

Projet - Paris Events

Développement Web - Brief

Objectif

Vous devez réaliser une application web permettant de rechercher, explorer et organiser des **événements Parisiens** en vous aidant de l’API Open Data Paris, qui fournit des données en temps réels.

Votre application sera constituée d’une fonctionnalité de recherche d’événements, et offrira une gestion de sauvegardes.

Les sauvegardes d’événéments seront stockées directement dans le navigateur (via l’API « localStorage »).

Le service (API) Open Data Paris

[](https://opendata.paris.fr/explore)

Ce service gratuit offre la possibilité aux développeurs de travailler avec plusieurs jeux de données, qu’ils appellent des “**dataset**”.

De très nombreux jeux de données sont accessibles et classés selon différentes thématiques. Il est possible par exemple de connaître la [disponibilité en temps réel des Vélibs’ Parisien](https://opendata.paris.fr/explore/dataset/velib-disponibilite-en-temps-reel/), ou encore de connaître les [emplacements de stationnement sur la voie publique](https://opendata.paris.fr/explore/dataset/stationnement-voie-publique-emplacements/).

Le jeu de données avec lequel vous travaillerez pour ce sujet permet de rechercher les prochains événements référencés sur le territoire Parisien.

Il se nomme **“**[**Que Faire à Paris ? - Evènements et activités**](https://opendata.paris.fr/explore/dataset/que-faire-a-paris-/api/)**”**

Dans un premier temps, allez explorer ce jeu de données et analysez son fonctionnement.



Vous noterez que l’API existe en [**V2 (beta)**](https://opendata.paris.fr/api/v2/console).  
Pour l’instant, le site met en avant la V1, mais celle-ci ne permet pas de retrouver un événement spécifique via son ID.  
  
Vous devrez donc utiliser **la V2 plutôt que la V1 pour ce projet**!

### Rechercher des événements :

[https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records/?**search**=danse&**sort**=title&**limit**=15](https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records/?search=danse&sort=title&limit=15)

Cette URL va renvoyer **15** résultats correspondants aux événements relatifs au terme “**danse**”, triés par titre (champs “**title**”)

### Récupérer un événement via son ID

[https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records/**08de0bed488e150320eeef8e86b3e5502e33cd2b**](https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records/08de0bed488e150320eeef8e86b3e5502e33cd2b)

Cette URL va renvoyer l’événement qui correspond à l’ID “**08de0bed488e150320eeef8e86b3e5502e33cd2b**” (ici par exemple, un événement sur le parkour)

L’application

Le projet que vous devez réaliser est une **application web utilisant les informations du jeu de données des événements à Paris**. Votre application doit être constituée de 4pages au total :

* Une page d’accueil
* Une page de recherche d’événements par titre
* Une page pour afficher un événement en détails
* Une page listant les événements sauvegardés de l’utilisateur

# Détails et instructions pour chaque page

Les visuels qui suivent sont présents à des fins d’illustration. **Il ne s’agit pas de maquettes à intégrer**. Vous restez maître de l’aspect visuel, de l’agencement des informations et de l’UX (experience utilisateur) de votre interface finale !

