

## Arcadia Zoo Read me

### Description :

Arcadia Zoo est une application web conçue pour améliorer l'interaction des visiteurs en fournissant des informations détaillées sur les habitats des animaux et les services du zoo. L'application met l'accent sur les thèmes écologiques, reflétant l'engagement du zoo envers la durabilité.

### Aperçu du projet :

L'application offre plusieurs espaces pour les visiteurs, les administrateurs, les vétérinaires et les employés, fournissant un accès personnalisé aux informations et fonctionnalités du zoo. Les espaces utilisateurs comprennent :

- Espace Visiteur :
  - Consultation des habitats et animaux du zoo
  - Affichage des services disponibles
  - Soumission d'avis sur les animaux et services
  - Page de contact pour les demandes et questions
- Espace Administrateur :
  - Gestion des comptes (employés, vétérinaires)
  - Gestion des services, habitats et animaux
  - Tableau de bord des statistiques de consultation
- Espace Employé :
  - Validation des avis des visiteurs
  - Gestion des services du zoo
  - Suivi de l'alimentation quotidienne des animaux
- Espace Vétérinaire :
  - Gestion des comptes rendus médicaux des animaux
  - Commentaires sur les habitats
  - Suivi alimentaire des animaux

### Identifiants de Connexion

- Admin :
  - Username : ``admin``
  - Mot de passe : ``loulou25``
- Vétérinaire :
  - Username : ``veterinaire``
  - Mot de passe : ``zoovetoarcadia``

- Employé :
  - Username : ``employe``
  - Mot de passe : ``zooarcadia``

La page rapport vétérinaire est protégée par un login, seuls les personnes qui peuvent se connecter au panneau de configuration peuvent y accéder.

#### Pile Technologique:

- Frontend : HTML, CSS (Bootstrap), JavaScript
- Backend : Django (Python)
- Bases de données :
  - Relationnelle : SQLite3
  - NoSQL : MongoDB
- Contrôle de version : Git

#### Configuration Locale

Cloner le dépôt :

```
git clone https://github.com/Ameut/Arcadia_zoo.git
cd Arcadia_zoo
```

Installer les dépendances :

```
pip install -r requirements.txt
```

Configurer les bases de données :

SQLite est configurée par défaut avec Django.

Pour MongoDB, assurez-vous que MongoDB fonctionne localement ou connectez-vous à une instance MongoDB Atlas.

#### Étapes pour se Connecter avec MongoDB Compass

Téléchargez MongoDB Compass depuis le site officiel :

<https://www.mongodb.com/try/download/compass>

Installer MongoDB Compass :

Suivez le tutoriel d'installation pour Windows, Mac ou Linux :

<https://www.mongodb.com/docs/compass/current/install/>

Lancer les migrations :

```
python manage.py migrate
```

Créer un superutilisateur :

```
python manage.py createsuperuser
```

Démarrer le serveur de développement :

```
bash
```

Copy code

```
python manage.py runserver  
Accédez a http://127.0.0.1:8000
```

Pour toute question, veuillez contacter [ameur.ouafi@hotmail.fr](mailto:ameur.ouafi@hotmail.fr) ou  
[ouafiameur@gmail.com](mailto:ouafiameur@gmail.com).