Documentation v0.3 – Gestion des demandes et les recherche optimisés

Sommaire

Introduction

La version v0.3 marque une avancée majeure pour l’application en intégrant un **système de notifications en temps réel**, une **gestion fluide des demandes d’intérêt**, et une **expérience utilisateur optimisée**.

Cette documentation couvre :

1. Les nouvelles fonctionnalités développées.
2. Les **flux utilisateurs** pour chaque action clé.
3. Les **endpoints backend** utilisés.
4. L’**impact métier** et la valeur ajoutée des améliorations.
5. Un **diagramme de séquence UML** pour illustrer le workflow.

🚀 Grandes Fonctionnalités

🔔 2.1 Gestion des Demandes d'Intérêt & Notifications en Temps Réel

Description

Cette fonctionnalité regroupe tout le flux des demandes d'intérêt, y compris les notifications en temps réel. Les utilisateurs peuvent envoyer et recevoir des demandes d'intérêt, puis être notifiés de l'acceptation ou du refus de la demande en temps réel.

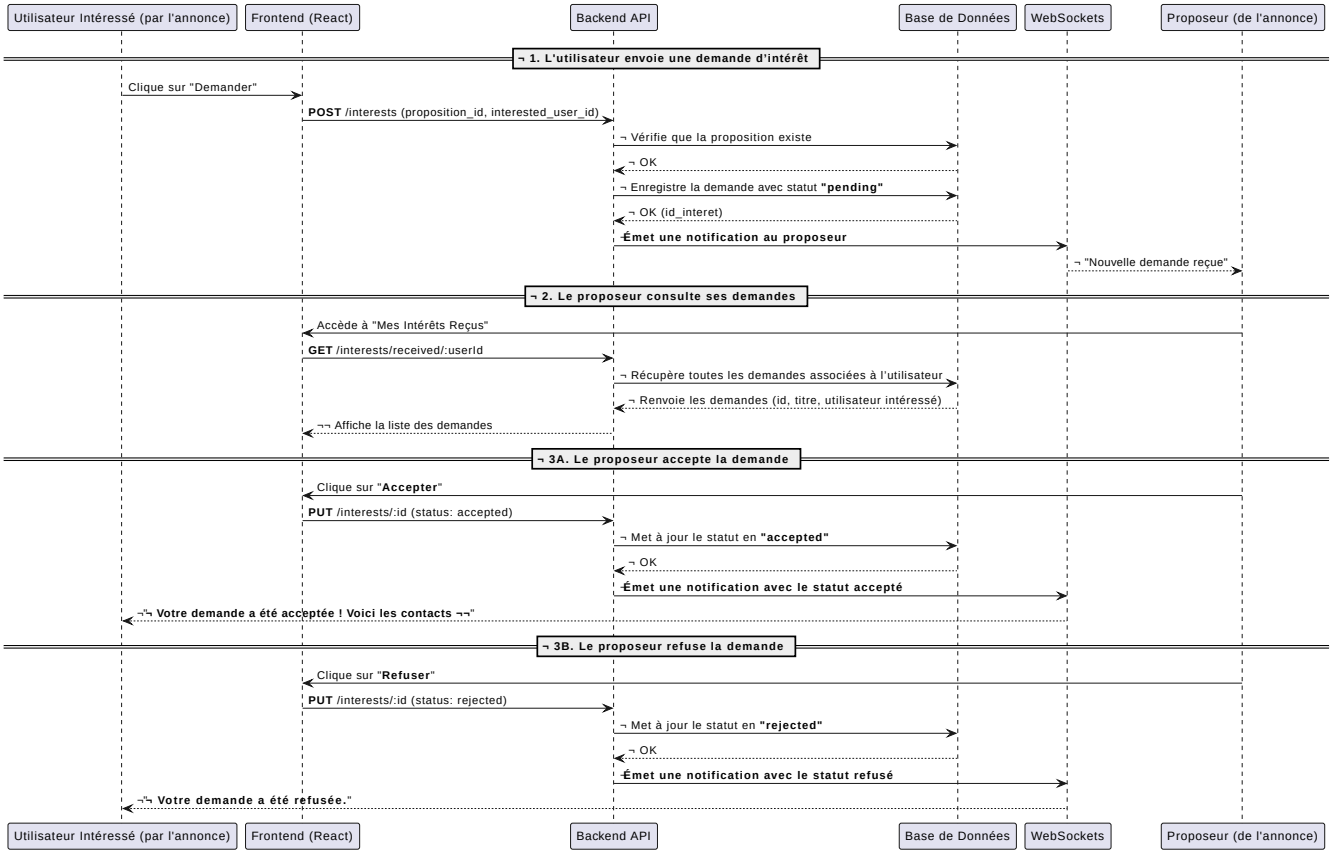
Flux Utilisateur

1. **Envoi d'une Demande d'Intérêt**
   - L'utilisateur intéressé clique sur "Demander".
   - La demande est enregistrée en base (via **POST /interests**).
   - Une notification en temps réel est envoyée au proposeur.
2. **Consultation des Demandes d'Intérêt**
   - Le proposeur accède à "Mes Intérêts Reçus".
   - Il voit la demande et peut l'accepter ou la refuser.
3. **Notifications et Actions en Temps Réel**
   - Si la demande est acceptée, le demandeur est notifié avec les coordonnées du proposeur.
   - Si la demande est refusée, le demandeur est informé de la décision.

Endpoints Backend

Méthode	Endpoint	Description
POST	`/interests`	Créer une demande d'intérêt.
GET	`/interests/received/:userId`	Récupérer les demandes reçues.
GET	`/interests/sent/:userId`	Récupérer les demandes envoyées.
PUT	`/interests/:id`	Accepter ou refuser une demande.
GET	`/notifications/:userId`	Récupère toutes les notifications d'un utilisateur.
POST	`/notifications`	Crée une nouvelle notification.
DELETE	`/notifications/:notifId`	Supprime une notification spécifique.
DELETE	`/notifications/all/:userId`	Supprime toutes les notifications d'un utilisateur.

Diagramme de Séquence : Demande d'Intérêt et Notifications



2.2 Recherche de Propositions

Description

Cette fonctionnalité permet aux utilisateurs de rechercher des propositions en fonction de plusieurs critères : mots-clés, catégorie et distance géographique.

