

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

Fonctionnalités Clés

Domaine	Description	Rôles Impliqués
Authentification	Inscription pour Locataire/Propriétaire et Connexion (génération de token JWT).	Tous
Annonces (Public)	Consultation des annonces validées . Recherche par quartier, type et prix.	Tous (non authentifié)
Annonces (Propriétaire)	Création, modification, suppression d'une annonce. L'annonce passe en statut EN_ATTENTE après chaque création/modification.	ROLE_PROPRIETAIRE
Modération (Admin)	Lister, valider et rejeter les annonces en attente.	ROLE_ADMIN
Images	Téléchargement sécurisé d'images et service public pour les afficher.	ROLE_PROPRIETAIRE (Upload) / Tous (Consultation)
Favoris	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
1. Connexion	POST	/auth/login	{ "email": "...", "password": "..." }	admin@yaoundeloc.com, password
2. Inscription Locataire	POST	/auth/register	{ "email": "...", "password": "...", "full_name": "...", "phone": "..." }	locataire@mail.com, pass123
3. Inscription Proprio	POST	/auth/register-proprietaire	{ "email": "...", "password": "...", "full_name": "...", "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)

Action	Méthode	Endpoint	Header	Body	Notes
4. Lister Annonces Validées	GET	/annonces	Aucun	Aucun	Retourne toutes les annonces avec StatutAnnonce.VALIDEE.
5. Rechercher Avancée	GET	/annonces/search? quartier=ba st**&prixMi n=10000	Aucun	Aucun	Utilise des query params optionnels.
6. Détail Annnonce	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
7. Créer une Annonce	POST	/annonces	{ "titre": "...", "description": "...", "prix": 50000, "quartier": "...", "typeBien": "..." }	L'annonce passe en statut EN_ATTENTE.
8. Lister Mes Annonces	GET	/annonces/my-listings	Aucun	Retourne toutes les annonces (y compris EN_ATTENTE et REJETEE) du propriétaire.
9. Modifier une Annonce	PUT	/annonces/{id}	(Même JSON que la création)	L'annonce repasse en statut EN_ATTENTE pour re-modération.
10. Supprimer une Annonce	DELETE	/annonces/{id}	Aucun	Le propriétaire doit être l'auteur de l'annonce.

4. Images (Upload)

Action	Méthode	Endpoint	Header	Body	Notes
11. Uploader une Image	POST	/annonces/ {annonceId}/ upload-image	Header + Content-Type: multipart/form-data	Body : Utiliser form-data, champ key: file, value: [sélectionner un fichier].	Nécessite un token ROLE_PROPRIÉTaire ET que l'utilisateur soit propriétaire de l'annonce.
12. Afficher l'Image	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

Action	Méthode	Endpoint	Body	Notes
13. Ajouter aux Favoris	POST	/favoris/{annonceld}	Aucun	L'annonceld doit exister et être VALIDEE.
14. Lister Mes Favoris	GET	/favoris	Aucun	Liste tous les favoris du Locataire connecté.
15. Supprimer un Favori	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@yaounde.loc.com).

Key : Authorization Value : Bearer VOTRE_TOKEN_ADMIN

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