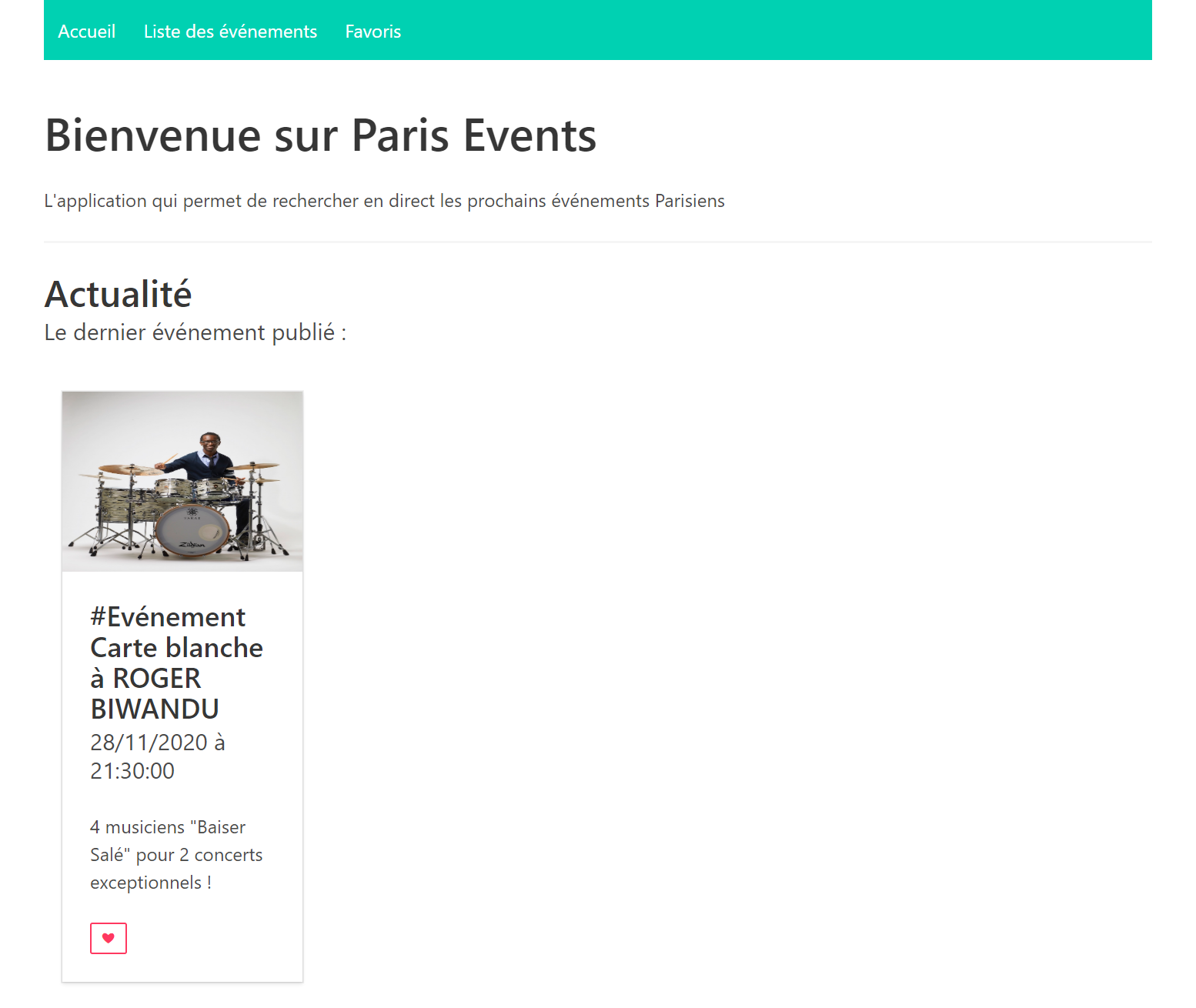
Vous devez faire preuve de bon sens et de créativité pour présenter au mieux les informations sur chaque page.

*NB : Ces visuels sont également disponible au format PNG, et joints avec ce brief*

1. Page d’accueil

Une simple page de bienvenue sur l'application, qui affichera au visiteur le dernier événement ajouté sur l’API.

*Capture d’écran :*



1. Page de recherche

La page doit être composée d’un formulaire avec 1 champs de recherche par titre

Ce paramètre peut être passé à l’API OpenData de la façon suivante : [https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records?**search**=theatre](https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records?search=theatre)

Le paramètre « **?search=** » correspond à la recherche. L’API renverra tout ce qui correspond à ce terme de recherche.

