

# Compte rendu de projet

# SACC

— M. Huet Fabrice

**Groupe sacc-quarantine**

Paul-Marie DJEKINNOU

Paul KOFFI

Florian AINADOU

Djotiham NABAGOU

# PLAN

<b>Vue d'ensemble et choix d'implémentation</b>	<b>3</b>
<b>Routes REST</b>	<b>4</b>
<b>Architecture du projet</b>	<b>7</b>
<b>Modèle de données Firestore</b>	<b>8</b>
<b>Lien de la vidéo de démo d'élasticité</b>	<b>8</b>

## Vue d'ensemble et choix d'implémentation

Nos choix ont évolué au cours du développement de l'application. Pour ce qui est de la base de données, nous avons fait deux choix:

- **Firestore pour stocker les utilisateurs, les administrateurs et les messages de proximité**

Firestore a été choisi parce que les données stockées dans la bd sont fortement évolutives. Typiquement le fait de rajouter un champ de date lors du marquage de personne en tant que personne of interest n'aurait pas été aussi simple. Car avec du NoSQL, le format des documents peut varier sans réellement poser de problème.

- **Cloud Storage pour le stockage de fichiers**

Cloud Storage a été choisi pour le faible coût de ce service qui permet cependant de stocker tout type de fichiers sans contrainte de taille minimale ainsi que des fichiers volumineux. Elle permet une accessibilité en temps rapide, est sécurisée et permet de choisir des politiques de conservation (Ex : Suppression d'une stat après 30 jours si on veut car jugé inutile).

Pour les mécanismes de communication, nous avons essentiellement utilisé:

- **les pushes queues pour les requêtes simples**

En utilisant les pushes queues, les appels aux servlets sont mieux traités. La configuration utilisée pour les push queues est disponible dans fichier **queue.yaml** a la racine. Les queues créées nous permettent de gérer **20 requêtes concurrentes** avec un taux de **20 requêtes par seconde**. Nous avons plusieurs queues afin de répartir au mieux le trafic.

- **les pub/sub**

Nous avons utilisé les pub/sub pour les statistiques car même si les statistiques sont simples lorsque la base de données deviendra très grande le parcours des données peut devenir long et donc une push queue ne serait adaptée. Et pour les statistiques plus complexes, elles sont de longues par défaut et donc le pub/sub est appropriée

- **envoi des mails**

Pour l'envoi des mails à un administrateur, nous avons utilisé un service tiers : [Mailjet](#) qui est un service de messagerie mondial qui offre un quota gratuit aux utilisateurs d'App Engine (6000 Mails gratuits par mois). La création de compte est rapide et après génération de ses clés publiques et privées, l'utilisation avec Java 8 est facile à prendre en main grâce à leur API.

- **Elasticité**

Pour l'élasticité, nous avons mis en place la configuration suivante dans le fichier appengine-web.xml :

```
<application>sacc-quarantine</application>
<module>default</module>
<version>uno</version>
<instance-class>F2</instance-class>
<automatic-scaling>
  <target-cpu-utilization>0.5</target-cpu-utilization>
  <min-instances>1</min-instances>
  <max-instances>5</max-instances>
  <max-concurrent-requests>50</max-concurrent-requests>
</automatic-scaling>
```

## Routes REST

### 1. Routes Admin

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/addPersonOfInterest>

Requête **POST** permettant l'ajout d'une Pol en passant par une queue.

```
{
  "phoneNumber" : "077779127",
  "personOfInterest": true
}
```

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/admin>

Requête **POST** permettant l'ajout d'un admin en passant par une queue.

```
{
  "email": "florain.ainadou97@gmail.com",
  "name": "Djodjo"
}
```

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/deleteUploadStats>

Requête **POST** permettant la suppression des statistiques générées en passant par une queue.

