

## 05/05 : 15ème réunion

### Ce que j'ai fait :

- App : ajout de l'option "add sensor" au navigation drawer
  - Simplification de la communication entre les activités, les fragments et les threads au niveau de l'app grâce à un *EventBus*
    - Ca a permis de corriger quelques bugs (rotation de l'écran, reload, ...)
    - Librairie : <https://github.com/greenrobot/EventBus>
      - EventBus is publish/subscribe event bus optimized for *Android*. *EventBus...* simplifies the communication between components decouples event senders and receivers performs well with Activities, Fragments, and background threads avoids complex and error-prone dependencies and life cycle issues makes your code simpler is fast is tiny (<50k jar) is proven in practice by apps with 100,000,000+ installs has advanced features like delivery threads, subscriber priorities, etc.
  - VM avec IP publique et déploiement du serveur
    - correction de dependencies conflicts au niveau de *Maven*
    - period between timestamps is 60 seconds since Tue Apr 28 11:17:00 CEST 2015
    - à voir avec Antoine :
      - ce que je fais quand serveur not connected (period between two timestamps > 60 secs) ?
      - calculer uniquement les stats sur les valeurs correspondant à une période de 60 secondes ?
  - (en cours) Correction d'un bug lié au stat côté serveur
    - Sens des valeurs retournées par *Flukso* : à ce timestamp précis, on consomme autant
      - POWER (Watts, or W) is the RATE of using or producing electrical energy (or how much is being used right now).
      - ENERGY (kilowatt-hours, or kWh) is the TOTAL amount of electricity used or produced over a period of time.
- |                  |                            |  |  |
|------------------|----------------------------|--|--|
| maxTimestamp     | date                       |  |  |
| maxValue         | watt                       |  |  |
| minTimestamp     | date                       |  |  |
| minValue         | watt                       |  |  |
| average          | average des values en watt |  |  |
| consumption      | wh                         |  |  |
| consumptionDay   | wh                         |  |  |
| consumptionNight | wh                         |  |  |
| diffLastTwo      | wh                         |  |  |
- J'ai adapté le controller rest pour les stats (object mapping + API publique myconsumptionserver)
  - Pas besoin d'adapter l'app grâce au DTO
  - Rédaction du TFE : problem description, design, use cases (features discussed, compromises)

### Ce que je projette de faire

- Améliorer la sécurité : authentification en cours, https, accès à *MongoDB*
  - API : l'utilisateur doit pouvoir accéder uniquement ses capteurs
- Statistiques :
  - Corriger spinner affiche bon sensor
  - Corriger les calculs diff et jour/nuit côté serveur
- Comparaison :
  - Spinner pour choisir le sensor
- Continuer les cas d'utilisation :
  - Alerte si consommation anormale (il faut gérer les services)