

Cahier des charges SolarPerform

1. Présentation générale

Nom du projet : SolarPerform

Commanditaire : Only Sun

Version : 1.0

Date : 25/03/2025

2. Contexte

Only Sun, **spécialisée dans l'installation de panneaux solaires** et la **distribution de solutions IoT pour le secteur de l'énergie**, souhaite développer une plateforme pour :

- Suivre la production et la consommation énergétique des installations
- Connecter et configurer des appareils IoT (capteurs, onduleurs, compteurs...)
- Centraliser les données de terrain (via SFTP ou MQTT)
- Offrir un portail de suivi énergétique aux utilisateurs finaux

3. Objectifs du projet

- Centraliser les données énergétiques issues de différents équipements
- Gérer la connectivité et la configuration des appareils à distance
- Offrir un dashboard clair de performance énergétique
- Automatiser l'intégration des appareils lors de l'installation sur site
- Structurer une infrastructure logicielle modulaire, scalable, sécurisée

4. Cibles

- Clients particuliers et entreprises ayant une installation solaire
- Techniciens installateurs (Suivie de client)
- Équipe technique interne (ops, support, monitoring)
- Responsable énergie / collectivités

5. Description fonctionnelle

Modules principaux :

Module	Fonction
Authentification	Gestion des comptes utilisateurs
Configuration	Association des équipements à un utilisateur
Monitoring	Visualisation des consommations et productions
Importation	Traitement de fichiers CSV via SFTP
Réception en temps réel	Via MQTT (topics utilisateur)
Supervision	Agrégation, anomalies, alertes
Admin	Gestion des accès, profils, logs

6. Architecture technique

- Microservices Node.js/Python
- PostgreSQL (schémas séparés)
- MQTT (Mosquitto)
- SFTP (serveur isolé)
- API Gateway (avec vérification JWT)
- Docker / Docker Compose
- CI/CD GitHub Actions
- Linux (Ubuntu)

7. Données manipulées

- Consommation électrique (kWh)
- Production (panneaux solaires)
- Données horodatées (timestamp, type de source)
- Données utilisateurs et appareils liés
- etc

8. Contraintes

- Communication entre appareils & backend → fiable et sécurisée
- Authentification via JWT
- Gestion multi source (SFTP et MQTT)
- Résilience & scalabilité

9. Livraison

- Cahier des charges validé
- Prototype de l'infrastructure
- Dockerisation complète
- Backoffice web de supervision
- API REST documentée