**Automating Token Management for REE (ESIOS) API**

**Author: Amir Torbati**

**Objective**

The goal of this document is to outline the vision, motivation, and future plan to implement an AI-assisted system that manages the expiration and renewal process of the API token required for accessing the ESIOS API (provided by Red Eléctrica de España).

**Background**

Amir Torbati has developed a real-time data collection pipeline that fetches 15-minute resolution solar photovoltaic (PV) generation data from Spain. This data is used for modeling, analysis, and future battery optimization systems (BESS).

* The data is collected every 15 minutes.
* Stored in structured daily CSV files: data/YYYY-MM-DD.csv
* Synced automatically using GitHub Actions.
* Powered by the official ESIOS API which requires an access token.

**Token Dependency**

The API requires a personal token (API key), which:

* Is issued manually via [https://api.esios.ree.es](https://api.esios.ree.es/)
* Can expire or be revoked without warning
* Is essential for each API request

If the token becomes invalid:

* GitHub Actions will fail
* No data will be collected until a new token is received
* Missing data may accumulate until resolved manually

**Motivation to Automate Token Handling**

To reduce downtime, minimize data loss, and build a resilient system, we plan to:

1. Detect token expiration automatically (403 error handling).
2. Use an AI or scripted agent to send an email request to REE asking for a new token.
3. Monitor incoming emails for replies from REE.
4. Extract the new token using parsing or LLMs (e.g., GPT).
5. Update the GitHub repository's token via GitHub Secrets API.
6. Trigger a backfill script to fetch missing data.

This would make the data pipeline:

* Resilient
* Autonomous
* Self-healing

**What Will Be Automated in the Future**

| **Step** | **Description** |
| --- | --- |
| 1. | Monitor API responses and detect 403 errors |
| 2. | Automatically send email to REE using Gmail API |
| 3. | Check Gmail inbox regularly for replies |
| 4. | Extract token using regex or AI (ChatGPT/GPT-4) |
| 5. | Update GitHub Actions Secret using GitHub API |
| 6. | Trigger backfill script to fetch any missed data |

**Implementation Will Be Deferred**

This automation will be implemented after:

1. The historical data (from 2023) is fully collected
2. The full database is built (CSV, Parquet, DuckDB)
3. The append script is integrated

Only once the core pipeline is solid, will the token management AI agent be introduced to reduce manual steps.

**Why This Matters**

* Reduces data loss
* Avoids long downtimes due to expired tokens
* Saves manual effort
* Enables 24/7 autonomous operation
* Prepares the system for scaling and production

**Final Note**

This idea is not only practical, but visionary. It aligns with building intelligent, self-healing infrastructure for energy data systems. Amir Torbati is leading this design with foresight and technical rigor.

This document serves as the blueprint and justification for future AI-powered automation of the ESIOS token management process.