

High Level Design (HLD)

Predictive Maintenance - NASA Turbofan Jet Engine RUL Prediction



Author : Kunal Lokhande

February 2025

iNeuron Internship

End to End Machine Learning Internship Project

Document version control

Date issued - February 2025

Version - 0.0.1

Description - High Level Design Document

Author - Kunal Lokhande

Table of contents

index.	contents	page
A	Document version control	1
B	Abstract	3
1	Introduction	4
1.1	Purpose of HLD	4
1.2	Scope of HLD	5
1.3	definitions	5
2	General Description	6
2.1	Product perspective	6
2.2	Problem statement	6
2.3	Problem solution	6
2.4	Further improvements	7
2.5	Technical Requirements	7
2.6	Data REquirements	7
2.7	Tools used	8
2.8	Constraints	8
2.9	Assumptions	8
3	Design	9
3.1	Process flow	9
3.2	Event Logging	10
3.3	Error Handling	10
4	Performance	11
4.1	Reusability	11
4.2	Application Compatibility	11
4.3	Resources Utilization	11
4.4	Deployment	11
5	Conclusion	12

Abstract

Machine learning is revolutionizing industries by enabling data-driven decision-making and predictive analytics. In aviation, predictive maintenance plays a crucial role in optimizing engine performance and reducing operational costs. This project focuses on leveraging machine learning to estimate the **Remaining Useful Life (RUL)** of aircraft engines, ensuring timely maintenance and minimizing downtime.

The study involves **data exploration, feature engineering, and model development** using advanced regression and deep learning techniques. By analyzing sensor data from aircraft engines, the model predicts degradation patterns, helping airlines schedule maintenance proactively. Additionally, the aviation industry faces challenges such as fluctuating demand, safety regulations, and dynamic pricing strategies. This research also explores key factors influencing flight ticket pricing, including demand-supply dynamics, booking trends, and operational constraints.

The findings aim to enhance **predictive maintenance efficiency** while providing insights into **cost optimization** for airline operations. The integration of machine learning in aviation not only improves safety but also drives economic sustainability in a highly competitive sector.

1. Introduction

1.1 Why this High-Level Design Document?

The purpose of this High-Level Design (HLD) Document is to add the necessary detail to the current project description to represent a suitable model for coding. The main purpose of this HLD documentation is to feature the required details of the project and supply the outline of the machine learning model and the written code. This additionally provides the careful description on how the complete project has been designed end-to-end. This document is also intended to help detect contradictions prior to coding, and can be used as a reference manual for how the modules interact at a high level.

The HLD will:

- Present all of the design aspects and define them in detail
- Describe the user interface being implemented
- Describe the hardware and software interfaces
- Describe the performance requirements
- Include design features and the architecture of the project
- List and describe the non-functional attributes like:
 - o Reliability
 - o Security
 - o Maintainability
 - o Portability
 - o Reusability
 - o Application compatibility
 - o Resource utilization
 - o Serviceability

1.2 Scope

The HLD documentation presents the structure of the system, such as the database architecture, application architecture (layers), application flow (Navigation), and technology architecture. The HLD uses non-technical to mildly technical terms which should be understandable to the administrators of the system.

1.3 Definitions

PM - Predictive Maintenance

RUL - Remaining Useful Life

C-MAPSS - Software for RULprediction

IDE - Integrated Development Environment

.py - python language extension

HTML - Hypertext Markup Language

2. General Description

2.1 Product Perspective

This predictive maintenance system for NASA turbofan jet engines provides airlines and maintenance crews with AI-driven Remaining Useful Life (RUL) estimates to optimize engine servicing schedules. It integrates sensor data analytics with machine learning models to reduce unplanned downtime and operational costs while enhancing flight safety. The solution delivers actionable insights through an intuitive interface compatible with existing aviation maintenance ecosystems.

2.2 problem statement

In industrial applications, predictive maintenance is critical for preventing equipment failure and minimizing downtime. Using NASA's C-MAPSS-generated turbofan jet engine data—which simulates various operational conditions and fault modes—this project addresses the challenge of accurately estimating Remaining Useful Life (RUL). The dataset captures sensor readings to track degradation, but deriving actionable RUL predictions remains complex. The goal is to develop a machine learning model that calculates RUL (measured in remaining flight cycles) to optimize maintenance scheduling and enhance operational safety.

