TURBINE HEALTH PLATFORM - TEST CASE REPORT

Date: 24-09-2025

This document provides a summary of the test cases executed for the Turbine Health Platform. All test cases have passed successfully.

TEST CASE DESCRIPTIONS

[TC_API_001] GET /health-summary

Objective : Verify API returns aggregated fuel usage & decay status.

Pre-condition : Fast API application is running.

Test Steps :

1. Send GET request to /health-summary endpoint.
2. Observe the response payload.

Expected Result : Status 200 OK with JSON containing 'fuel_usage' key.

Actual Result : As Expected.

Status : PASS

[TC_API_002] GET /sensor-metrics

Objective : Verify API returns latest telemetry for turbine ID.

Pre-condition : Turbine ID exists in the database.

Test Steps :

1. Send GET request to /sensor-metrics?turbine_id=1.
2. Observe response payload for telemetry values.

Expected Result : Status 200 OK with JSON containing temperature, pressure metrics.

Actual Result : As Expected.

Status : PASS

[TC_API_003] POST /anomaly-alerts

Objective : Verify anomaly alerts are logged successfully.

Pre-condition : FastAPI app and DB logging are enabled.

Test Steps :

1. Send POST request with anomaly JSON payload.
2. Observe confirmation message in response.

Expected Result : Status 200 OK with confirmation message.

Actual Result : As Expected.

Status : PASS

[TC_DATA_001] CSV Ingestion

Objective : Validate ingestion of CSV logs into DataFrame.

Pre-condition : Sample CSV file available.

Test Steps :

1. Load CSV into Pandas DataFrame.
2. Check DataFrame structure and non-empty records.

Expected Result : Non-empty DataFrame with correct columns.

Actual Result : As Expected.

Status : PASS

[TC_DATA_002] Data Cleaning - Null Handling

Objective : Verify null values are handled correctly.

Pre-condition : DataFrame with missing values exists.

Test Steps :

1. Apply forward fill on missing values.
2. Verify no NULL values remain.

Expected Result : Clean DataFrame without NULLs.

Actual Result : As Expected.

Status : PASS

[TC_KPI_001] Efficiency Calculation

Objective : Check correctness of turbine efficiency formula (Power/Fuel).

Pre-condition : Valid input DataFrame with power and fuel columns.

Test Steps :

1. Apply efficiency formula: Power_Output / Fuel_Input.
2. Calculate average efficiency.

Expected Result : Correct average efficiency value.

Actual Result : As Expected.

Status : PASS

Conclusion:

All test cases executed successfully. The Turbine Health Platform APIs, data ingestion, cleaning, and KPI calculations meet the defined acceptance criteria and are ready for release.