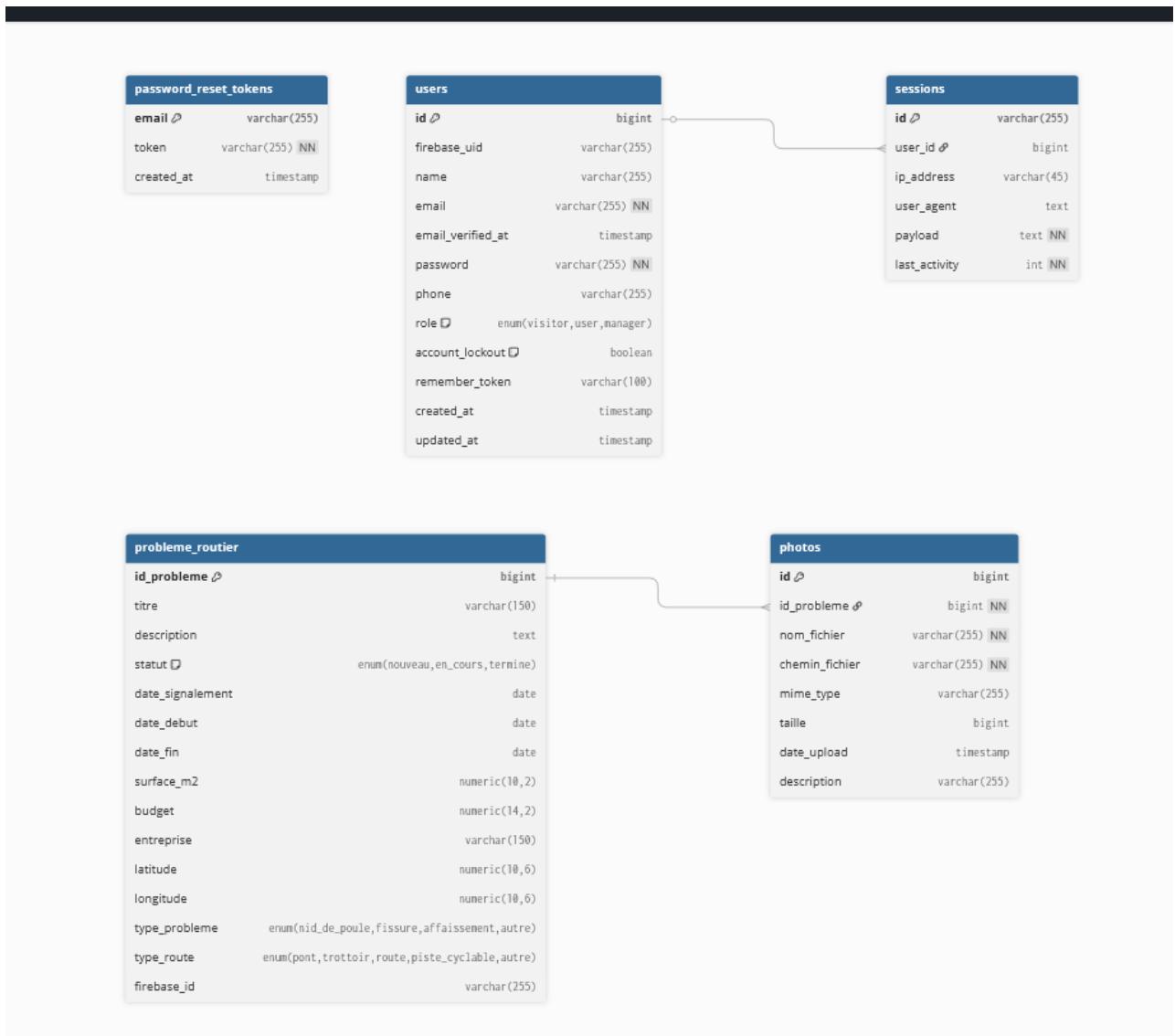


DOCUMENT TECHNIQUE PROJET CLOUD S5

- RAMANGASON Andrianina Irintsoa **ETU003168 (Michou00)**
- RAMORASATA Fifaliana Samiah **ETU003798 (FySamiah)**
- ROBEL Tsilavina **ETU003384 (Tsilat3384)**
- HARINJATOMALALA Hainiavo Kassaina **ETU003114 (kassa-hash)**

CAPTURE D'ECRAN DE MCD :



SCENARIOS D' UTILISATION :

MODULE AUTHENTIFICATION (test avec postman)

Scénario 1 : Inscription d'un utilisateur via API

Pré-condition : L'API est accessible (Firebase ou base locale).