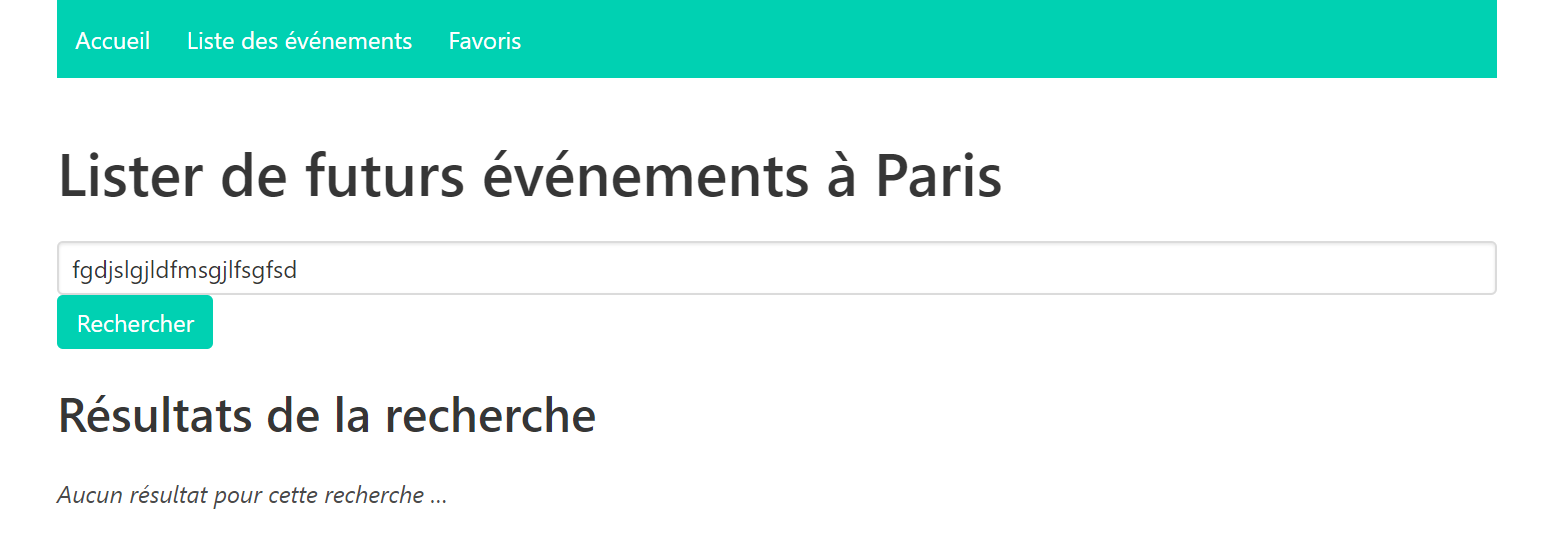
**Plus de détails sur la documentation :** [**https://opendata.paris.fr/api/v2/console#!/dataset/getRecords**](https://opendata.paris.fr/api/v2/console#!/dataset/getRecords)

Les résultats de la recherche doivent être présentés sous forme de liste d’encarts et afficher pour chaque encart les éléments suivants :