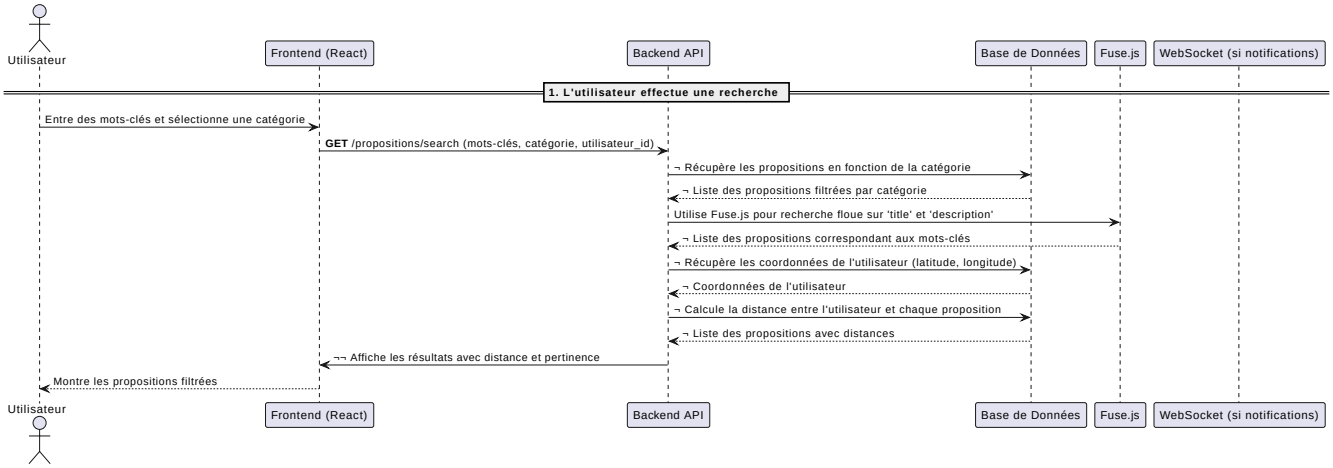
Flux Utilisateur

1. L'utilisateur entre des mots-clés et sélectionne une catégorie de service.
2. Le système effectue une recherche floue sur les titres et descriptions des propositions.
3. Le système filtre les propositions par catégorie sélectionnée.
4. Le système calcule la distance géographique entre l'utilisateur et les propositions.
5. Les résultats sont affichés, triés par proximité géographique.

Endpoints Backend

Méthode	Endpoint	Description
GET	`/propositions/search`	Recherche des propositions en fonction des mots-clés, catégorie et distance.

Diagramme de Séquence : Recherche de Propositions



🔍 2.3 Recherche Avancée des Événements

Description

Cette fonctionnalité permet aux utilisateurs de rechercher des événements en fonction de plusieurs critères : mots-clés, catégorie et ville. Grâce à la bibliothèque **Fuse.js**, la recherche est floue et permet de retrouver des événements qui correspondent partiellement aux mots-clés recherchés, même en cas d'erreur de frappe.