Étapes :

1. Envoyer une requête POST vers /signup avec l'email et le mot de passe de l'utilisateur.
2. L'API vérifie la validité des informations.
3. L'API crée le compte dans Firebase si internet disponible, sinon dans la base locale Docker.

Résultat attendu :

- Code

HTTP 201 Created.

- Réponse JSON indiquant que l'utilisateur a été créé avec succès et l'ID de l'utilisateur.

The screenshot shows a Postman interface with the following details:

- Request URL:** http://localhost:8000/api/firebase/register
- Method:** POST
- Body:** Raw JSON (selected)
{"email": "test@example8.com", "password": "motdepasse123"}
- Response Status:** 201 Created
- Response Body (Raw JSON):**

```
1 {  
2   "source": "firebase",  
3   "id_token": "eyJhbGciOiJSUzI1NiIsImtpZCI6IjFjMzIx0TgzNGRhNTBlMjBmYWVhZWElYzg2Y2U3YjU1MzhmMTdizTEiLCJ0eXAiOjJKV1QifQ.  
4     eyJpc3MiOiJodHRwczovL3NlY3VyZXRxva2VuLmdvb2dsZS5jb20vY2xvdWQtODAyZk1LCJhdWQioiJjbG91ZC04M0djoSISimF1dGhfdglzS16Mtc20Dg50Tkx0SwidXN  
5       lc19pZCI6IjBiv1ldKdrTsJz0dwJQ1dqZ3R6dlV30dzTTi1LCJzdWIoiIW1ZXScn0bUo2TnViUEJXamdeXZVd09Ha2UyIiwiawF0IjoxNzY400k50TE5LCJ1eHAiOj  
E3Njg5MDM1MTksImVtYwlsIjoidgVzdEBleGFtcGx10C5jb20iLCJ1bwFpbf922XpZml1ZCIeZnFsc2UsImZpcmViYXNl1jp7ImlkZw50aXRpZXMiOnsizW1haWWiolsid  
GVzdeBleGFtcGx10C5jb20ixXOsInPz25faW5fcHJvdmlkZXI10iJwvXNzd29yZC39fQ.  
Ncr3f7r-yUbGlVKSPuYoZp-HWKy-P98dkFE2KxytJoz1R1vpewwwQdER4vy0419rSn7G9y93Qz5bSobpSOKs1K9NUL3A91EiWkff9z-Ravnva6Am0r1GQcZGpWZhLmjzID  
1rzCarw@2yR5pACFFHeQxs8AwDrzuACMabtYyNwW744a8e9hw4N0@t0xorrJFvF7PQw0BMd9jnzxJRD07XRFWubWKKr-G0k6ss6_Ecijmp4qN9goOPffaSj1o6v38Wu2  
KwLjxGuL0Gdv-Zt3QsKgzb_v1eck0egPz4wVH4K-PA4IdQ6HX6MrIk8f4zSV069f1EgaHWe6q-kophaa",  
"refresh_token":  
"AMf-v8x3cfkx08m8KUgXS_sCys-isURS3dDPiHEfuwQKk0X3IPy90qbRCuKS4h3FdVt__7887EMDHzRPCvcLT-101e_CQqY9iizNC_qirGdwZp7s156J6e06Uk2PB1wt09B  
arai1NNm27UtywL3Wkx0jMqJVsRHmj3wSbuacXhzV5jxwsX3iUTtaZcSRv2jeRsQKVz0ucpY5",  
"expires_in": 3600,  
"uid": "0bVWJw4mJ6NubPBWjgtvyUw0Gke2"
```

Scénario 2 : Connexion (Authentification) via API

Pré-condition : L'utilisateur est inscrit.

Étapes :

1. Envoyer une requête POST vers /login avec l'email et le mot de passe de l'utilisateur.
 2. L'API vérifie les identifiants : si internet, elle utilise Firebase ; sinon, la base locale Docker.
 3. Si les identifiants sont corrects, une session active est créée avec une durée de vie paramétrable.
 4. Si les identifiants sont incorrects, le compteur de tentatives est incrémenté.
 5. Après 3 tentatives échouées, le compte est automatiquement bloqué.

Résultat attendu :

- Code HTTP 200 OK si succès, 401 Unauthorized si échec.
 - Réponse JSON indiquant la connexion réussie et le token de session.
 - Blocage automatique du compte après dépassement du nombre de tentatives.

