## **Liste des étapes**

### **1. Ajout d’un Événement**

1. **Conception de l’interface**
   * Ajouter un **bouton** "Créer un événement" sur la page d’affichage des événements.
   * Au clic, ouvrir un **pop-up** ou un **side panel** permettant de saisir les informations de l’événement (nom, date, lieu, description, etc.).
2. **Création du formulaire**
   * Champs nécessaires : titre de l’événement, date, heure, lieu, description, image éventuelle, etc.
   * Penser à la **validation** des champs (par exemple, champ requis, format de date, etc.).
3. **Envoi des données**
   * Au **submit** du formulaire, envoyer les informations à une **route API** (ex. POST /api/events) qui s’occupera de l’enregistrement dans la base de données.
   * Mettre à jour la liste des événements affichés après la création réussie (rechargement ou mise à jour via un état React/Next.js).
4. **Retour utilisateur**
   * Afficher un message de succès ou d’erreur en fonction de la réponse du serveur (ex. "Événement créé avec succès !").

### **2. Détails de l’Événement**

1. **Création de la page "Event Details"**

Dans Next.js, créer un composant/page qui s’appuie sur une route dynamique, par exemple:  
  
pages/events/[id].js

* + Cette page doit récupérer l’identifiant de l’événement depuis l’URL et faire un appel à l’API (ex. GET /api/events/[id]) pour obtenir les détails de l’événement.

1. **Contenu de la page de détails**
   * Afficher le **titre**, la **date**, le **lieu**, la **description** et éventuellement d’autres informations pertinentes de l’événement (visuel, liste des intervenants, etc.).
   * Prévoir un **bouton** ou un **lien** "Réserver cet événement" qui mènera vers la page de réservation.

### **3. Page de Réservation**

1. **Accès à la page de réservation**
   * Depuis la page de détails de l’événement, un clic sur "Réserver cet événement" doit rediriger vers la page de réservation (ex. pages/events/[id]/reserve.js ou pages/reservation/[eventId].js).
2. **Formulaire de réservation**
   * Champs possibles : nom, prénom, email, nombre de places, moyen de paiement, etc.
   * Possibilité d’inclure une section récapitulative de l’événement (titre, date, etc.) afin de contextualiser la réservation.
3. **Validation et envoi des données**
   * Au **submit** du formulaire, envoyer la requête à la route API de réservation (ex. POST /api/reservations).
   * Gérer la **validation** (email valide, nombre de places disponible, etc.).
4. **Feedback utilisateur**
   * Une fois la réservation effectuée, afficher un message de confirmation ("Votre réservation a bien été enregistrée !") et éventuellement un numéro de confirmation.

### **3.4. Création des Routes API**

Pour la bonne gestion des événements et des réservations, nous aurons besoin de **routes API** dans Next.js. Voici les principales :

1. **POST /api/events**
   * Créer un nouvel événement dans la base de données.
   * Corps de la requête : objet contenant les informations de l’événement.
2. **GET /api/events**
   * Récupérer la liste de tous les événements (utilisé pour la page principale d’affichage).
3. **GET /api/events/[id]**
   * Récupérer les détails d’un événement spécifique (utilisé pour la page de détails).
4. **POST /api/reservations**
   * Créer une nouvelle réservation pour un événement donné.
   * Corps de la requête : informations du réservataire (nom, email, nombre de places, etc.) et l’ID de l’événement associé.
5. (Optionnel) **GET /api/reservations**
   * Permettre, si nécessaire, de lister toutes les réservations ou de filtrer par événement.