# **Data Engineer Interview Challenge**

**Context**: Pricing and Shipping Optimization for a Medical E-Commerce Platform  
**Submission Format**:

* 1–2 page system design document (PDF)
* Code (GitHub repository or ZIP file)
* 60-second video demo

## **Overview**

In this challenge, you will design and prototype part of a data platform that supports dynamic product pricing and shipping cost estimation for a B2B e-commerce company operating in the life sciences and medical supply space.

The goal is not to build a complete system at scale but to understand how you think about system design, engineering trade-offs, and rapid prototyping. You are encouraged to use AI-assisted development tools such as GitHub Copilot, ChatGPT, or others to speed up your process. If you do, please indicate where and how you used them.

We recommend incorporating **Databricks** into your solution (e.g., for ETL, Delta Lake, MLflow, or orchestration), but you may use any tools and environments that enable you to demonstrate your thinking and technical ability.

## **Part 1: System Design (1–2 pages)**

### **Scenario**

The company you are supporting operates a platform with thousands of SKUs and millions of transactions. It sources from hundreds of suppliers and serves scientific institutions, labs, and hospitals. The business is evolving from rule-based pricing and flat-rate shipping to a more intelligent system driven by real-time data.

You are tasked with designing a data architecture that supports **one or both** of the following:

1. **Dynamic Product Pricing**
   * Account for anchor-product awareness, cost margins, historical conversion rates, and category-specific price tiers
2. **Shipping Price Estimation**
   * Account for product weight, dimensions, supplier locations, delivery zones, and margin sensitivity

### **Data Sources**

Design your architecture to ingest and process the following types of data:

* Product metadata: SKU, category, packaging, manufacturer, fulfillment method
* Transactional data: price paid, quantity, timestamp, customer location
* Supplier data: cost, availability, shipping lead time
* Clickstream data: product impressions, add-to-cart behavior, conversions
* Market data: competitor pricing
* Shipping tables: rates based on weight, size, and distance

*Note: We will not provide you with sample data, and you are invited to generate sample data yourself for demonstration purposes. You can explore openly available datasets or generate plausible synthetic data. For example exploring datasets provided on kaggle could be an option.*

[*https://www.kaggle.com/datasets/thedevastator/unlock-profits-with-e-commerce-sales-data*](https://www.kaggle.com/datasets/thedevastator/unlock-profits-with-e-commerce-sales-data)

### **Deliverables**

Submit a **1–2 page PDF** that includes the following:

* A high-level architecture diagram (digital or hand-drawn)
* Description of the core components:  
  + Ingestion layer for batch and/or streaming data
  + Storage design (e.g., using Delta Lake) and partitioning strategy
  + Feature engineering and transformation logic
  + Experimentation support (e.g., A/B toggles, logs)
  + Serving layer for real-time or scheduled access to outputs
  + Monitoring and alerting setup for reliability and quality
* Where and how Databricks is integrated (if used)
* Reliability and scalability considerations
* Where you would leverage AI-assisted development tools
* What you would do differently with more time
* Any trade-offs you made and reasoning

## **Part 2: Prototype Implementation**

Choose **one** of the following two use cases to implement. You may use Databricks (Community Edition is acceptable), a local Spark setup, or another environment that allows you to demonstrate the logic effectively.

You are welcome to use AI-assisted development tools. Please note where they were used and how they supported your workflow.

While we encourage the use of Databricks, particularly Delta Lake and Notebooks, you may prototype using the tools of your choice.

### **Option A: Dynamic Product Pricing MVP**

* Load product and transaction data
* Identify anchor products based on popularity, margin, or conversion rate
* Suggest pricing for non-anchor products using anchor-aware price tiering within categories
* Write results to Delta Lake or Parquet format

### **Option B: Shipping Price Estimation MVP**

* Load product, shipping, and location data
* Estimate shipping cost based on weight, distance, and product dimensions
* Apply margin-aware logic (e.g., offer free shipping above margin threshold)
* Write results to Delta Lake or Parquet format

### **Submission Requirements**

* Submit your code via a **GitHub repository** or a **ZIP file**
* Include a README.md with:  
  + A summary of your solution
  + Setup instructions
  + How to run your pipeline
* Clearly comment and structure your code for readability
* Mention any AI-assisted coding tools used

## **Part 3: Demo & Technical Reflection**

### **60-Second Video**

Record a brief video (screen recording acceptable) that:

* Shows your system or prototype in action
* Explains the use case and logic
* Mentions what tools you used, including Databricks and AI assistants