POST http://localhost:8000/api/firebase/login

Save Share

POST http://localhost:8000/api/firebase/login

Params Authorization Headers (9) Body Scripts Settings

none form-data x-www-form-urlencoded raw binary GraphQL JSON

Cookies Beautify

```
1 {  
2   "email": "test@example8.com",  
3   "password": "motdepasse123"  
4 }  
5
```

Body Cookies (2) Headers (8) Test Results 200 OK 12.40 s 1.46 KB

{ } JSON Preview Visualize

```
1 {  
2   "source": "firebase",  
3   "token_type": "Bearer",  
4   "id_token": "eyJhbGciOiJSUzIiNiImsmpZC16IjFzMzIx0TgzNGRhNTB1MjBmYwVhZwE3Yzg2Y2U3YjU1MzhmMTdizTE1LCj0eXAi0iJKV1Qifq.  
eyJpc3Mi0iJodHRwczovL3NlY3VyZXRxva2VuLmdvb2dsZS5jb20VY2xvdWQtODA3YzkilCjhwdWQi0iJjb91zC04MDdjOSiImF1dGhfdGltZSi6MtC20DkwmDA3NCwidXN  
1c19pZC16IjB1vlKQdrTSjZ00WJQQldqZ3R5i1V3T0dzT1iLCjdwIi0iIwY1ZXSnclb0u02TnV1UE3Xam0d0exZvd09Ha2UyIiwiawF0Ijoxnzy40TAwMdc0LCj1eHA1j  
E3Njg5MDm2NsInVmYtVsIj0idgVzdEBleGftcGx10C5jB20i1C1wPbf92ZXJzpmlzCI6ZmFsc2UsImZpcmViYXN1Ijp7Im1kZW5oaxRpZXm1Ons1ZwhawW0lsid  
GVzdEBleGftcGx10C5jB20i1X0sInNpZ25faW5fchJvdmIkZX1i0iJwYXNzd29yZCJ9fQ.  
N-eosZPb6x5yoTAzRoIrvyGPYPCn8-F3hevikaibmif1Jyy1Gv2gs-3DB-D8pwit1VAix9hLqJr4DvuK7ep_E6hA-mJ6bMDtkX1JapVjBeDg_Lf_-CtjD-Pio0SN13SiO-  
s1n2sDzDnKA_YIVoDzT2qLu2vD-icZUhNgrHwedhJ5oQmAMujmBd_Y6L1L9dG52vTPq8N1A-DqKnsvpVvm6tq18k1HMz0HMPbj7GMv0jukHce6MrblAKOfac0x6xrdNBxy  
1ohVoyPrRbc04irL0_QNK5kxey0PggcanhkkJawKqTOjeVH5CoI6Zv78EXG1vbKCUx86ew8o4g3_N_0",  
5   "refresh_token":  
6     "AMF-vBxs8F5TAJ0UNGvAt0Lbbi_dhA6M6Pk4wqvThc0QmqnuaeEb4_4HqcIJnky7cNRKz1dx--zyT1RCLw75-388AvxVLdVjyrfie_wM9nVYXtHK4fxzqj-SYD_eqj1vI  
7       uLZKqXchT576Qncu-f15c4Kex8BZ3X0edxQinrCvg2Leju0TIRqqULqQuXh2vQP5W3ELQix-2",  
8   "expires_in": 3600,  
9   "uid": "0bVWJw4mJ6NuBWPbjgytvUw0Gke2"
```

Scénario 3 : Blocage / Déblocage via API

Pré-condition : Le compte est bloqué après dépassement de tentatives.

Étapes :

1. Envoyer une requête POST vers /users/{id}/reset-block pour réinitialiser le blocage.
2. L'API débloque le compte et réinitialise le compteur de tentatives.

Résultat attendu :

- Code HTTP 200 OK.
- Réponse JSON indiquant que le compte a été débloqué avec succès.

Scénario 4 : Vérification durée de vie des sessions

Pré-condition : L'utilisateur est connecté et possède un token valide.

Étapes :

1. Vérifier que la session expire après la durée configurée.
2. Envoyer une requête avec le token expiré : l'API doit refuser l'accès.

Résultat attendu :

- Code HTTP 401 Unauthorized si le token est expiré.
- Réponse JSON indiquant que la session a expiré.

MODULE CARTE

Scénario 1 : Installation et mise en route du serveur de carte offline

Pré-condition : Docker est installé.

Étapes :

- Télécharger et lancer le serveur de cartes (MapServer ou TileServer) sur Docker.
- Vérifier que le serveur répond aux requêtes locales.

Résultat attendu :

- Le serveur est actif et fournit les tuiles cartographiques.
- Les tuiles sont prêtes pour l'affichage dans l'application web.



