A close-up of a logo

AI-generated content may be incorrect.

**INTEL AI MANUFACTURING CERTIFICATION COURSE**

1. **Project Overview**

**a. Project Title :**

Cost Forecasting for Sourced Components (Local vs Import)

**b. Project Description :**

This project aims to develop a machine learning-based system that forecasts the total cost of sourcing industrial components either locally or via import. It provides a predictive comparison based on various factors such as base cost, quantity, lead time, transport charges, customs duty, and currency rate.  
By utilizing historical data and regression/classification models, the system helps stakeholders make informed procurement decisions and evaluate whether importing is cost-effective for a specific component. This tool can significantly optimize sourcing strategy in manufacturing and supply chain operations.

**c. Timeline :**

| **Week** | **Milestone / Activity** |
| --- | --- |
| **Week 1** | 🔹 Finalize project scope & objectives  🔹 Collect or generate dataset  🔹 Clean & preprocess data  🔹 Initialize GitHub repository and upload base files (app.py  , dataset, README.md) |
| **Week 2** | 🔹 Train & evaluate regression model (cost prediction)  🔹 Build classification logic for import vs local  🔹 Calculate cost-effectiveness  🔹 Commit model training code and EDA results to GitHub |
| **Week 3** | 🔹 Develop Streamlit web app interface  🔹 Integrate model with form inputs  🔹 Test app predictions  🔹 Add visualizations (charts, boxplots)  🔹 Push completed Streamlit app to GitHub |
| **Week 4** | 🔹 Final bug fixes & improvements  🔹 Add debug output (cost/unit comparison)  🔹 Prepare project report & documentation (README.md, screenshots, usage instructions)  🔹 Host app via **Streamlit Cloud** using GitHub repo  🔹 Final presentation/demo |

**d. Benefits :**

* **Informed Decision-Making**: Helps procurement teams decide when import is cost-effective.
* **Cost Optimization**: Minimizes total procurement cost using data-driven forecasting.
* **Efficiency Improvement**: Reduces the manual effort of cost evaluation.
* **Global Sourcing Insights**: Analyzes how international variables like customs and exchange rates affect costs.
* **Predictive Analytics**: Leverages AI/ML to support proactive procurement strategy.

**e. Team Members :**

| **Name** | **Enrollment No.** |
| --- | --- |
| Mahek Bhathawala | 220840131006 |
| Kishan Mistry | 220840131054 |
| Roshan Patil | 220840131113 |
| Prathana Tandel | 220840131120 |

**f. Risks :**

| **Risk Area** | **Description & Mitigation** |
| --- | --- |
| **Data Quality Issues** | Incomplete or inconsistent data can reduce model accuracy.  ➤ Mitigation: Clean and validate dataset before training. |
| **Model Performance** | Regression/classification may underperform on edge cases.  ➤ Use cross-validation and ensemble methods. |
| **Currency Volatility** | Real-time exchange rate changes affect import pricing.  ➤ Include user-adjustable currency input. |
| **Scope Creep** | Adding too many features can delay delivery.  ➤ Keep MVP focused on forecasting and import evaluation. |
| **User Misinterpretation** | Users may not understand cost-effectiveness logic. ➤ Add tooltips, help guides, and debug info in UI. |

1. **Project Objective**

**a. Primary Objective**

To develop a machine learning-based system that accurately forecasts the total cost of sourcing industrial components either locally or via import, and determines whether importing is cost-effective — all through a user-friendly web interface.

**b. Secondary Objectives**

* Design and implement a **Streamlit web application** for real-time cost prediction and analysis.
* Enable **manual user input** and **CSV upload** to simulate real-world sourcing scenarios.
* Use **classification logic** to determine if import sourcing is more cost-effective than local sourcing.
* Provide **visual insights** (scatter plot, box plot) for comparing actual vs predicted costs and cost/unit for Local vs Import.
* Improve decision-making efficiency in **supply chain management** and **component procurement**.
* Host the project on **GitHub** and deploy it publicly via **Streamlit Cloud** for easy access and demonstration.

**c. Measurable Goals :**

| **Goal** | **Target Value / Metric** |
| --- | --- |
| Achieve accurate **cost prediction model** | R² score ≥ 0.80, RMSE within acceptable threshold |
| Determine import viability with **classification model** | Accuracy ≥ 85% on validation data |
| Ensure app **usability and accessibility** | Fully working Streamlit app with live inputs |
| Enable **real-time decision support** | User gets prediction + import evaluation < 2 sec |
| Ensure version control and collaboration | Complete GitHub project with code, data, README |
| Successful **deployment of live app** | Streamlit Cloud hosting with public link accessible |

1. **Methodology**

**a. Approach :**

For this Cost Forecasting Project, we will adopt the **Agile methodology** to allow iterative progress with flexibility for adjustments based on feedback. Agile’s incremental delivery enables early identification of issues, continuous improvement, and stakeholder involvement throughout the project lifecycle.  
This approach suits the evolving nature of data analysis and machine learning model development, where data insights and model performance can guide refinements continuously.

**b. Phases :**

The project will be completed through the following key phases:

1. **Requirements Gathering**
   * Understand business needs related to local vs import cost forecasting
   * Define scope, objectives, and key features
   * Identify data sources and necessary datasets
2. **Design**
   * Design data preprocessing workflows
   * Select and design forecasting models (regression/classification hybrid if needed)
   * Plan architecture for data storage, processing, and model deployment
3. **Development**
   * Data cleaning and feature engineering
   * Model training and evaluation
   * Develop scripts or software components for forecasting
   * Implement data visualization dashboards or reports
4. **Testing**
   * Validate model accuracy and robustness using test datasets
   * Perform unit testing on software components
5. **Deployment**
   * Deploy forecasting tools and dashboards to production environment
   * Provide user training and documentation
   * Monitor model performance and update as needed

**c. Deliverables :**

* **Requirements Phase:** Requirement specification document, data source inventory
* **Design Phase:** Data pipeline and system design documents, model architecture design
* **Development Phase:** Cleaned dataset, feature-engineered dataset, trained model files, codebase (scripts/programs)
* **Testing Phase:** Test cases, test reports, validation metrics (accuracy, RMSE, etc.)
* **Deployment Phase:** Deployed application or dashboard, user manual, training materials

**e. Testing and Quality Assurance :**

* **Data Validation:** Ensure completeness, consistency, and accuracy of datasets
* **Model Evaluation:** Use metrics such as accuracy, precision, recall, RMSE, and cross-validation to verify model performance
* **Software Testing:** Conduct unit tests and integration tests on software components
* **User Acceptance Testing (UAT):** Gather feedback from stakeholders to ensure deliverables meet business needs
* **Continuous Monitoring:** Track model predictions post-deployment to detect drift or degradation

**f. Risk Management :**

* **Data Quality Risks:** Mitigated by rigorous data cleaning, validation, and sourcing from multiple reliable providers
* **Model Performance Risks:** Use multiple models and ensemble approaches; continuous retraining and validation
* **Project Schedule Risks:** Use Agile sprints for flexible planning and early deliverable reviews
* **Technical Risks:** Maintain backup environments and version control for code and models