

Auteurs : BOUCQ Maxime – MOUSSU Nathan – LAFAGE Robin

CONTEXTE ET OBJECTIF

ENSTO commercialise des **coffrets de contrôle (IED)** pour les lignes électriques, qui permettent en autres d'actionner des interrupteurs ou de **superviser** l'état des lignes, réduisant la durée et la fréquence des coupures électriques.

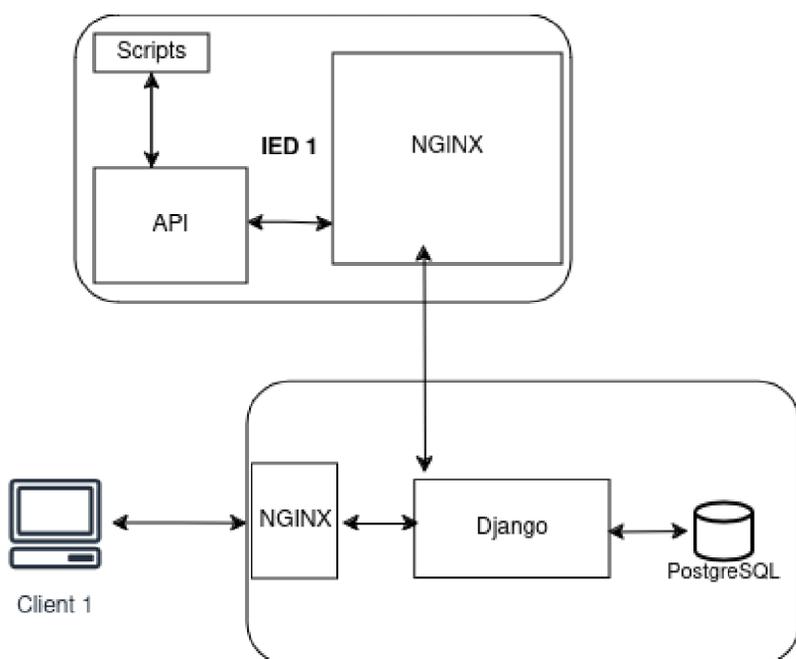
À l'heure actuelle, il n'est pas possible de communiquer avec plusieurs coffrets en simultanément. Pour pallier ce problème, il est nécessaire d'implémenter une **interface** dans les coffrets. Celle-ci prend la forme d'une **API REST**.

De plus, afin de communiquer avec plusieurs IEDs d'un coup, et d'afficher les résultats de manière simple et lisible, une **interface graphique** est nécessaire, prenant la forme d'un **serveur web**, sur lequel les opérateurs se connecteront via leur **navigateur web**.

ENSTO

Équipements de **distribution électrique** et solutions **d'automatisation des réseaux électriques**

MÉTHODES ET DÉVELOPPEMENTS



Le projet s'est divisé en 6 grandes étapes, de manière incrémentale :

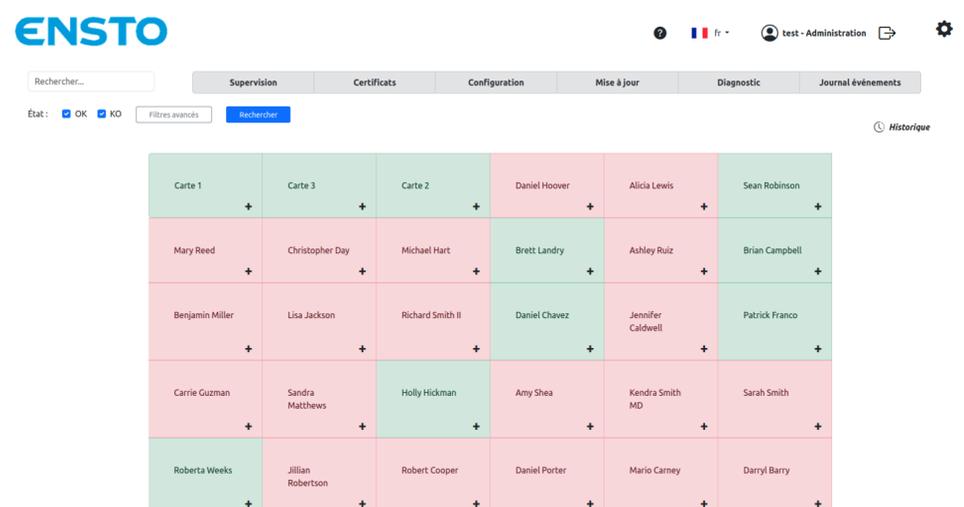
- **Analyse** du cahier des charges et **cadrage** du projet c'est à dire compréhension de la problématique et des besoins, conception de **l'architecture** et des choix techniques.
- **Développement** et **tests** de l'API, en utilisant le langage **C** et le framework **Ulfius**. La cross-compilation s'est faite à l'aide de **buildroot**. L'API respecte la norme de sécurité **MISRA C**.
- **Développement** et **tests** des pages web statiques, à l'aide d'**HTML**, **CSS**, et **Bootstrap**.
- **Développement** et **tests** du serveur web et des pages dynamiques, avec **Python** et le framework **Django**. Pour des raisons de sécurité, le serveur doit prendre en charge le protocole **HTTPS**. Il est déployé en utilisant **Nginx** et une base de données **PostgreSQL**.
- Rédaction des documentations, résumant comment **compiler**, **installer**, et **utiliser** l'API et le serveur. Une documentation du code a également été générée avec **Doxygen** pour l'API et **Sphinx** pour le serveur.
- **Tests complets** de l'API et du serveur dans les locaux d'**ENSTO**, puis légères **mises à jour** et correction des **bugs** détectés.

RÉSULTATS ET CONCLUSION

Aujourd'hui, il est possible pour les clients d'ENSTO de gérer un ensemble d'IEDs de manière **généralisée**, à l'aide d'une **interface** simple à utiliser.

De plus, l'installation de l'API et du serveur est rapide et intuitive. Pour l'API, son installation se fait via un **script** utilisant ssh et scp. Pour le serveur le déploiement prend la forme de **Dockers**, ou d'une **machine virtuelle** (au choix).

Enfin, la **documentation** permet une certaine compréhension du produit et du code, mais aussi comment faire des modifications et les intégrer au projet, ce qui assure une certaine **maintenabilité** pour l'équipe d'ENSTO.



Page d'accueil de l'interface graphique

MOTS-CLÉS : API REST, Serveur web, Réseaux, Cybersécurité, Développement embarqué