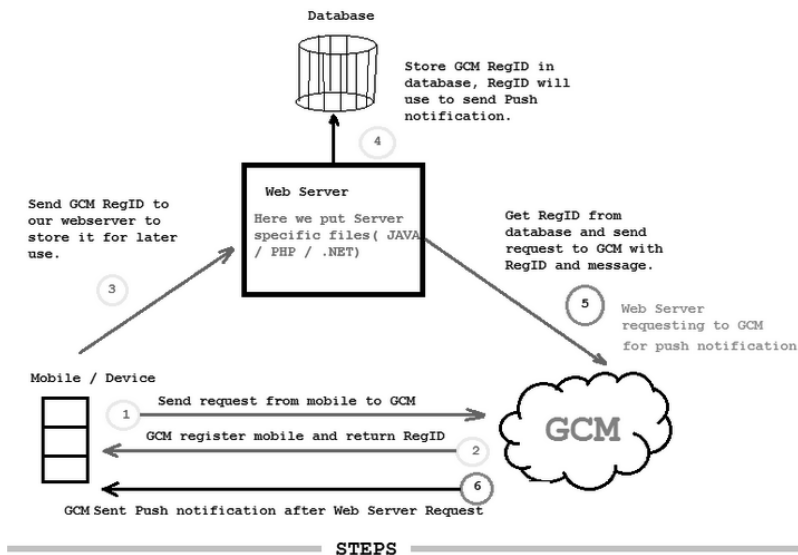


Réunion du 12/05

Ce que j'ai fait :

- Statistiques :
 - Eventbus, reload
 - Corriger spinner affiche bon sensor
 - Corriger les calculs diff et jour/nuit côté serveur
 - Correction conversion wh en kwh
 - Fix : no values found for this period
- Comparaison :
 - Eventbus, reload
 - Spinner pour choisir le sensor
- Push notification: alerte si consommation anormale



STEPS

- 1 Android device sends SENDER_ID to GCM server for registration.
 - 2 After successful registration GCM Server return registration ID to Android device.
 - 3 After get registration ID Android device send registration ID to Web server.
 - 4 Store GCM registration ID in our database at server.
 - 5 Whenever Push notification needed get RegID from our database and send request to GCM with RegID and message.
 - 6 After got Push notification request GCM send Push notification to Android device.
- - **Serveur** : algorithme pour générer les notifications au bon moment (une fois par jour)
 - Création d'une méthode POST pour stocker le registration ID côté *back end*
 - Il faut associer cet id à un user
 - Coté app, c'est enregistré dans les préférences
 - Api key properties externalised in application.properties
 - **Appli** :
 - Une notification par sensor maximum (pour ne pas être spammé)
 - Action: ouverture de l'app sur Chartactivity quand on clique sur la notif
 - Couleur de l'app, icône & default profile background
 - Rédaction du TFE : problem description, use cases, implementation back end front end (en cours)

Ce que je projette de faire

- Améliorer la sécurité : authentification en cours, https, accès à *Mongo Db*
 - API : l'utilisateur doit pouvoir accéder uniquement à ses capteurs
- Continuer les cas d'utilisation :
 - Permettre d'encoder manuellement sa consommation (voir si compliqué, ça peut être utile pour ceux qui n'ont pas *Flukso*)