



Diagnostic GPS - AccessiLieux

Problème identifié

Vous avez mentionné un problème avec la prise en charge des données GPS et l'affichage des lieux proches. J'ai créé plusieurs outils pour identifier précisément où se situe le problème.



Résultats du diagnostic initial



Backend fonctionnel

- L'API `/api/places` traite correctement les coordonnées GPS
- Le calcul de distance (formule de Haversine) fonctionne parfaitement
- Le tri par distance et filtrage par rayon sont opérationnels
- Test avec coordonnées de Paris : 7 lieux trouvés dans un rayon de 50km



À vérifier côté Frontend

Le problème semble venir de la façon dont les données GPS sont:

- Récupérées depuis le navigateur
- Transmises aux composants React
- Traitées dans l'interface utilisateur



Outils de diagnostic créés

1. Page de diagnostic interactive

URL: `http://localhost:3000/debug-gps`

Cette page permet de :

- ☒ Vérifier le support de la géolocalisation
- ☒ Tester les permissions GPS
- ☒ Récupérer les coordonnées réelles
- ☒ Tester l'API avec ces coordonnées
- ☒ Afficher les résultats en temps réel

2. Script de diagnostic backend

Fichier: `/scripts/debug-gps.ts`

Commande : `npx tsx scripts/debug-gps.ts`

Ce script teste :

- Les données GPS en base
- Le calcul de distance
- L'API places avec géolocalisation

3. Script de diagnostic frontend

Fichier: `/scripts/test-frontend-gps.js`

À copier/coller dans la console du navigateur (F12) pour :

- Tester la géolocalisation native du navigateur
- Vérifier les permissions
- Tester l'API avec coordonnées réelles
- Détecter les problèmes spécifiques mobile/desktop

Plan de diagnostic recommandé

Étape 1 : Test de la page de diagnostic

1. Allez sur `http://localhost:3000/debug-gps`
2. Cliquez sur "Tester GPS"
3. Autorisez la géolocalisation si demandé
4. Observez si ça fonctionne

Étape 2 : Si ça marche sur la page de diagnostic

- Le problème est dans l'interface utilisateur principale
- Ouvrir F12 sur `/places`
 - Copier le contenu de `/scripts/test-frontend-gps.js` dans la console
 - Identifier les différences de comportement

Étape 3 : Si ça ne marche pas sur la page de diagnostic

- Le problème vient du navigateur/permissions
- Vérifier les permissions dans les paramètres du navigateur
 - Tester sur un autre navigateur
 - Vérifier si c'est un problème mobile vs desktop

Causes probables identifiées

Problèmes côté navigateur :

- Permission de géolocalisation refusée
- Géolocalisation non supportée
- Problème de précision GPS
- Timeout de la requête GPS

Problèmes côté frontend :

- Erreurs React pendant le traitement des coordonnées (déjà corrigées)
- Callbacks multiples ou concurrents
- Mauvaise gestion des états de chargement

Problèmes côté backend :

- Coordonnées invalides transmises à l'API (peu probable)
- Erreur de calcul de distance (testé - fonctionne)

Prochaines étapes

1. **Testez la page de diagnostic :** `http://localhost:3000/debug-gps`
2. **Signalez-moi les résultats :**
 - Le test GPS fonctionne-t-il ?

- Les coordonnées sont-elles obtenues ?
- L'API répond-elle avec des lieux proches ?
- Y a-t-il des erreurs dans la console ?

3. **Si nécessaire**, je pourrai :

- Ajuster la logique de géolocalisation
- Améliorer la gestion des erreurs
- Optimiser la performance
- Ajouter plus de logging pour debug



Notes spécifiques mobile

Sur mobile, la géolocalisation peut être plus sensible :

- Permissions à accorder dans les paramètres du navigateur
 - GPS peut prendre plus de temps
 - Précision variable selon l'environnement
 - Économie d'énergie qui peut affecter le GPS
-



Conseil : Commencez par tester la page de diagnostic, c'est le moyen le plus rapide d'identifier où se situe exactement le problème !