Le processus de recherche est optimisé pour une expérience utilisateur fluide :

1. L'utilisateur saisit un mot-clé (et optionnellement, sélectionne une catégorie ou une ville).
2. Le système filtre les événements en fonction de la catégorie et de la ville sélectionnées.
3. La recherche floue est effectuée sur les titres et descriptions des événements en utilisant Fuse.js, avec un seuil de pertinence réglable pour affiner les résultats.
4. Les résultats sont retournés et triés par pertinence.

Flux Utilisateur

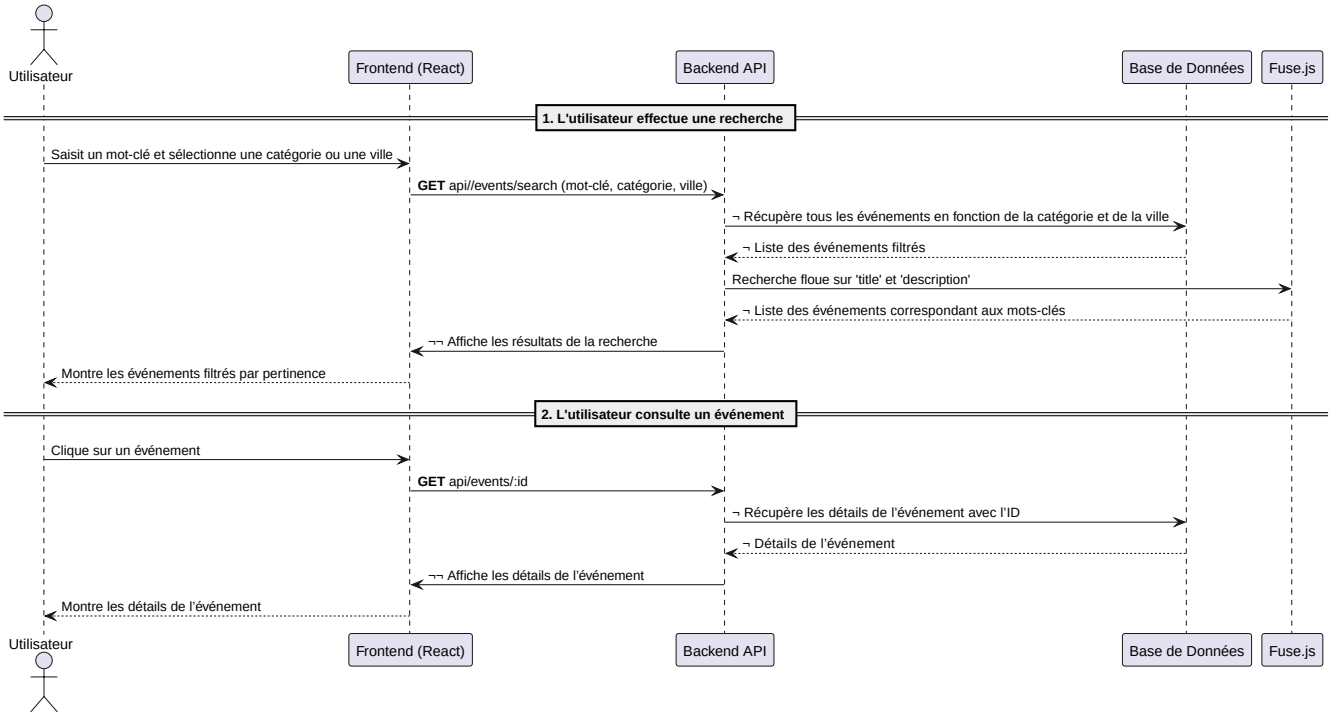
1. L'utilisateur entre un mot-clé de recherche et, si souhaité, sélectionne une catégorie et/ou une ville.
2. La recherche floue est effectuée dans les titres et descriptions des événements.
3. Les événements sont filtrés en fonction de la catégorie et de la ville, si spécifiés.
4. Les résultats de recherche sont retournés, affichés par pertinence.
5. L'utilisateur peut cliquer sur un événement pour consulter son détail.

