

System Specification Document

S.O.L.A.R. Project

January 5, 2026

Revision History

Version	Date	Author	Description
0.1	2025-12-17	Matteo	Initial draft
1.0	YYYY-MM-DD	Name	Approved release

Contents

1 Problem Definition	4
1.1 Business Problem	4
1.2 ML Problem Formulation	4
1.3 Key Performance Indicators (KPIs)	4
2 Data Specification	4
2.1 Data Source	4
2.1.1 Data Flow	4
2.1.2 Data Quality and Preprocessing	4
3 Functional Requirements	4
3.1 Use Cases	4
3.2 Functional Requirement List	5
4 Non-Functional Requirements	5
4.1 System Architecture	5
4.1.1 Training	5
4.1.2 Validation	5
4.1.3 Deployment	5
4.1.4 Monitoring	5
5 Risk Analysis	5

1 Problem Definition

1.1 Business Problem

Solar power plants are subject to strong variability due to weather conditions. Inaccurate short-term forecasts of power generation may cause grid imbalance penalties and inefficient energy dispatch. The objective of this project is to provide accurate and reliable predictions of solar power output to support operational decision-making.

1.2 ML Problem Formulation

Given historical weather data (temperature, irradiance, humidity, wind speed, cloud coverage), predict the continuous value of electrical power output (kW) for a solar plant at a given time.

This is formulated as a supervised regression problem.

1.3 Key Performance Indicators (KPIs)

- Root Mean Squared Error (RMSE)
- Mean Absolute Error (MAE)
- Coefficient of Determination (R^2)
- Business KPI: reduction of forecasting error compared to a persistence baseline

2 Data Specification

2.1 Data Source

- Weather variables: temperature, solar irradiance, humidity, wind speed
- Target variable: generated power (kW)
- Timestamp information

2.1.1 Data Flow

1. Raw data ingestion from CSV files
2. Schema validation and missing value checks
3. Feature engineering and normalization
4. Dataset splitting into training, validation, and test sets

2.1.2 Data Quality and Preprocessing

Missing values are handled through interpolation or forward-filling. Outliers caused by sensor malfunctions are detected using statistical thresholds. Numerical features are standardized to improve model convergence. Temporal features such as hour of day and day of year are extracted.

3 Functional Requirements

3.1 Use Cases

See in real time the power produced by the power plant. See how much power will be predicted in the same day and next day

3.2 Functional Requirement List

ID	Description
FR-01	The system shall ingest historical data.
FR-02	The model shall forecast timestamp predictions for the next 24 hours.
FR-03	The system shall forecast total energy yield for the next 24 hours.
FR-04	The system shall detect anomalies in the power production.
FR-05	The system shall identify underperforming inverters.
FR-06	The system shall display the list of underperforming inverters and their efficiency loss.
FR-07	The systems shall display the plot for the total predicted power generation against the actual one.
FR-08	The systems shall display the plot for the predicted power generation for each timestamp against the actual one.

4 Non-Functional Requirements

ID	Description
NFR-01	The forecasting model shall achieve a Mean Absolute Error (MAE) of less than a threshold on the validation set.
NFR-02	The anomaly detection algorithm shall have a low False Positive Rate (FPR) to prevent unnecessary maintenance dispatch.

4.1 System Architecture

4.1.1 Training

Offline model training is performed using scikit-learn pipelines.

4.1.2 Validation

Cross-validation is applied to reduce overfitting and assess generalization.

4.1.3 Deployment

The trained model is containerized using Docker and deployed via a FastAPI service.

4.1.4 Monitoring

The system monitors prediction errors and detects feature drift using statistical tests.

5 Risk Analysis

Risk	Impact	Mitigation
Data drift	Model degradation	Periodic retraining
Missing data	Incorrect predictions	Validation checks
Overfitting	Poor generalization	Cross-validation