

# Rapport d'Application Complet : YaoundeLoc

## I. Lancement et Configuration de l'Application

Pour lancer l'application, vous devez vous assurer que l'environnement de développement est prêt.

### 1. Prérequis

Java 17+ : Assurez-vous d'avoir le JDK 17 (ou supérieur) installé.

MySQL Server(MARIADB) : Le serveur de base de données MySQL doit être démarré.

Maven : Installé ou intégré dans votre IDE (visual studio code).

Dossier d'Upload : Un dossier nommé uploads sera créé automatiquement à la racine du projet par FileStorageService.java pour stocker les images.

### Démarrage de l'Application

Via Maven : Ouvrez votre terminal à la racine du projet et exécutez :

Bash :

**mvn clean install -DskipTests**

ensuite tapez :

**java -jar target/nom\_du\_fichier.jar** (fichier qui va être créé dans:  
/TP\_INF331\_MOO/TP331\_MOO/yaounde-loc/target)

**ex :** yaounde\_loc-0.0.1-SNAPSHOT.jar

## II.      Fonctionnement et Architecture

L'application suit une architecture N-tiers (ou couches) standard pour Spring Boot.

Couche	Fichiers Correspondants	Rôle
<b>Contrôleurs (API)</b>	*Controller.java	Point d'entrée HTTP. Gère le routage et le format de la réponse (JSON).
<b>Services (Logique Métier)</b>	*Service.java	Contient toute la logique de l'application (validation, calculs, modération). C'est le cœur de l'application.
<b>Accès aux Données (JPA)</b>	*Repository.java	Interface avec la base de données. Gère les CRUD (Create, Read, Update, Delete).
<b>Sécurité/ Modèles</b>	*Config.java, *TokenProvider.java, *Filter.java, *Entity.java	Gère l'authentification (JWT) et la structure des données (JPA).

## Fonctionnalités Clés

Domaine	Description	Rôles Impliqués
<b>Authentification</b>	Inscription pour Locataire/Propriétaire et Connexion (génération de token JWT).	Tous
<b>Annonces (Public)</b>	Consultation des annonces <b>validées</b> . Recherche par quartier, type et prix.	Tous (non authentifié)
<b>Annonces (Propriétaire)</b>	Création, modification, suppression d'une annonce. L'annonce passe en statut EN_ATTENTE après chaque création/modification.	ROLE_PROPRIETAIRE
<b>Modération (Admin)</b>	Lister, valider et rejeter les annonces en attente.	ROLE_ADMIN
<b>Images</b>	Téléchargement sécurisé d'images et service public pour les afficher.	ROLE_PROPRIETAIRE (Upload) / Tous (Consultation)
<b>Favoris</b>	Ajout et suppression d'annonces dans une liste de favoris personnelle.	ROLE_LOCATAIRE

### III. Guide Postman (API)

Toutes les requêtes doivent être adressées à l'URL de base :  
<http://localhost:8080>.

#### 1. Authentification & Utilisateurs (Endpoints Publics)

Action	Méthode	Endpoint	Body (JSON)	Exemples de Données
<b>1. Connexion</b>	POST	/auth/login	{ "email": "...", "password": "..." }	admin@yaoundeloc.com, password
<b>2. Inscription Locataire</b>	POST	/auth/register	{ "email": "...", "password": "...", "fullName": "...", "phone": "..." }	locataire@mail.com, pass123
<b>3. Inscription Proprio</b>	POST	/auth/register-proprietaire	{ "email": "...", "password": "...", "fullName": "...", "phone": "..." }	proprio@mail.com, pass123

**Résultat de la Connexion :** Le serveur retourne un JSON avec le **token JWT** et le **role**. Vous devez copier ce **TOKEN** pour les actions suivantes (voir le **Header** ci-dessous).