Scénario 2 : Téléchargement et intégration des données d'Antananarivo

Pré-condition : Serveur de carte offline opérationnel.

Étapes :

- Télécharger les données OSM pour Antananarivo (rues et quartiers).
- Importer les données dans le serveur offline et vérifier le rendu.

Résultat attendu :

- La carte affiche correctement les rues et zones principales.

Do you find OpenStreetMap data useful? Consider giving something back in the OSM's [funding campaign](#)

Download OpenStreetMap data for this region:

Madagascar

[one level up]

The OpenStreetMap data files provided on this server do **not** contain the user names, user IDs and changeset IDs of the OSM objects because these fields are assumed to contain personal information about the OpenStreetMap contributors and are therefore subject to data protection regulations in the European Union.
[Extracts with full metadata](#) are available to OpenStreetMap contributors only.

Commonly Used Formats

- [madagascar-latest.osm.pbf](#), suitable for Osmium, Osmosis, imposm, osm2pgsql, mkgmap, and others. This file was last modified 3 hours ago and contains all OSM data up to 2026-01-19T21:21:45Z. File size: 361 MB; MD5 sum: 818c4e42208a8e80691dcf2c0f432949.
- [madagascar-latest-free.shp.zip](#), yields a number of ESRI compatible shape files when unzipped. ([Format description PDF](#)) This file was last modified 1 hour ago. File size: 1010 MB; MD5 sum: 099ffcd2be1bbcddebc4e59f2459aa06.

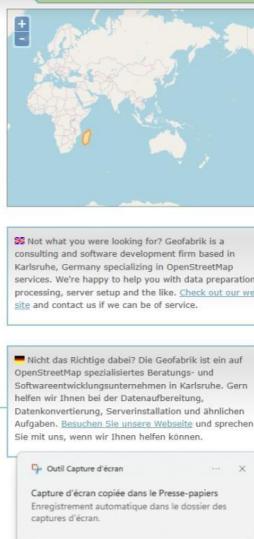
Other Formats and Auxiliary Files

- [madagascar-internal-shp.gpkg](#) The history file contains personal data and is available on the [internal server](#) only. See notice above for further information.
- [.osm.bz2](#) file that describes the extent of this region.
- [experimental vector tile package](#) conforming to [Shortread schema](#) for use with MapLibre and other MVT capable software
- [.osm.gz](#) files that contain all changes in this region, suitable e.g. for Osmosis updates
- [Taginfo statistics for this region](#)
- [raw directory index](#) allowing you to see and download older files

Sub Regions

No sub regions are defined for this region.

Data processed by [Geofabrik GmbH](#) and created by [OpenStreetMap Contributors](#) | License: [ODbL 1.0](#) | Contact | Legal Notice/Privacy Policy



Scénario 3 : Affichage et manipulation de la carte dans l'application web

Pré-condition : Serveur et données prêts.

Étapes :

- Charger Leaflet dans l'application web et configurer pour utiliser le serveur offline.
- Ajouter zoom, déplacement et marqueurs pour signalements.
- Permettre l'affichage des informations au clic sur un marqueur.

Résultat attendu :

- La carte est interactive et exploitable.
- Les utilisateurs peuvent zoomer, déplacer et consulter les informations des points.