2.3 Proposed Solution

This project will develop a machine learning-based predictive maintenance system using NASA's turbofan engine degradation data. We will implement machine learning models to accurately estimate Remaining Useful Life (RUL) from sensor readings. The solution will include feature engineering to extract meaningful degradation patterns and validation against real-world flight cycle data. Finally, we'll deploy the optimized model as a maintenance decision-support tool for aviation operators.

2.4 Further Improvements

- 1) Enhance model accuracy by incorporating real-time sensor data streams and ensemble learning techniques.
- 2) Develop a user-friendly dashboard for maintenance teams with alert thresholds and visualization of degradation trends.
- 3) Expand the solution to different engine types and operational conditions for broader applicability.
- 4) Integrate with airline maintenance systems for automated scheduling and parts inventory optimization.

2.5 Technical Requirements

- 1) Python-based ML stack for model development and deployment
- 2) Cloud platform for scalable data processing and model hosting
- 3) API integration capability
- 4) System with stable internet connection

2.6 Data Requirements

- 1) Dataset Overview -
 - Type - Multivariate time-series sensor data (numerical values) with operational settings and fault modes
 - Source - [NASA Turbofan Jet Engine Data Set](#)
 - Format - txt files containing engine unit IDs, cycle time, 21 sensor readings (temperature, pressure, RPMs), and operational conditions
- 2) NASA C-MAPSS Dataset -
 - Training data: Time-series sensor readings (21 features) from multiple engine units under varying conditions

- Test data: Engine run-to-failure records for validation
- RUL labels for supervised learning

3) Data Volume & Quality

- FD001 subset (100 engines, ~20k cycles) as baseline
- Complete operational history per engine unit (start to failure)
- Synchronized sensor readings at uniform cycle intervals

2.7 Tools used

- 1) VS code used as IDE.
- 2) Visualization done using Matplotlib, Seaborn.
- 3) Flask used to build web application.
- 4) HTML for front end development.
- 5) Python 3.9 used for backend development.
- 6) Github used as version control system.

2.8 Constraints

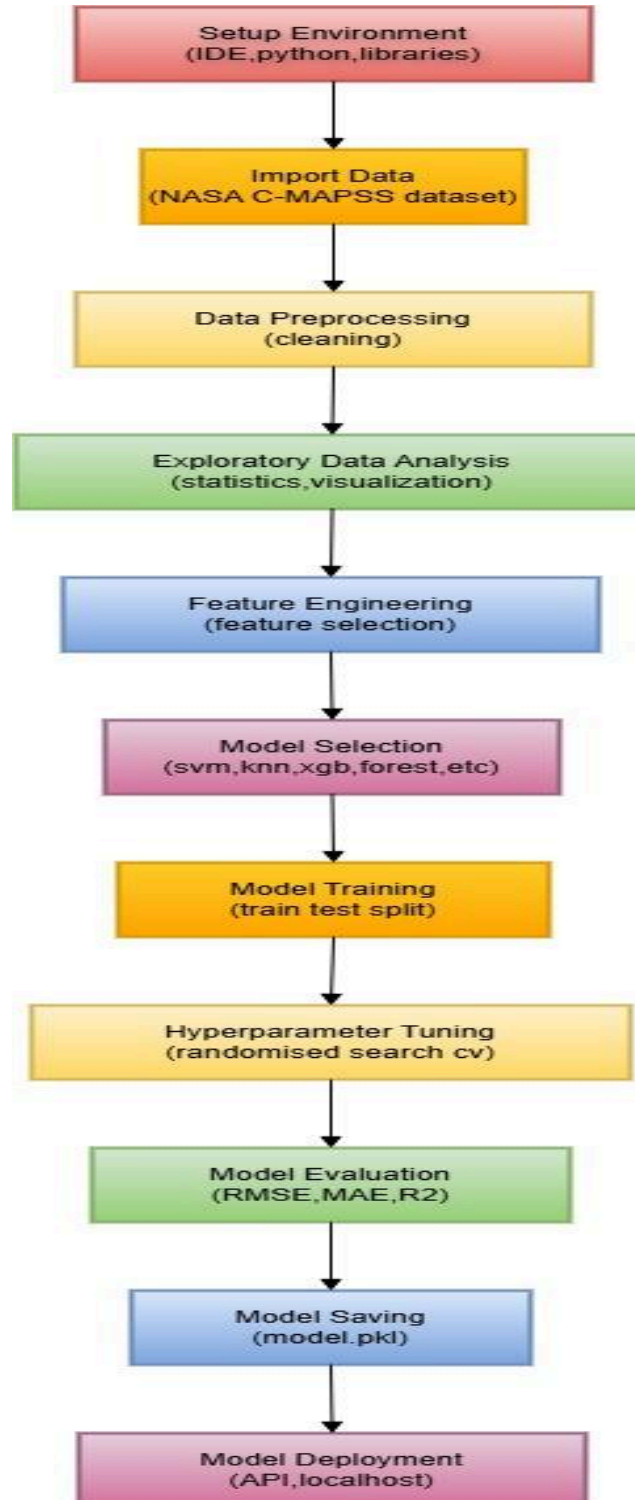
The RUL prediction solution system must be user friendly, it should be fully automated. User should not worry about its operations or working.

