

Rencontre avec Romaric

15 décembre

- Mode vacances ? Plus de modes genre nuit et tout ?

- **IHM : visuel**

- *Les interrupteurs virtuels :*

Romaric tient à ce que les groupes et les interrupteurs virtuels puissent se distinguer. Les groupes c'est pour classer et les interrupteurs c'est pour agir de groupe. Ce qu'on doit faire : Présenter dans l'onglet « maison » les interrupteurs virtuels favoris (définis par l'administrateur) de ce niveau : « éclairage de maison » « chauffage de maison », etc. De plus, dans les groupes et les salles, rajouter également interrupteurs virtuels : « électroménager de la cuisine » « éclairage de la cuisine », etc..

Prévoir la possibilité d'associer un interrupteur virtuel à un groupe, mais en option. De même , il faut pouvoir associer un interrupteur virtuel à un interrupteur physique (qui doit alors apparaître dans l'IHM). Utiliser le touché prolongé est une bonne idée.

Une seule charge peut être dans plusieurs groupes/interrupteurs virtuels.

- *Le code couleur :*

Le code couleur affiché correspond à l'état effectif de la source.

Vert	Actif
Gris	Désactivé
Rouge	Normalement ON mais délesté

Il faut colorier les différentes sources également. Mais ce choix n'est pas arrêté.

L'admin doit pouvoir choisir les couleurs.

- *La prévision*

Pas dans le cahier des charges mais plutôt une bonne idée. Échelle proportionnelle ok. Le mettre dans la consommation pas en tête en gondole tout le temps.

- *Menu détaillé des charges*

L'alerte doit se distinguer du menu détaillé de la charge. Virer aussi la consommation. Ajouter un ON/OFF.

- *Priorité de la panne*

Il faut que chaque panne ait un ordre de priorité qui classe l'importance de ce qui cloche. Mention spéciale pour appeler le technicien.

- *Nature des statistiques :*

On calcule les kW. C'est à l'admin de mettre le prix du kW, et le programme pourra éventuellement présenter ce chiffre. Ne pas oublier d'ajouter le %. Penser aussi à faire ces champs pour les sources (si on a le temps).

Faire les graphes dans un second temps, se concentrer sur le tableau de valeur dans un premier temps

Discussion sur qu'est ce qu'on met sur la première de la consommation : graphe puis tableau.

Toutes les statistiques sont glissantes.

- **Base de donnée**

- *L'état par défaut :*

L'État par défaut est pour l'état de démarrage du système, y compris après un délestage.

- *Sources :*

Il y aura bien des capteurs.

- *USER :*

Tous les mêmes avec les mêmes droits.

- *Rapport :*

Faire figurer les groupes dans le rapport : des graphes et des chiffres.

- *Gérer la localisation :*

L'administrateur doit faire le lien entre la charge et le boîtier. Donc il doit pouvoir modifier la localisation de chaque charge. Si on modifie le câblage de la maison. ID du boîtier secondaire, ID de la charge. Coté admin, ça peut ne pas être très joli.

La solution serait de mettre un numéro sur chaque prise pour donner son ID.

Penser à pouvoir changer

On doit aussi pouvoir gérer l'ajout de charges

- *Gestion des données :*

L'administrateur doit pouvoir gérer l'organisation par plan électrique. Une simple image pour rappeler l'architecture suffit, rien de dynamique.

- *Restrictions base de donnée :*

Aucune contradiction sur la base de donnée. On aura toute la puissance qu'il faut. Tout va tourner sur un Raspberry sous système Linux.

C'est le Raspberry qui remplira la base de donnée

- *Modification de la Base de donnée :*

Il faut une MAJ des actions.

Pour l'instant on considère que notre base de donnée est modifiée par un programme et on peut la changer aussi.

Comment on fait pour tester la simulation ? Romaric essaiera de fournir un certain nombre d'informations.

- *Distinction boîtiers primaires et secondaires :*

Boîtiers primaires et secondaires sont vraiment différents, donc on ne peut pas le fusionner.

- **Android :**

- Pour l'Android, on peut se calmer un peu. C'est *vraiment* facultatif. Faire plutôt une appli adaptative design.
- En revanche, si on part sur une appli, autant faire du JEE, parce qu'on peut utiliser des Web Services. La puissance de ce truc c'est qu'une fois qu'on a défini ça, quand on code l'appli on appelle des méthodes qui sont dans la librairie, on croit appeler des méthodes locales, mais en fait c'est le serveur qui fait le taff.