MODULE WEB

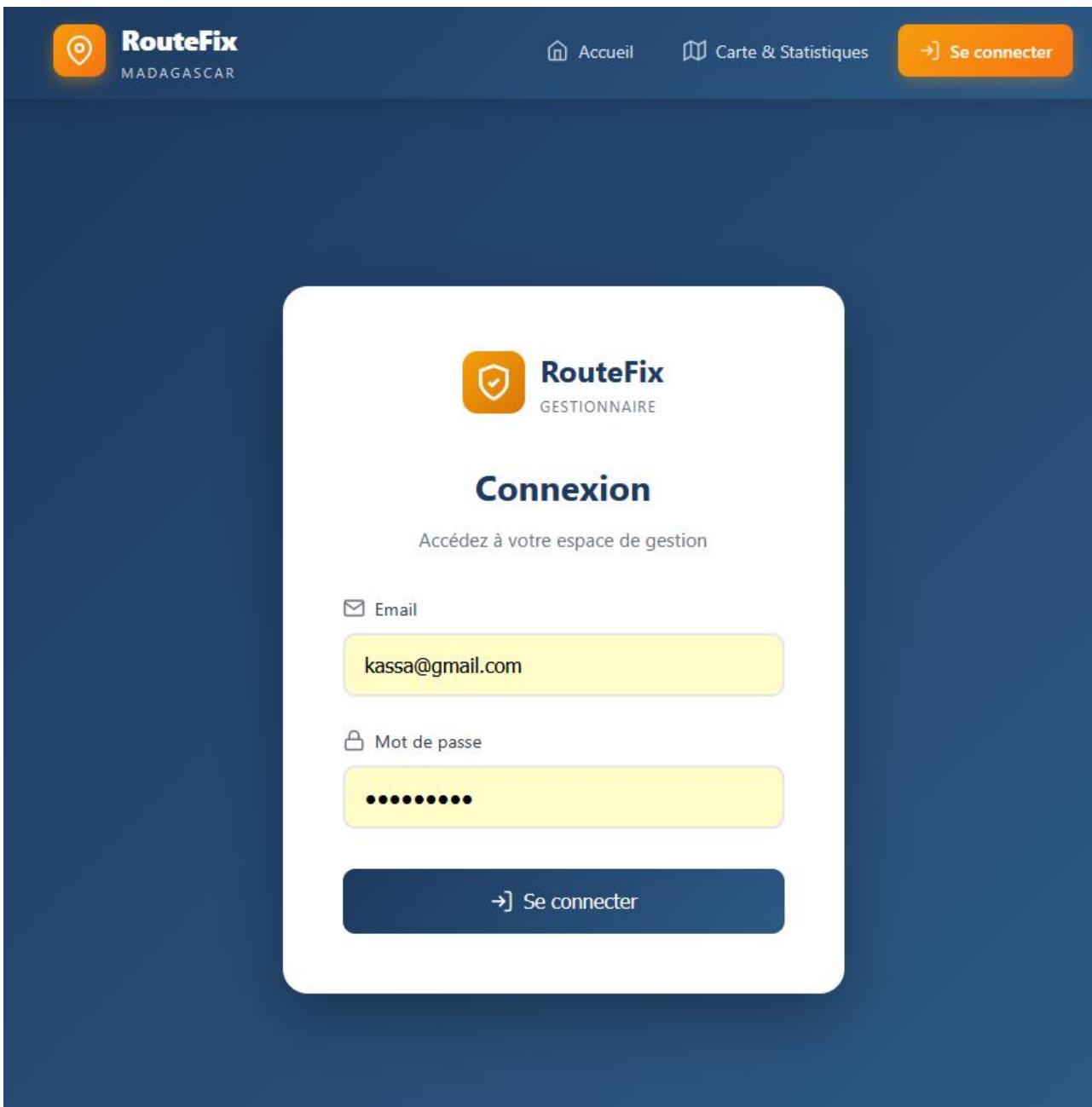
Suivi des travaux routiers à Antananarivo

Scénario 1 : Accès à la page de connexion

Pré-condition : L'application Web est accessible.

Étapes : L'utilisateur accède à l'URL de l'application Web. La page de connexion s'affiche.

Résultat attendu : Le formulaire de connexion est visible.

