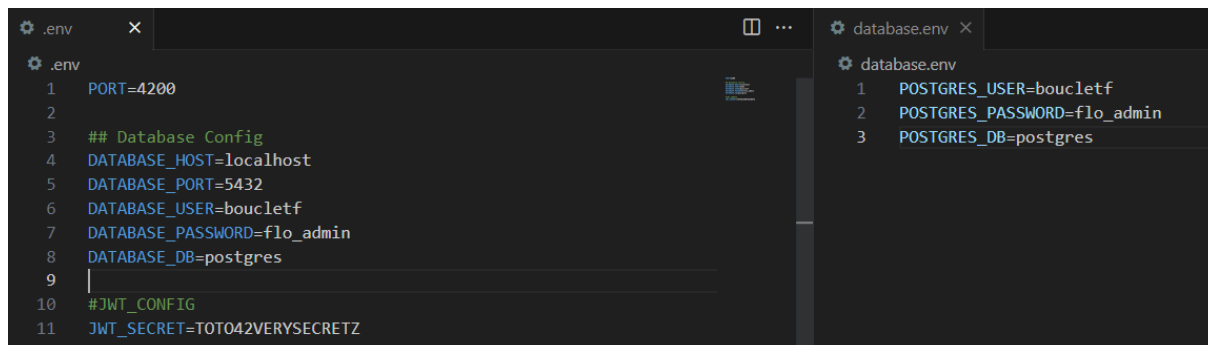
La documentation technique est présente dans le dossier ECF de l'application front end, vous pourrez y retrouver les différents diagrammes.

### **E. Charte graphique**

La charte graphique est présente dans le dossier ECF de l'application front end, vous pourrez y retrouver les différentes maquettes ainsi que la palette.

## ANNEXES

Afin d'utiliser l'application en local, vous aurez besoin d'avoir accès à la base de donnée. Lorsque vous aurez cloné l'image Docker comme expliqué dans le ReadMe de l'API, vous devrez créer deux fichiers à la racine du projet, .env et database.env que je n'ai pas pu partager sur GitHub pour des raisons de sécurité.



```
.env
1  PORT=4200
2
3  ## Database Config
4  DATABASE_HOST=localhost
5  DATABASE_PORT=5432
6  DATABASE_USER=boucletf
7  DATABASE_PASSWORD=flo_admin
8  DATABASE_DB=postgres
9
10 #JWT_CONFIG
11 JWT_SECRET=TOT042VERYSECRETZ

database.env
1  POSTGRES_USER=boucletf
2  POSTGRES_PASSWORD=flo_admin
3  POSTGRES_DB=postgres
```