Endpoints Backend

Méthode	Endpoint	Description
GET	`api/events/search`	Recherche des événements en fonction des mots-clés, catégorie et ville.

GET	`api/events/:id`	Récupère les détails d'un événement spécifique.

Diagramme de Séquence : Recherche Avancée des Événements



🔹 2.4 Modification et Suppression des Événements

Description

Les utilisateurs peuvent désormais **modifier** ou **supprimer** leurs événements à partir de l'interface. Cela permet une gestion complète des événements, incluant l'actualisation ou la suppression de données obsolètes.

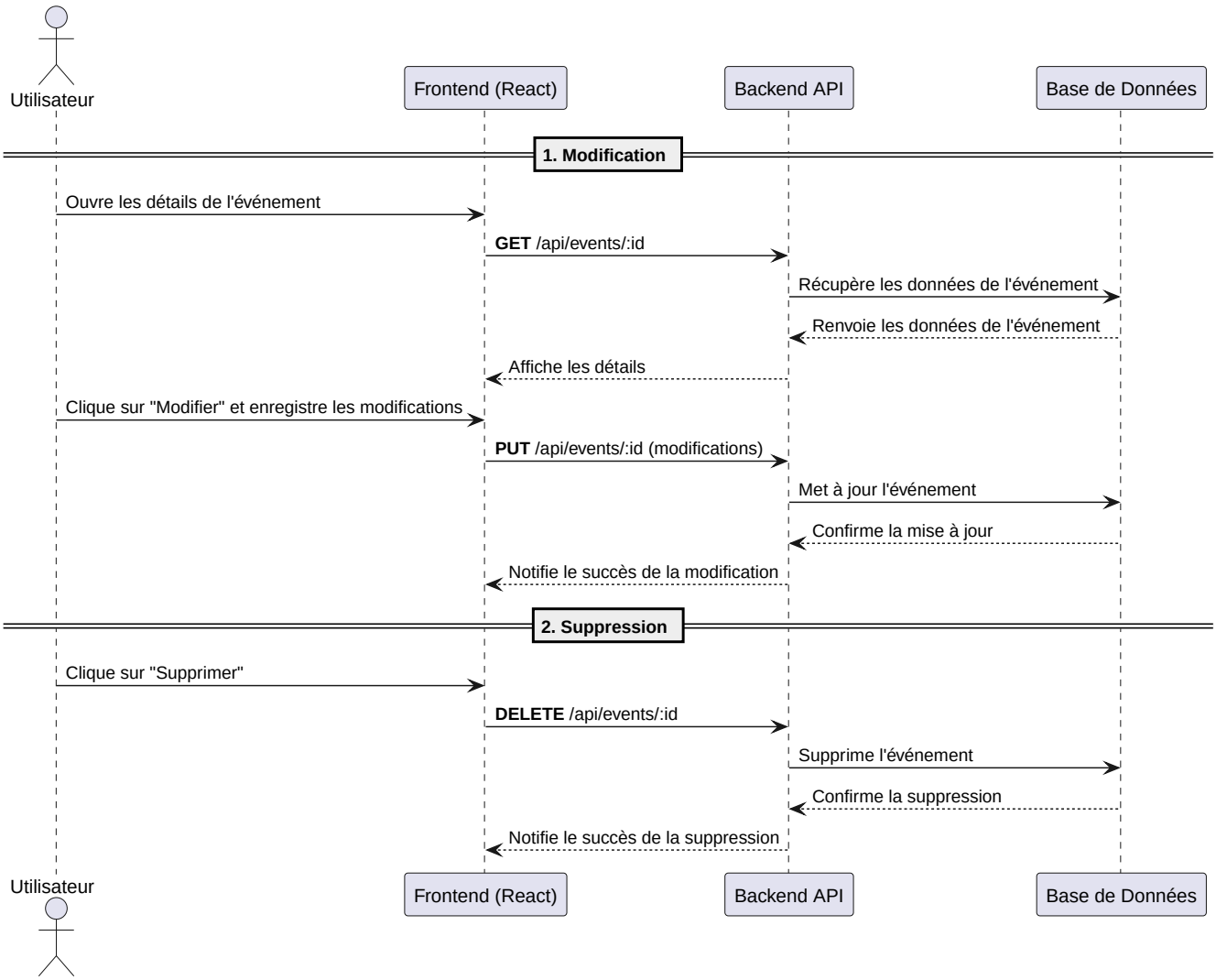
Flux Utilisateur

1. **Modification**
   - L'utilisateur ouvre les détails de son événement.
   - Il clique sur le bouton "**Modifier**".
   - Un formulaire pré-rempli s'affiche avec les informations actuelles.
   - Après modification, il clique sur "**Enregistrer**" pour sauvegarder les modifications.
2. **Suppression**
   - L'utilisateur ouvre les détails de son événement.
   - Il clique sur le bouton "**Supprimer**".
   - Une confirmation s'affiche avant suppression définitive.

Endpoints Backend

Méthode	Endpoint	Description
PUT	`/api/events/:id`	Met à jour un événement existant.