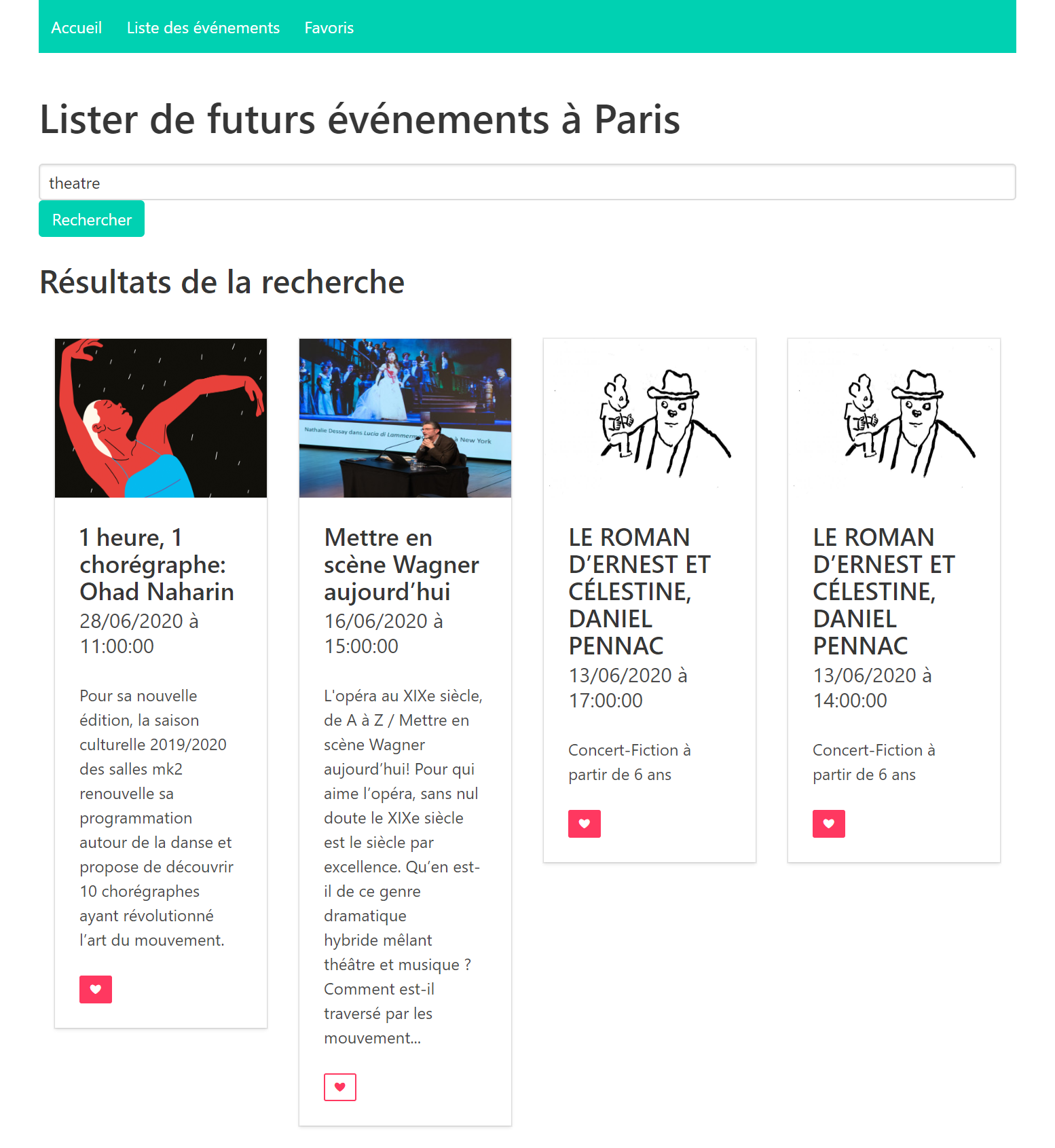
* Photo de l’événement
* Titre de l’événement
* Date de début de l’événement
* Une courte description de l’événement
* Un bouton permettant de **sauvegarder** ou **retirer** l’élément de la sauvegarde

**Si la recherche de l'utilisateur ne donne aucun résultat sur l'API de Deezer, un message devra s'afficher à la place.**

**Chaque événement listé doit être cliquable et doit rediriger vers une page détaillant l’événement en question.**

****

*Avec résultats :*

****

1. Page d’affichage d’un événément en détails

Cette page devra permettre d’afficher en détail un événement sur lequel on aura cliqué.

L’identifiant de l’événement sera transmi via l’URL de la page. En récupérant cet identifiant en JavaScript, il sera possible d’émettre une requête vers l’API OpenData de la façon suivante :

<https://opendata.paris.fr/api/v2/catalog/datasets/que-faire-a-paris-/records/1ea9c9dbfdcc44df66c941a81a989c88688d2c4e> (où **1ea9c9dbfdcc44df66c941a81a989c88688d2c4e** est l’ID de l’événement)

**Plus de détails sur la documentation :** [**https://opendata.paris.fr/api/v2/console#!/dataset/getRecord**](https://opendata.paris.fr/api/v2/console#!/dataset/getRecord)