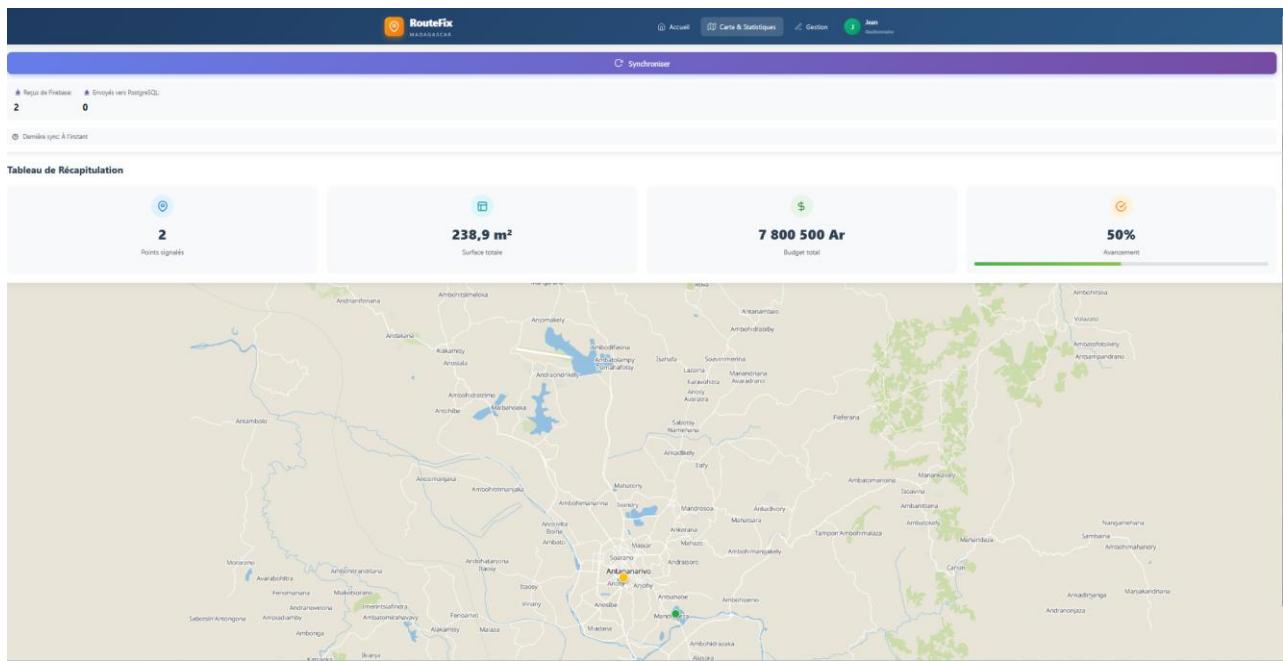


Scénario 2 : Connexion du manager

Pré-condition : Le compte manager existe.

Étapes : Le manager saisit son email et son mot de passe. L'application envoie les informations à l'API REST d'authentification. L'API valide les identifiants.

Résultat attendu : Connexion réussie. Accès à l'interface de gestion.



Gestion des Signalements
Synchronisez, gérez et suivez les problèmes routiers

Synchronisation Firebase
Récupérez les nouveaux signalements depuis Firebase vers PostgreSQL

Synchronisation...

Envoi vers PostgreSQL (2/2)

Reçus de Firebase: 2 Envoyés vers PostgreSQL: 0

2 Total signalements	0 Nouveaux	1 En cours
1 Terminés		

Scénario 3 : Création d'un compte utilisateur par le manager

Pré-condition : Le manager est connecté.

Étapes : Le manager accède à la page de création de compte. Il saisit les informations de l'utilisateur. Les données sont envoyées à l'API REST.

Résultat attendu : Le compte utilisateur est créé avec succès.

Inscription

Nom complet
Jean Dupont

Email
kassa@gmail.com

Téléphone (optionnel)
+33 6 00 00 00 00

Mot de passe
.....

Confirmer le mot de passe
.....