DELETE	`/api/events/:id`	Supprime un événement spécifique.

Diagramme de Séquence : Modification et Suppression des Événements



🔹 Petites Fonctionnalités

🔹 2.1 Page “Mes Demandes Envoyées” 🔹

Description

Ajout d’une nouvelle section permettant aux utilisateurs de **suivre leurs demandes** et voir si elles sont **acceptées ou refusées**.

Flux Utilisateur

1. L'utilisateur consulte **la section "Mes demandes envoyées"**.
2. Il voit **toutes ses demandes** avec leur statut actuel.
3. **Si la demande est acceptée**, il accède aux **coordonnées du proposeur**.

Endpoints Backend

Méthode	Endpoint	Description
GET	`/interests/sent/:userId`	Retourne les demandes envoyées par l'utilisateur.
PUT	`/interests/:id`	Met à jour le statut d'une demande.

🔔 2.2 Notifications en Temps Réel 🔔

Description

Les notifications sont envoyées en temps réel à l'utilisateur lorsqu'une action importante se produit (acceptation/refus d'une demande, etc.). Cela permet une interaction fluide et réactive avec l'application.

Flux Utilisateur

1. L'utilisateur effectue une action qui génère une notification.
2. Une notification apparaît instantanément dans le panneau des notifications.
3. L'utilisateur peut la consulter et la supprimer.