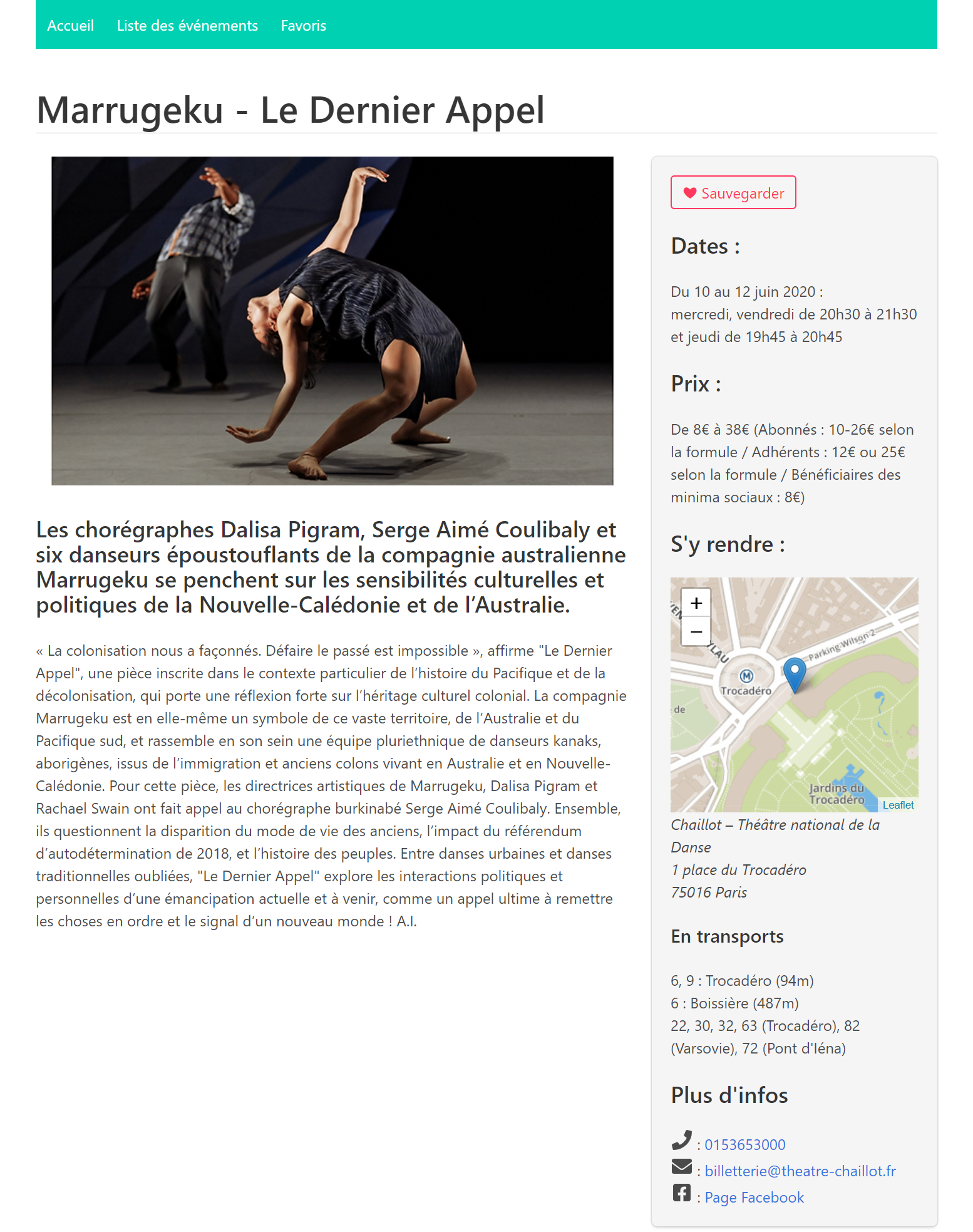
**Si l’identifiant de l’événement en URL est invalide ou inexistant, on est redirigé vers la page d’accueil**

L’événement doit s’afficher en intégralité dans la page avec au minimum les détails suivants :

* Photo de l’événement
* Titre de l’événement
* Description complète de l’événement
* Dates de l’événement
* Prix de l’événement
* Lieu de l’événement (adresse postale)
* Accès à l’événement en transports (si précisé dans les données)
* Informations complémentaires comme le numéro de téléphone et/ou la page facebook de l’événement (si précisé dans les données)
* Un bouton permettant de **sauvegarder** ou **retirer** l’élément de la sauvegarde

Livre à vous de personnaliser et adapter la mise en page, et d’exploiter librement toute les informations mises à votre disposition par l’API.

