

# Infrastructure technique de SolarPerform

## Vue d'ensemble

SolarPerform est une plateforme composée de plusieurs microservices communiquant entre eux via une API Gateway. Les données sont reçues depuis des appareils connectés via MQTT ou SFTP, puis traitées et exposées via API REST sécurisée.

## Composants principaux

- **gateway**  
Point d'entrée unique de l'API REST. Vérifie les JWT (authentification) et route le trafic vers les microservices.
- **auth-service**  
Gère l'inscription, la connexion, les tokens JWT. Utilise PostgreSQL (auth.users).
- **user-service**  
Gère le profil utilisateur, les appareils. Utilisé lors de l'inscription pour lier un appareil. PostgreSQL (user.devices).
- **hardware-service**  
Configure automatiquement les appareils (via leur IP). Envoie les infos SFTP/MQTT au device via requête HTTP.
- **sftp-service**  
Crée/supprime des dossiers sur le serveur SFTP. Stocke le chemin dans la DB : PostgreSQL (sftp.paths).
- **files-service**  
Récupère, parse et insère les fichiers CSV du SFTP. Stocke les mesures : PostgreSQL (files.measurements).
- **mqtt-service**  
Gère le broker MQTT. Écoute les topics utilisateurs et insère les données dans PostgreSQL (mqtt.payloads).
- **monitoring-service**  
Expose les données agrégées. Utilisé par le frontend pour les dashboards. PostgreSQL (monitoring.daily\_stats, measurements).

## Base de données PostgreSQL

Tout est centralisé dans une seule base `solarperform`, avec des schémas séparés : auth, user, sftp, files, mqtt, monitoring

## Authentification et sécurité

Chaque utilisateur reçoit un token JWT lors de la connexion. Le gateway utilise un middleware `verifyToken` pour valider chaque requête. Les `userId` sont injectés dans les requêtes vers les services.

## Outils & Technologies

- JS pour les services
- PostgreSQL avec schémas par domaine
- Mosquitto MQTT pour les appareils connectés
- SFTP (atmoz/sftp) pour les fichiers CSV
- Docker & docker-compose pour le déploiement local
- GitHub Actions pour la CI/CD
- JWT pour la sécurité des APIs

## Arborescence du backend

backend/

```
|— gateway/  
|— auth-service/  
|— user-service/  
|— sftp-service/  
|— files-service/  
|— mqtt-service/  
|— hardware-service/  
|— monitoring-service/  
|— shared/  
|— docker-compose.yml  
└─ Makefile
```