## **2. Annonces (Accès Public)**

<b>Action</b>	<b>Méthode</b>	<b>Endpoint</b>	<b>Header</b>	<b>Body</b>	<b>Notes</b>
<b>4. Lister Annonces Validées</b>	GET	/annonces	Aucun	Aucun	Retourne toutes les annonces avec StatutAnnonce.VALIDEE.
<b>5. Rechercher Avancée</b>	GET	/annonces/ search? quartier=ba st*&prixMi n=10000	Aucun	Aucun	Utilise des query params optionnels.
<b>6. Détail Annonce</b>	GET	/annonces/ {id}	Aucun	Aucun	Fonctionne uniquement pour les annonces VALIDEE.

### 3. Annonces (Actions du Propriétaire)

**Header :** Pour ces actions, vous devez utiliser le token d'un utilisateur `ROLE_PROPRIETAIRE`.

**Key :** Authorization **Value :** Bearer  
VOTRE\_TOKEN\_PROPRIETAIRE

Action	Méthode	Endpoint	Body (JSON)	Notes
<b>7. Créer une Annonce</b>	POST	/annonces	{ "titre": "...", "description": "...", "prix": 50000, "quartier": "...", "typeBien": "..."} }	L'annonce passe en statut <code>EN_ATTENTE</code> .
<b>8. Lister Mes Annonces</b>	GET	/annonces/ my-listings	Aucun	Retourne toutes les annonces (y compris <code>EN_ATTENTE</code> et <code>REJETEE</code> ) du propriétaire.
<b>9. Modifier une Annonce</b>	PUT	/annonces/{id}	(Même JSON que la création)	L'annonce repasse en statut <code>EN_ATTENTE</code> pour re-modération.
<b>10. Supprimer une Annonce</b>	DELETE	/annonces/{id}	Aucun	Le propriétaire doit être l'auteur de l'annonce.

## 4. Images (Upload)

Action	Méthode	Endpoint	Header	Body	Notes
<b>11. Uploader une Image</b>	POST	/annonces/ {annonceId}/ upload-image	<b>Header</b> + Content-Type: multipart/form-data	<b>Body :</b> Utiliser form-data, champ key: file, value: [sélectionner un fichier].	Nécessite un token ROLE_PROPRIETAIRE ET que l'utilisateur soit propriétaire de l'annonce.
<b>12. Afficher l'Image</b>	GET	/files/ {fileName}	Aucun	Aucun	Endpoint public. Le fileName est renvoyé après l'upload.

## **5. Favoris (Actions du Locataire)**

**Header :** Pour ces actions, vous devez utiliser le token d'un utilisateur ROLE\_LOCATAIRE.

**Key :** Authorization **Value :** Bearer  
VOTRE\_TOKEN\_LOCATAIRE

<b>Action</b>	<b>Méthode</b>	<b>Endpoint</b>	<b>Body</b>	<b>Notes</b>
<b>13. Ajouter aux Favoris</b>	POST	/favoris/{annonceld}	Aucun	L'annonceld doit exister et être VALIDÉE.
<b>14. Lister Mes Favoris</b>	GET	/favoris	Aucun	Liste tous les favoris du Locataire connecté.
<b>15. Supprimer un Favori</b>	DELETE	/favoris/{favorild}	Aucun	Le {favorild} est l'ID de l'objet Favoris, obtenu par l'action 14.



## **6. Administration (Actions de Modération)**

**Header :** Pour ces actions, vous devez utiliser le token de l'utilisateur **ADMIN** (ex: admin@yaoundeloc.com).

**Key :** Authorization **Value :** Bearer VOTRE\_TOKEN\_ADMIN

<b>Action</b>	<b>Méthode</b>	<b>Endpoint</b>	<b>Body</b>	<b>Notes</b>
<b>16. Lister les Annonces en Attente</b>	GET	/admin/annonces/en-attente	Aucun	Liste toutes les annonces avec StatutAnnonce.EN_ATTENTE .
<b>17. Valider une Annonce</b>	PATCH	/admin/annonces/{id}/valider	Aucun	L'id est celui de l'annonce à valider.
<b>18. Rejeter une Annonce</b>	PATCH	/admin/annonces/{id}/rejeter	Aucun	L'id est celui de l'annonce à rejeter.