*Par exemple, si vous en avez le temps, pourquoi ne pas afficher une map Google ou Leaflet avec les coordonnées de latitude et longitude (en bonus)*



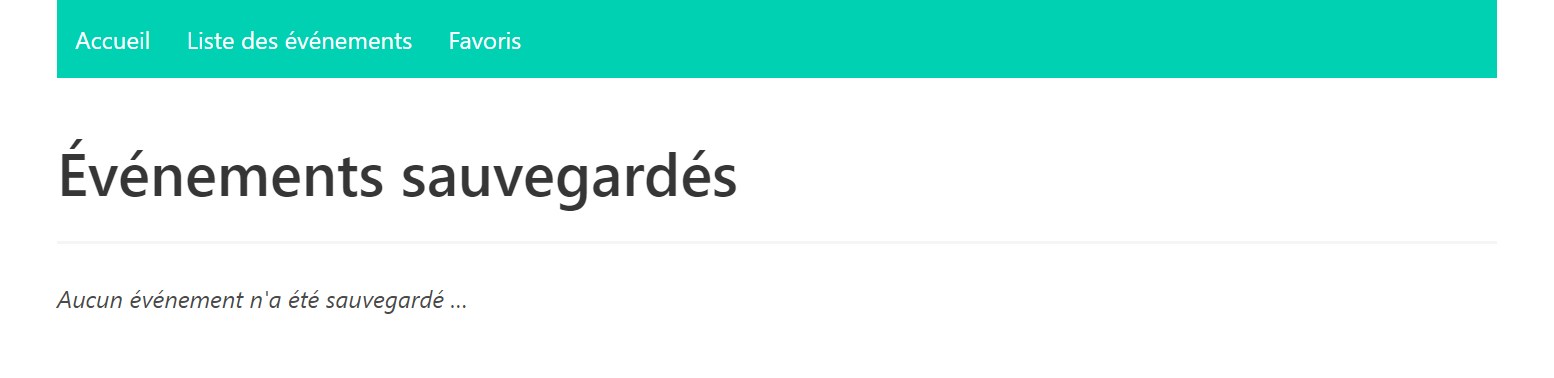
1. Page des événements sauvegardés

Une page listant les événements que le visiteur a sauvegardés durant son utilisation de l'application.

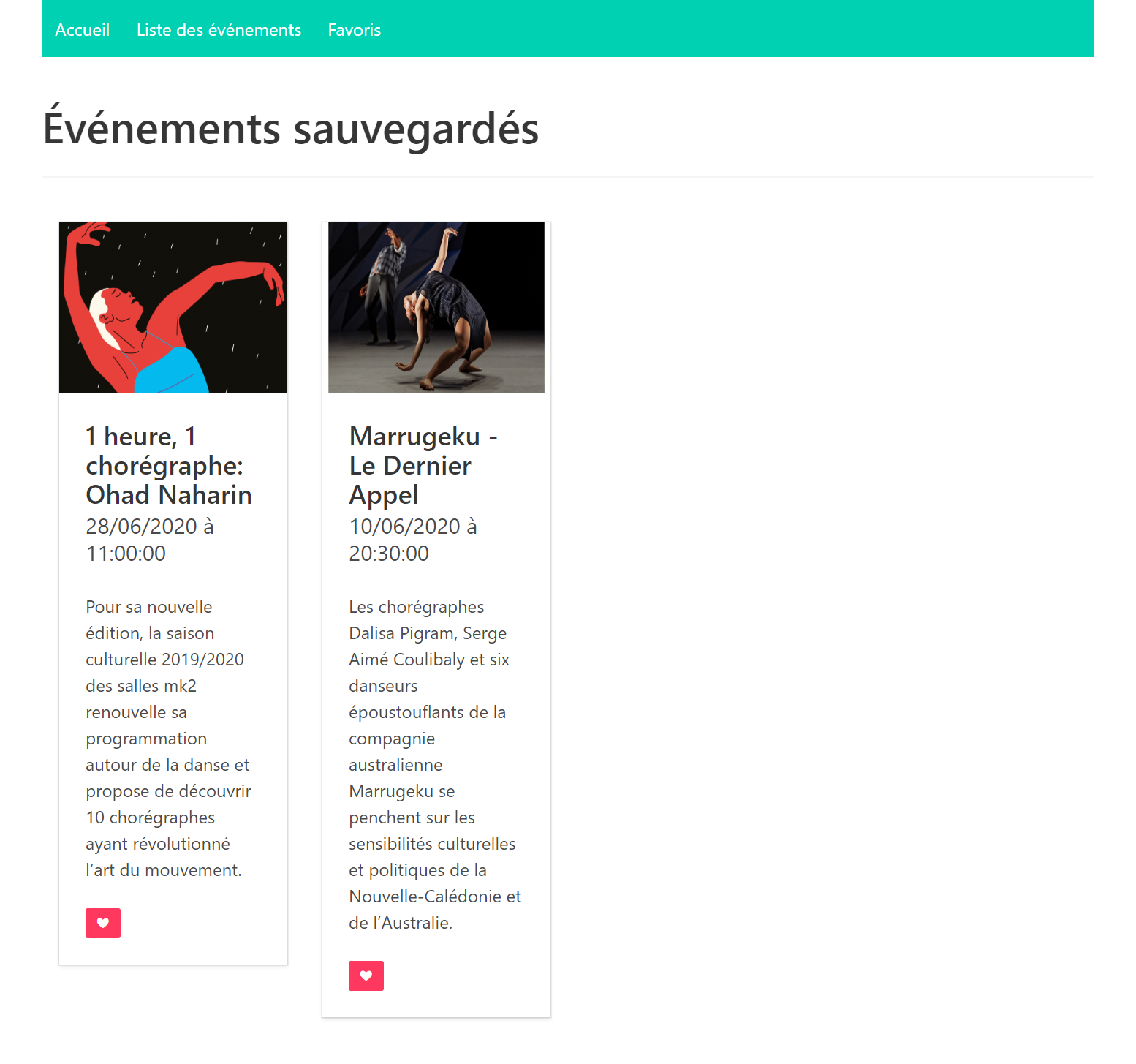
Il doit être possible depuis cette page de retirer un événement sauvegardé.