Endpoints Backend

Méthode	Endpoint	Description
POST	`/notifications`	Crée une nouvelle notification.
GET	`/notifications/:userId`	Récupère toutes les notifications d'un utilisateur.
DELETE	`/notifications/:notifId`	Supprime une notification spécifique.
DELETE	`/notifications/all/:userId`	Supprime toutes les notifications d'un utilisateur.

🔍 2.3 Détails d'un Événement

Description

Les utilisateurs peuvent désormais visualiser les détails d'un événement. Cette page affiche les informations complètes de l'événement sélectionné, comme son titre, sa description, sa date, son lieu, sa catégorie, et son image associée.

Flux Utilisateur

1. L'utilisateur clique sur un événement dans la liste des événements.
2. Une fenêtre modale s'affiche, contenant les détails complets de l'événement.

Endpoints Backend

Méthode	Endpoint	Description
GET	`/api/events/:id`	Récupère les détails d'un événement spécifique.

2.4 Gestion des Images des Événements (Frontend)

Description

La prise en charge des images d'événements a été ajoutée dans : - Le formulaire de création et de modification des événements. - La page de détails des événements.

Les utilisateurs peuvent visualiser une image par défaut (si aucune image n'est fournie) ou une image personnalisée associée à l'événement.

Flux Utilisateur

1. Lors de la création ou modification d'un événement, l'utilisateur peut spécifier l'URL d'une image.
2. Si l'utilisateur ne renseigne pas d'image, une image par défaut est utilisée.
3. La page de détails affiche l'image associée à l'événement.

Endpoints Backend

Méthode	Endpoint	Description
GET	`/api/events/:id`	Récupère les détails de l'événement, y compris l'URL de l'image.
POST	`/api/events`	Permet de créer un événement avec une image associée.
PUT	`/api/events/:id`	Permet de modifier l'image associée à un événement.
GET	`/api/validate-image`	Permet de vérifier si une URL d'image est valide.

2.5 Filtre par Villes

Description

Un filtre par villes a été ajouté pour permettre aux utilisateurs de rechercher des événements en fonction de leur localisation.

Flux Utilisateur

1. L'utilisateur sélectionne une ville dans la liste déroulante des filtres.
2. Les événements affichés sont automatiquement filtrés pour correspondre à la ville sélectionnée.

Endpoints Backend

Méthode	Endpoint	Description
GET	`/cities`	Récupère les villes disponibles pour les événements.

Note : Les filtres sont appliqués côté frontend en combinant les critères de recherche pour offrir une expérience utilisateur optimale.

2.6 Améliorations UX/UI

L'application a été **remaniée graphiquement** pour une **meilleure expérience utilisateur** :

- **Nouvelle navbar fixe** avec **navigation fluide**.
- **Popup de notifications stylée** avec **mise en forme propre**.
- **Suppression du bleu flashy** et **adoption d'un design plus épuré**.
- **Animations CSS** pour un rendu **plus dynamique**.
- **Espacement et marges ajustés** pour **une meilleure lisibilité**.

Impact Métier & Valeur Ajoutée

Fonctionnalité	Valeur Ajoutée
Notifications en temps réel	Permet aux utilisateurs d'être informés instantanément des actions importantes.
Gestion des demandes d'intérêt	Simplifie l'interaction entre utilisateurs, rendant le processus plus intuitif.
Suivi des demandes envoyées	Apporte de la transparence sur l'état des interactions.

Expérience utilisateur améliorée	Favorise l’adoption de la plateforme grâce à une interface plus intuitive et agréable.
Recherche avancée des événements	Permet une recherche rapide et précise des événements grâce à la recherche floue, même avec des erreurs typographiques.

- **Notifications en temps réel** : Fonctionnent sans latence.
- **Gestion des statuts (pending, accepted, rejected)** : Bien mise à jour en base.
- **UI et UX fluides** : Interface réactive et intuitive.