



VerveBridge

Crafting Careers, Empowering Tomorrow

Predictive Maintenance System (Python)

Objective:

The goal is to develop a Predictive Maintenance System using Python to predict equipment failures before they occur, allowing for proactive maintenance and reducing downtime.

What You Need to Do:

- **Research:**

Explore existing predictive maintenance systems and their features. Review relevant Python libraries and machine learning techniques for predictive maintenance (e.g., scikit-learn, TensorFlow, pandas).

- **Data Collection:**

Obtain historical data on equipment usage, maintenance records, sensor readings, and failure logs.

Ensure data is clean, consistent, and well-organized for analysis.

Data Analysis:

- **Exploratory Data Analysis (EDA):**

Perform EDA to understand the data distribution, identify patterns, and detect anomalies.

- **Feature Engineering:**

Create relevant features from raw data to improve model performance.

Implement techniques like rolling averages, time-based features, and statistical aggregations.

Model Development:

- **Model Selection:**

Choose appropriate machine learning models for predictive maintenance (e.g., Random Forest, XGBoost, LSTM).

- **Model Training:**

Train models on historical data and validate their performance.

- **Model Evaluation:**

Evaluate models using metrics such as accuracy, precision, recall, and F1-score.

Model Tuning:

Fine-tune model hyperparameters to improve performance.

Deployment:

- **Real-time Predictions:**

Develop a system to provide real-time predictions on equipment health.

- **User Interface:**

Create a dashboard to visualize predictions, historical data, and maintenance schedules.

- **Alerts and Notifications:**

Implement an alert system to notify maintenance teams of potential failures.

Testing:

- **Functionality Testing:**

Verify all features work as intended.

- **Performance Testing:**

Ensure the system performs efficiently under various loads.

- **Usability Testing:**

Get feedback from users to improve the interface and user experience.

Deliverables:

- **Research Summary:**

Summarize your findings on predictive maintenance systems and relevant techniques.

- **Data Analysis:**

Present your data analysis, including EDA and feature engineering.

Model Development:

Include model selection, training, evaluation, and tuning processes.

- **Code:**

The complete code for the Predictive Maintenance System.

Submission requirements:

1. Offer Letter:

After the introduction session, you have to upload the offer letter on LinkedIn between 10 am and 2 pm and also tag Vervebridge.

2. Task repo and video :

All the repos you create to upload the code on GitHub should be in the name of Vervebridge and all the videos you upload on LinkedIn should have the Vervebridge logo.

3. Architecture:

Whatever code you write, its details should be given in the comment above the code as to why that code is written.

4. Project demo video:

The project demo video should not be more than 2 minutes long and the Vervebridge logo should be mentioned in the video.

5. The project LinkedIn a post:

Whatever projects you get, you have to keep doing them on LinkedIn as they get completed and when all the projects are completed, you will get a certificate.

VerveBridge

Crafting Careers, Empowering Tomorrow