Les données doivent être stockées dans le **localStorage** du navigateur web.

*Un rappel vers le lien d’Alsacréations expliquant en détails le fonctionnement de l’API Web Storage :* [*https://www.alsacreations.com/article/lire/1402-web-storage-localstorage-sessionstorage.html*](https://www.alsacreations.com/article/lire/1402-web-storage-localstorage-sessionstorage.html)



*Avec des sauvegardes :*



Démo

Une démo de l’application est disponible [en cliquant ici](https://jmpp.github.io/paris-events-proto/)

Modalités du projet

Ce projet est à réaliser sur une période d’une semaine, en centre de formation, en présence d’un intervenant assurant le suivi technique du projet.

* Vous devez **obligatoirement être présent** (ou connecté si à distance) durant les heures de suivi.
* La communication et l’échange entre les étudiants est encouragé, cependant le travail à fournir reste **individuel**.

Technologies

Vous devez utiliser vos connaissances en HTML, CSS, JavaScript et ReactJS pour réaliser ce projet.

L’utilisation de la bibliothèque ReactJS est obligatoire.

Versionning

Vous utiliserez Git pour versionner votre projet, et ferez des commits cohérents et réguliers pour sauvegarder votre travail.

Rendus et mise en ligne

Le rendu s’effectue le Dimanche (deadline : 23h59) qui suit la fin de semaine du projet. Le projet sera à déposer sur un lien Google Drive qui vous sera fourni en amont.

**Note importante : Quand vous déposerez votre archive sur un Google Drive, merci de NE PAS INCLURE LE DOSSIER /NODE\_MODULES/ dans votre archive ! Assurez-vous en revanche d’avoir le fichier `package.json` à jour.**

**Une archive ZIP contenant un dossier /node\_modules/ ne sera pas corrigée et la note attribuée sera de 0 (soit, projet non acquis)**

Critères de notation

Seront pris en compte les critères suivant pour la note du projet :

* Application fonctionnelle (respect des fonctionnalités demandés sur chacune des 4 pages)
* Maîtrise de la bibliothèque ReactJS
* Utilisation conforme d’AJAX et de l’API OpenData
* Ergonomie et qualité de l’interface
* Qualité du code HTML, CSS et JavaScript (respect des normes W3C)

Points bonus :

Des points bonus pourront être attribués à la discrètion du formateur sur les critères suivants:

* Prise d’initiative sur l’amélioration des fonctionnalités demandées.
* Ajout d’animations améliorant l’ergonomie et l’interface.
* Gestion des erreurs (par ex, dans le cas où l’application n’a plus de réseau)

De la même manière, des points pourront être retirés en cas de constat de triche évidente, de non-présence durant les heures de cours, ou de mauvais comportement.