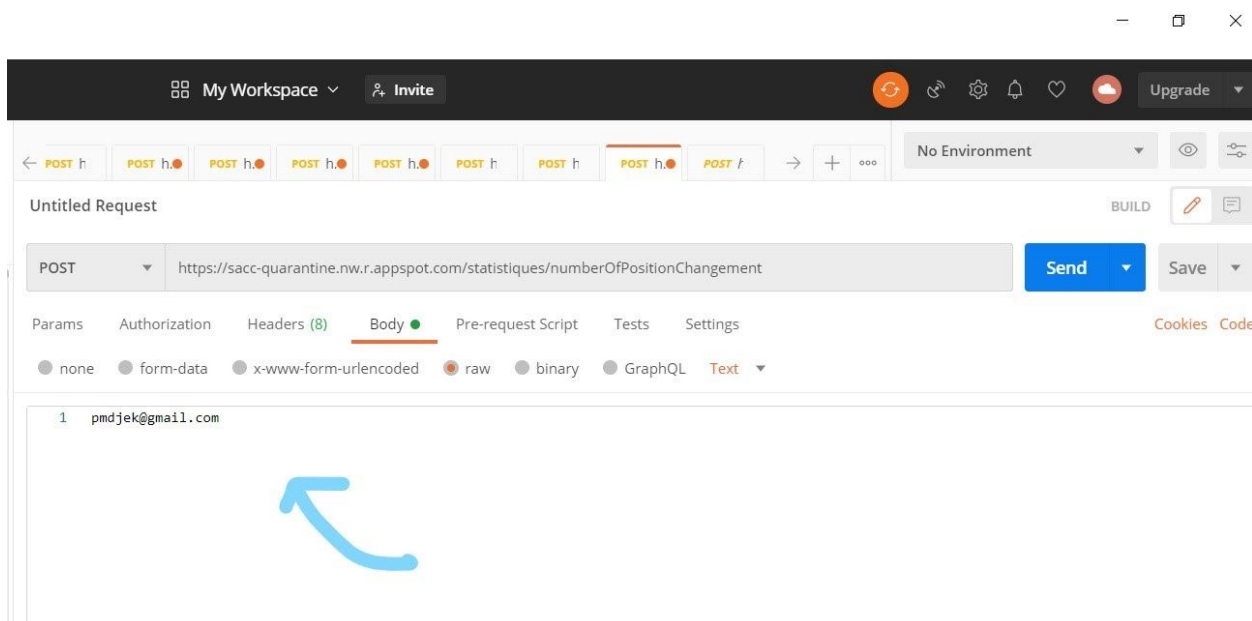
### 2. Routes proximityMessage

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/sendProximityMsg>

Requête **POST** permettant d'envoyer un message de proximité en passant par une queue. Elle ne fonctionnera que si les deux utilisateurs existent dans la base de données.

```
{
  "user1PhoneNumber": "077779127",
  "user2PhoneNumber": "077779128",
  "user1CurrentLocation": {
    "longitude" : 44.5,
    "latitude" : 44.4
  },
  "user2CurrentLocation": {
    "longitude" : 44.5,
    "latitude" : 44.4
  }
}
```

### 3. Routes de Statistiques



Les stats sont envoyés à l'adresse mail renseignée

#### ○ Statistiques de base

- <https://sacc-quarantine.nw.r.appspot.com/statistiques/numberOfPOI>  
Requête **POST** permettant d'avoir le nombre de Pol.  
Body de la requête : [adminAdressMail@xxxx.com](mailto:adminAdressMail@xxxx.com)  
**PS : sans côtes**
- <https://sacc-quarantine.nw.r.appspot.com/statistiques/numberOfPositionChangement>  
Requête **POST** permettant d'obtenir le nombre de changement de position des utilisateurs de l'application.  
Body de la requête : [adminAdressMail@xxxx.com](mailto:adminAdressMail@xxxx.com)  
**PS : sans côtes**
- <https://sacc-quarantine.nw.r.appspot.com/statistiques/numberOfUsers>  
Requête **POST** permettant d'avoir le nombre total d'utilisateurs.  
Body de la requête : [adminAdressMail@xxxx.com](mailto:adminAdressMail@xxxx.com)  
**PS : sans côtes**

#### ○ Statistiques complexes

- <https://sacc-quarantine.nw.r.appspot.com/last24hours>  
Requête **POST** permettant d'avoir les personnes à proximité de POI les dernières 24 heures.  
Body de la requête : [adminAdressMail@xxxx.com](mailto:adminAdressMail@xxxx.com)  
**PS : sans côtes**

#### 4. Routes d'utilisateurs

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/changePosition>

Requête **POST** permettant de changer la position d'un utilisateur en passant par une queue.

