# **Documentation Technique et Fonctionnelle de l'Application du Parc d'Attraction**

URBANC Ewan

[**Documentation Technique et Fonctionnelle de l'Application du Parc d'Attraction 1**](#_e0k32ffjg4ke)

[Introduction 3](#_wi2lib4j7tf4)

[Architecture de l'application 3](#_xjcr4s1em01)

[1. Technologies utilisées 3](#_cm162yn5j41n)

[2. Structure des composants 3](#_40aef3c22rhw)

[Fonctionnalités 3](#_ht2k3pw0ab32)

[1. Liste des Attractions 3](#_npa7v4xld2ni)

[2. Ajout de Critiques 4](#_3yi9detvashh)

[3. Affichage des Avis 4](#_xcth9cog1zlt)

[Schéma de Base de Données 4](#_5rucajn4b34a)

[Tables 4](#_4nenn6mz2zow)

[Attraction 4](#_k14ya0kcc5o5)

[Critique 4](#_jww9apwktka7)

[Améliorations Possibles 5](#_hjnb7ryxcgaa)

## **Introduction**

Cette documentation décrit l'application web du parc d'attractions permettant d'afficher des attractions, d'ajouter des critiques et de consulter les avis des utilisateurs. L'application utilise Angular avec Angular Material pour l'interface utilisateur, ainsi qu'un backend accessible via une API REST.

## **Architecture de l'application**

### **1. Technologies utilisées**

* **Frontend** : Angular 15
* **UI Library** : Angular Material
* **Backend API** : Flask (Python)
* **Base de données** : MySQL
* **RxJS** : Gestion des flux asynchrones

### **2. Structure des composants**

* AccueilComponent : Composant principal affichant la liste des attractions et les critiques associées.
* AttractionService : Service permettant de récupérer la liste des attractions.
* CritiqueService : Service permettant de récupérer et d'ajouter des critiques.
* DataService : Service de base effectuant les requêtes HTTP.

## **Fonctionnalités**

### **1. Liste des Attractions**

* Récupération des attractions depuis l'API REST.
* Filtrage des attractions visibles.
* Affichage des informations principales : nom, description, difficulté.

### **2. Ajout de Critiques**

* Formulaire permettant d'ajouter un commentaire avec une note.
* Vérification de la validité du formulaire avant soumission.
* Réinitialisation du formulaire après envoi.

### **3. Affichage des Avis**

* Chargement des avis associés à chaque attraction.
* Affichage de la note et du commentaire.

## **Schéma de Base de Données**

### **Tables**

#### **Attraction**

| **Colonne** | **Type** | **Description** |
| --- | --- | --- |
| attraction\_id | INT PK | Identifiant unique |
| nom | VARCHAR | Nom de l'attraction |
| description | TEXT | Description de l'attraction |
| difficulte | INT | Difficulté de l'attraction (1-5) |
| visible | BOOLEAN | Indique si l'attraction est visible |

#### **Critique**

| **Colonne** | **Type** | **Description** |
| --- | --- | --- |
| critique\_id | INT PK | Identifiant unique |
| attraction\_id | INT FK | Référence vers Attraction |
| commentaire | TEXT | Commentaire utilisateur |
| note | INT | Note (1-5) |

## 

## 

## 

## 

## **Améliorations Possibles**

| **Amélioration** | **Description** | **Estimation (jours)** |
| --- | --- | --- |
| Pagination des attractions | Charger les attractions par pages | 2 |
| Système d'authentification | Ajout des utilisateurs avec authentification | 3 |
| Modification des critiques | Permettre la mise à jour des avis | 2 |
| Ajout de la recherche | Rechercher une attraction par nom | 1 |
| Interface responsive | Adapter l'UI pour mobile et tablette | 2 |