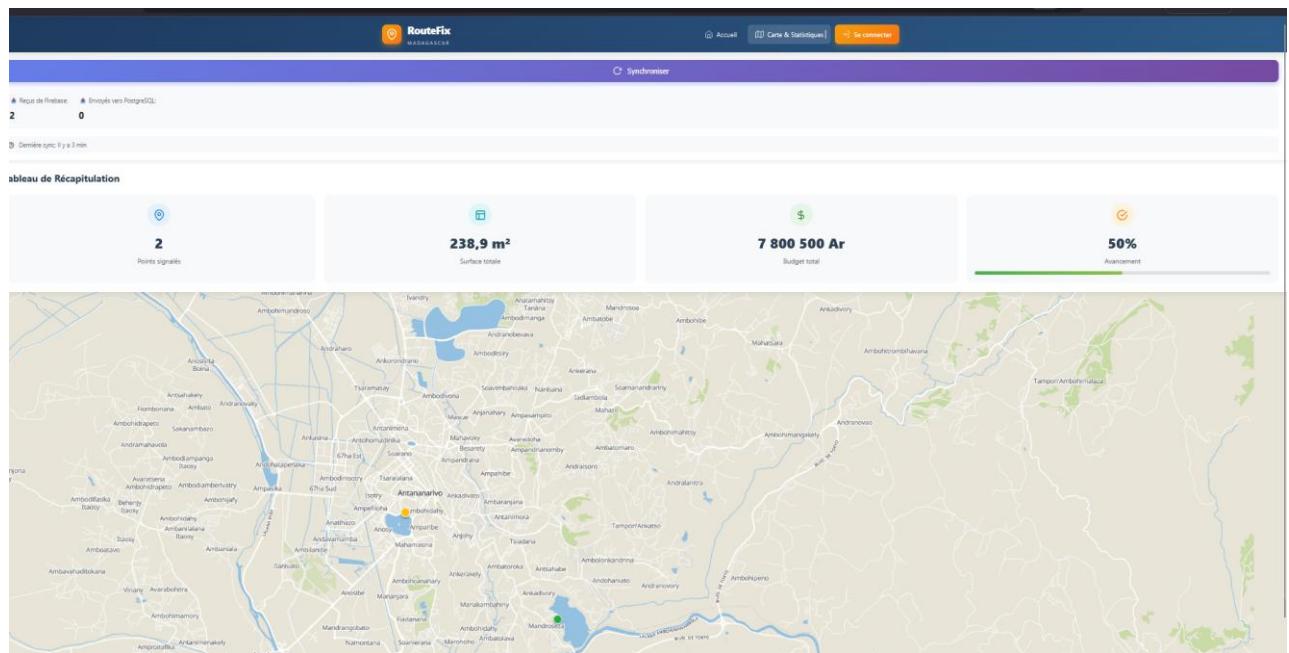
S'inscrire

Scénario 4 : Consultation de la carte par un visiteur

Pré-condition : Des signalements existent.

Étapes : L'utilisateur accède à l'application . La carte d'Antananarivo s'affiche. Les points de signalement sont visibles.

Résultat attendu : La carte affiche les problèmes routiers.

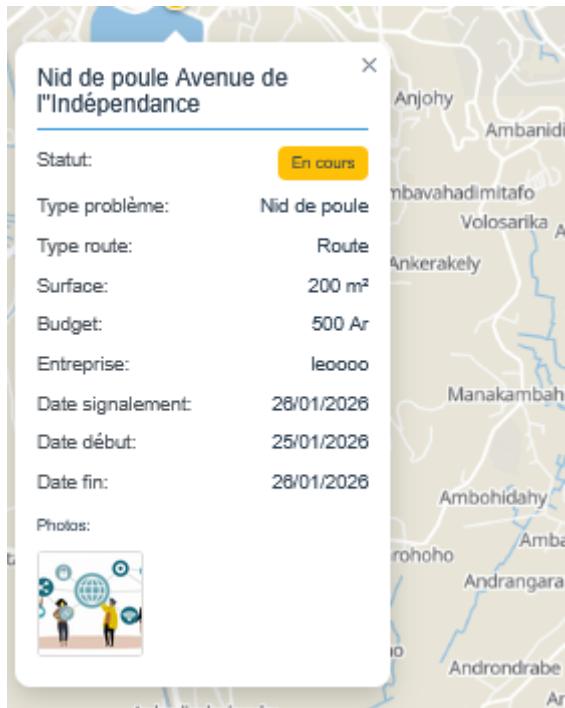


Scénario 5 : Consultation des informations d'un signalement

Pré-condition : La carte est affichée.

Étapes : Le visiteur ou le manager survole un point. Une infobulle s'affiche.

Résultat attendu : Affichage de la date, du statut, de la surface en m², du budget et de l'entreprise concernée, de la photo



Scénario 6 : Consultation du tableau de récapitulation

Pré-condition : Des signalements sont enregistrés.

Étapes : L'utilisateur consulte la page principale. Le tableau de récapitulation est affiché.

Résultat attendu : Affichage du nombre de points, de la surface totale, de l'avancement global et du budget total.

Scénario 7 : Gestion des signalements par le manager

Pré-condition : Le manager est connecté.

Étapes : Le manager accède au formulaire d'ajout. Il sélectionne la localisation sur la carte. Il saisit les informations du signalement.

Résultat attendu : Le signalement est enregistré avec succès.

Scénario 8 : Modification d'un signalement

Pré-condition : Un signalement existe.

Étapes : Le manager sélectionne un signalement. Il modifie les informations ou le statut. Les modifications sont enregistrées.

Résultat attendu : Le signalement est mis à jour sur la carte.

Scénario 9 : Synchronisation des données avec Firebase

Pré-condition : Une connexion Internet est disponible.

Étapes : Le manager clique sur le bouton de synchronisation. Les données Web sont envoyées vers Firebase. Les données Mobile sont récupérées.

Résultat attendu : Les données sont synchronisées entre le Web et le Mobile.

Scénario 10 : Déblocage d'un utilisateur

Pré-condition : Un utilisateur est bloqué.

Étapes : Le manager accède à la gestion des utilisateurs. Il sélectionne l'utilisateur bloqué. Il déclenche le déblocage.

Résultat attendu : Le compte utilisateur est débloqué.

Connexion

Email

bbb@gmail.com

Mot de passe

.....

Se connecter

Compte bloqué après trop de tentatives. Veuillez contacter un administrateur.

