

## Etude de cas :

# Application web Angular de gestion d'événements

### 1. Contexte du projet

Le projet consiste à développer une application web Angular de gestion d'événements et de billetterie en ligne avec un système de feedback.

Le rôle de l'étudiant est celui d'un développeur frontend :

- Le backend (API REST) sera fourni et déjà opérationnel.
- La mission est de construire l'interface utilisateur (Angular 18, architecture modulaire) et d'intégrer les fonctionnalités en consommant cette API.

### 2. Etude fonctionnel du projet

#### 2.1. Acteurs

- Utilisateur : consulte les événements, réserve/achète des tickets, laisse un feedback.
- Organisateur : crée et gère des événements, suit les tickets vendus, consulte les retours.
- Administrateur : supervise la plateforme, gère utilisateurs et événements.

#### 2.2. Besoins fonctionnels

- Événements : consultation, création/édition (organisateur), suppression.
- Tickets : réservation, achat, historique des tickets.
- Utilisateurs : inscription, connexion, gestion profil.
- Feedback : soumettre un avis/note, consulter les feedbacks.

#### 2.3. Besoins non-fonctionnels

- Application responsive (desktop/mobile).
- Navigation fluide et intuitive.
- Sécurité (authentification JWT, autorisations).
- Bonne performance et clarté du code (structure modulaire).

#### 2.4. Cas d'utilisation (côté frontend)

- ✓ L'utilisateur consulte la liste des événements.
- ✓ L'utilisateur consulte les détails d'un événement.
- ✓ L'utilisateur réserve/achète un ticket.
- ✓ L'organisateur crée/modifie un événement.
- ✓ L'utilisateur laisse un feedback sur un événement.
- ✓ L'administrateur supprime ou valide un événement.

## 2.5. La couche modèle

Entités principales :

- Utilisateur (id, nom, email, rôle, mot de passe, image)
- Événement (id, titre, description, date, lieu, prix, organisateurId, image)
- Ticket (id, userId, eventId, dateAchat, statut)
- Feedback (id, userId, eventId, note, commentaire)

Relations :

- Un utilisateur peut réserver plusieurs tickets.
- Un utilisateur peut laisser plusieurs feedbacks.
- Un organisateur crée plusieurs événements.