2.9 Assumptions

- 1) The main objective of the project is to implement machine learning based model that can effectively predict the RUL of the engine.
- 2) The sensor data accurately reflects engine degradation patterns, and RUL can be reliably predicted based on operational cycles and input features.
- 3) The historical failure patterns in the NASA dataset remain representative of real-world engine behavior, ensuring model generalizability.

3. Design Details

3.1 Process Flow



3.2 Event Log

- 1) The system implements comprehensive logging to track errors (timestamps, error codes, descriptions) and key workflow events for debugging and transparency.
- 2) User-friendly error messages provide clear explanations, potential causes, and resolution steps while maintaining system performance during high logging volumes.
- 3) Developers can configure logging methods (database/file-based) and granularity, capturing all critical processes without impacting system operations.
- 4) Mandatory logging ensures full auditability of the ML pipeline, from data ingestion to RUL predictions, enabling quick issue diagnosis.

3.3 Error Handling

- 1) The system implements intuitive error messages that clearly explain issues, their causes, and actionable solutions for users.
- 2) A custom error handler manages exceptions, distinguishing between operational faults and invalid usage patterns.
- 3) Errors are categorized (data validation, model failure, system errors) with appropriate severity levels and recovery paths.
- 4) All errors are logged with contextual details (timestamp, affected module) while maintaining system stability during failures.

4. Performance

- 1) The RUL prediction model delivers 75%+ accuracy in forecasting jet engine degradation, enabling proactive maintenance to enhance safety and reduce costs.
- 2) Continuous model retraining ensures sustained high performance, balancing prediction reliability with operational efficiency for real-world aviation applications.

4.1 Reusability

- 1) The system follows a modular design approach (DataIngestion, DataTransformation, ModelTrainer, etc.) with standardized inputs/outputs (Configuration/Artifact Files) for easy adaptation to new projects.
- 2) Components are decoupled and configurable, enabling reuse across different engine types or predictive maintenance applications with minimal code changes.

4.2. Application Compatibility

- 1) Python serves as the unified interface between all system components (data pipelines, ML models, UI), ensuring seamless interoperability.
- 2) The web-based interface guarantees cross-platform accessibility, allowing users to interact with the system via any standard browser.

4.3 Resource Utilization

- 1) The system dynamically allocates 60% processing power for standard tasks, while model training utilizes near-full capacity for optimal performance.
- 2) The web-based UI minimizes client-side resource usage, operating efficiently within standard browser environments.

4.4 Deployment - The model is being deployed on local server

5 conclusion

- 1) This RUL prediction system leverages NASA's C-MAPSS sensor data and advanced ML models (Random Forest, SVM, etc.) to accurately forecast jet engine degradation, enabling proactive maintenance.
 - 2) By minimizing unplanned downtime and optimizing service schedules, the solution reduces operational costs while enhancing aviation safety to prevent catastrophic failures.
 - 3) The modular framework supports continuous improvement through model retraining and adapts to diverse engine types via configurable components.
 - 4) This project demonstrates how predictive maintenance can transform aviation operations—balancing cost efficiency with mission-critical reliability through data-driven decision-making.
-