Gestion des Comptes Bloqués

1 Compte(s) bloqué(s)

Actualiser

EMAIL	NOM	RÔLE	TYPE DE BLOCAGE	TENTATIVES	DÉBLOQUÉ JUSQU'À	ACTIONS
bbb@gmail.com	bbb	USER	BLOQUÉ	3/3	27/01/2026 13:05	<button>Débloquer</button>

MODULE MOBILE

Suivi des travaux routiers à Antananarivo

Scénario 1 : Connexion utilisateur

Description : L'utilisateur se connecte à l'application mobile en utilisant son email et son mot de passe via Firebase.

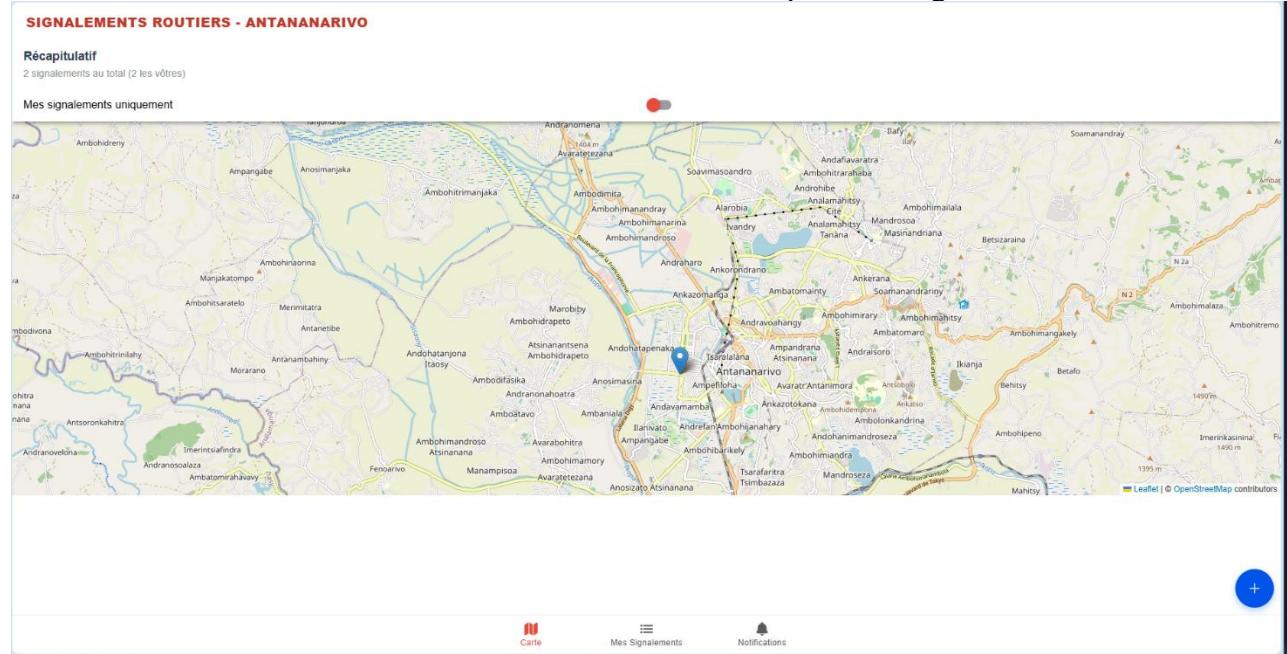
Résultat attendu : L'utilisateur accède à l'application et à la carte.



Scénario 2 : Affichage de la carte des signalements

Description : L'application affiche la carte d'Antananarivo avec les différents points représentant les problèmes routiers.

Résultat attendu : La carte interactive s'affiche avec les marqueurs de signalement.



Scénario 3: Signalement d'un problème routier

Description : L'utilisateur signale un nouveau problème routier en sélectionnant une localisation sur la carte et en saisissant les informations nécessaires.

Résultat attendu : Le signalement est enregistré et apparaît sur la carte.

Scénario 4 : Notification lors du changement de statut d'un signalement

Description : L'utilisateur reçoit une notification lorsque le statut d'un de ses signalements est modifié par le manager.

Résultat attendu : L'utilisateur est informé en temps réel du changement de statut de son signalement.

The screenshot shows a 'NOTIFICATIONS' section with three items:

- Signalement Validé**: Your report "Rue de la paix" has been validated by authorities. (2 hours ago)
- Nouveau statut**: The status of your report in Analakely has changed to "In progress". (Yesterday)
- Bienvenue sur RoadRed**: Thank you for using RoadRed to improve our roads! (27/01/2026)

GITHUB

Lien : <https://github.com/Kassa-hash/projetFinalS5.git>

branche : web

(les noms d'utilisateurs des membres du groupes dans le git sont tenus hauts du pdf avec les noms)