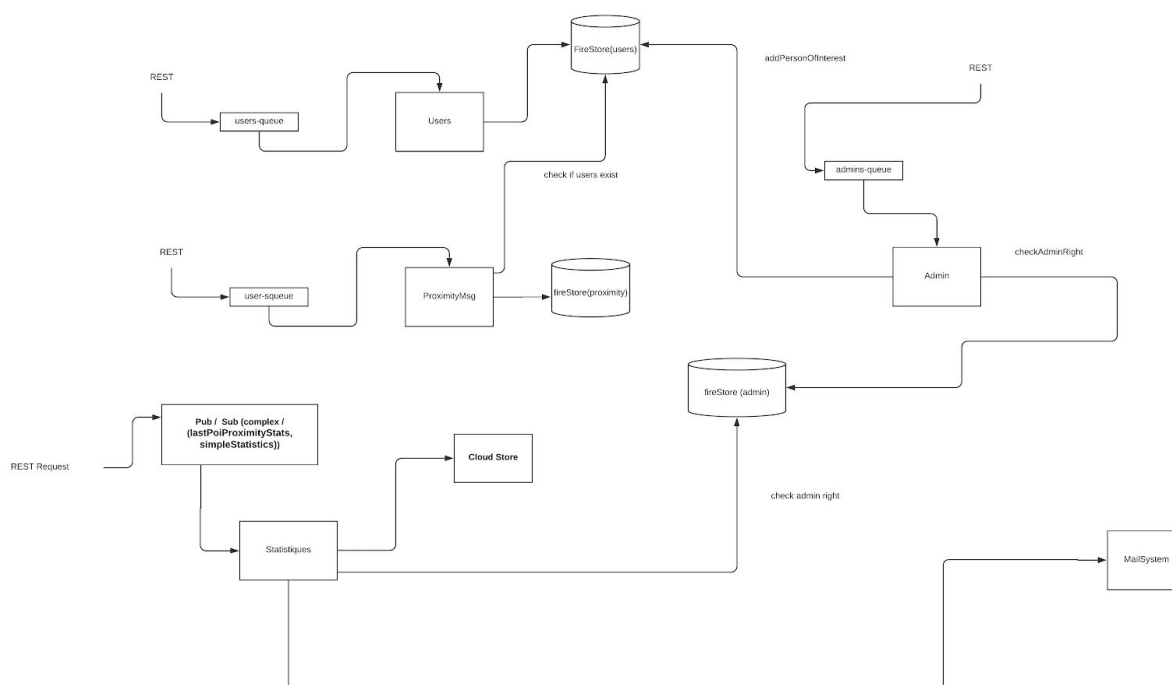
```
{
  "phoneNumber": "077779127",
  "location": {
    "longitude" : 44.5,
    "latitude" : 44.4
  }
}
```

- <https://sacc-quarantine.nw.r.appspot.com/taskqueue/user>

Requête **POST** permettant d'ajouter des utilisateurs.

```
{
  "phoneNumber": "087779128",
  "email": "blackmail@gmail.com",
  "location": {
    "longitude" : 44.5,
    "latitude" : 44.4
  }
}
```

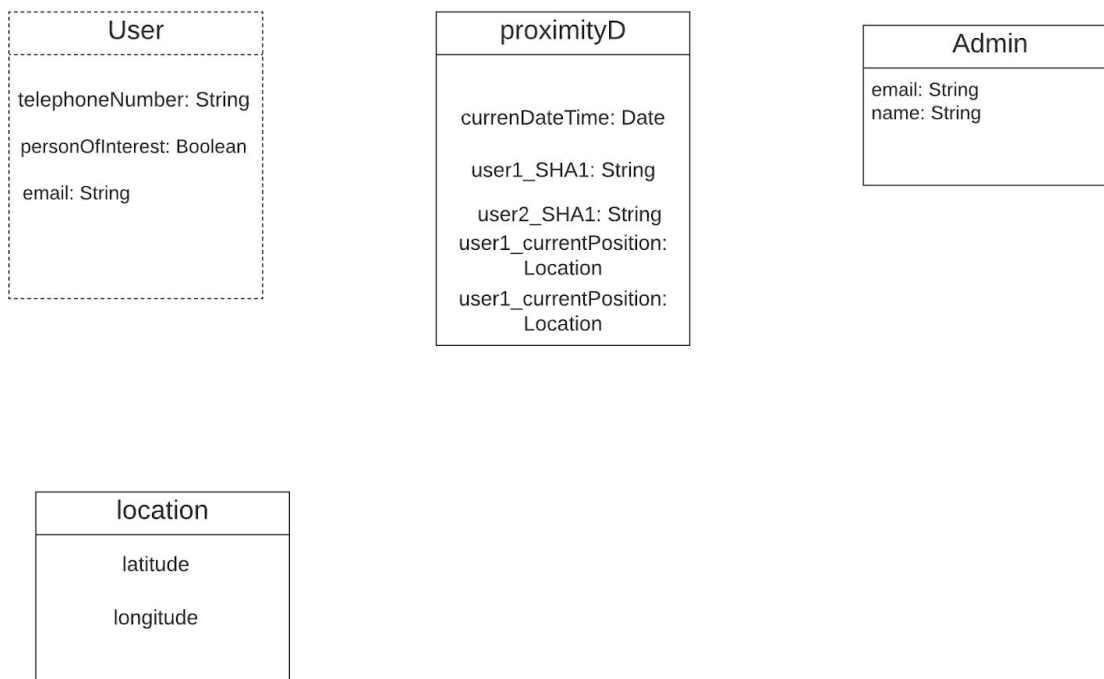
## Architecture du projet



*Figure 1 : Architecture du projet*

(PS: Disponible en taille réelle dans le readme du projet)

## Modèle de données Firestore



*Figure 2 : Données Firestore*

## Lien de la vidéo de démo d'élasticité

<https://drive.google.com/file/d/1Rq9RPlwxrDVso98rgOe4KmkEjffE-uH8/view?usp=sharing>