

Projet JEE – Rapport

Équipe Tonnerre de Zeus

Laszlo ABADIE

Adem BENKEDDAD

Samy CHAABI

Sami TAKI

Louis THOMAS-GIRARDEY

7 mai 2025



Table des matières

1	Structure du projet	3
1.1	Arborescence	3
1.2	API	3
2	Choix d'implémentation	4
3	État du projet	4

Table des figures

1	Arborescence du projet (fichiers java)	3
---	--	---

Liste des tableaux

List of Algorithms

1 Structure du projet

1.1 Arborescence

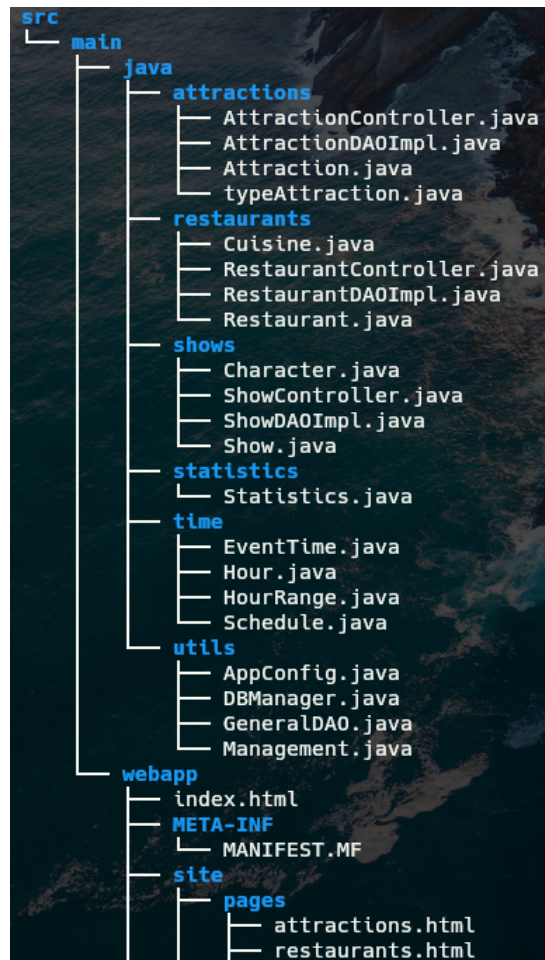


FIGURE 1 – Arborescence du projet (fichiers java)

Le projet est découpé en plusieurs packages java.

1.2 API

Nous avons choisi d'implémenter une API REST suivante, avec trois points d'entrée :

- /api/attractions;
- /api/restaurants;
- /api/shows;

Avec par exemple pour les attractions :

1	GET /api/attractions	- Retreives a list of all attractions
2	GET /api/attractions/{id}	- Retreives data for the attraction of id 'id'
3	POST /api/attractions	- Creates a new attraction
4	PUT /api/attractions/{id}	- Updates an existing attraction
5	DELETE /api/attractions/{id}	- Deletes an existing attraction

2 Choix d'implémentation

Il y a trois classes principales : **Attraction**, **Restaurant** et **Show**. Ces classes représentent respectivement les attractions, restaurants et les spectacles.

En plus de ces classes, on a pour chacune une classe **DAOImpl**, qui contient des méthodes permettant d'échanger avec la base de données. Il y a également pour chaque classe une classe **Controller** qui permet de créer les endpoints.

Il y a une interface **GeneralDAO** qui est paramétrée par un type (qui sera une des classes), qui est implémentée par les **DAOImpl**. Elle contient aussi des méthodes statiques.

Nous avons choisi de d'implémenter la logique (*i.e* les vérifications d'horaires, de chevauchement, etc.) dans des triggers sur la base de donnée.

Pour le frontend, nous avons factorisé le code javascript de génération des pages HTML, comme elles sont très similaires.

3 État du projet

Nous avons un backend fonctionnel : exposition de l'API, communication entre le java et le SQL, les triggers.

Le frontend est partiellement fonctionnel, mais ne communique pas bien avec le backend.