**Cost Prediction Model – Proof of Concept UI (Mockup)**

**Objective:**

Develop a **static C#-based UI mockup** (WinForms or ASP.NET) that simulates a cost prediction model for the construction industry, specifically for evaluating the costs of data center projects. The goal is to demonstrate a **proof of concept** that allows **cost accountants** to train the model and obtain predictions based on structured input data.

**Key Features and Requirements:**

**1. Technology Preference**

* **Primary options:** C# **WinForms** or **ASP.NET Web Application**
* Excel VBA acceptable if time-constrained, but C# is preferred
* Windows Server VM will be provided for web-hosted mockup if needed

**2. User Roles**

* **Cost Accountants**: Primary users during estimation and training – non-IT, but domain experts.
* **IT Users**: May be involved in initial rule validation or extended configuration (future phase).

**3. User Interface Design**

* **Style:** Desktop-style UI, simple and intuitive (no real-time interactivity required)
* **Inputs:**
  + Manually enter data for **one project at a time**
  + Minimum of **4 mandatory fields** (as defined in Excel)
  + Additional optional fields visible and usable
* **Outputs:**
  + Display **total cost predictions** for fields AX, AY, and BA
  + Use **grid view** to display full row prediction (e.g., vertically or horizontally)
  + Exporting to Excel is acceptable but not required for this phase
* **Interaction:**
  + A **"Predict Costs"** button initiates the prediction process
  + After prediction, show **suggested cost driver rules** with Yes/No radio buttons per rule for user review
  + Cost accountant can **accept or reject rules**

**4. Mock Data & Model Training**

* Use **synthetic project data** derived from the client's Excel file (3500 rows)
* Model should simulate:  
    
  + Detecting **correlations between input fields and costs**
  + Generating **rule-based insights**, e.g., “Cable tray length correlates with area”
  + Displaying **AI-suggested rules** and allowing manual approval/rejection
* Option to **manually add new rule logic** (e.g., “for OFFICE BUILDINGS, number of fixtures is based on surface”)

**5. Backend Logic (Simulated for Mockup)**

* No real AI model needed yet—hardcoded or rule-based logic can simulate behavior
* Example rules:  
    
  + If Year = 2010 and Surface = 5000m² → Apply inflation coefficient
  + If Type = "DATA CENTER" → Transformer count usually ≥ 4

**6. Training Feedback Mechanism**

* After prediction:  
    
  + List all AI-suggested cost driver correlations
  + For each: “Do you confirm this rule?” → [Yes] [No]
* Store responses temporarily (no database integration required)
* This interaction mimics model training guided by domain experts

**7. Deliverables**

* Fully functional **C# WinForms** or **ASP.NET static prototype**
* Clean, organized code with comments
* Support files (if applicable): Excel import logic, sample dataset
* **Deadline for delivery: May 20, 2025**
* **Demo Date:** May 26, 2025

**8. Post-Mockup Notes**

* No need to persist outputs; temporary display is enough
* No need for branding or color scheme compliance now
* Final model integration and interactive UI will be covered in a **separate contract**

**Optional Enhancements (if time permits)**

* Export results as Excel
* Allow switching between horizontal and vertical result display
* Save accepted rules locally for mockup (e.g., XML/JSON)

This project is part of a larger project of course. This one is just a demonstrator, a POC that we’ll demonstrate to our customers in May and we’ll move forward after that to extend it, to make it more flexible, to integrate it in our tools in the framework of other contracts.