Capstone Project:

Supply Chain Data Engineering & Analytics on Azure Cloud

Business Scenario

Project Overview

This capstone project replicates a real-world Supply Chain analytics workflow. Learners will ingest, transform, and analyze complex datasets covering shipment orders, vendor performance, warehouse inventory, delivery logs, and insurance claims. The project integrates Data Engineering, Machine Learning, API development, and Cloud deployment using Python, SQL, FastAPI, and Azure services.

Learners will build end-to-end data pipelines, apply analytics for shipment optimization, and visualize KPIs via Power BI dashboards. This simulation prepares learners for roles in supply chain data engineering, operations analytics, and ML engineering.

Business Context

Modern supply chains span continents, involving multiple vendors, warehouses, carriers, and compliance layers. Challenges like stockouts, delayed deliveries, cost overruns, and vendor risk demand robust data pipelines and predictive models.

A supply chain analytics engine that consolidates multi-source data—inventory, vendors, shipping, and claims—enables:

- Early warning on vendor delays
- Optimized inventory and reorder planning
- Shipment routing efficiency
- Warehouse-level performance monitoring
- Proactive fraud and damage claim detection

This capstone mimics an enterprise-grade analytics engine empowering data-driven decision-making in supply chain operations.

Project Objectives

1. Apply Python, SQL, and algorithms to ingest and process supply chain datasets.
2. Build ETL pipelines and warehouse models for vendor, inventory, delivery, and claims data.
3. Apply ML to forecast delivery delays, cluster vendor performance, and flag anomalies.
4. Expose insights and ML predictions using REST APIs (FastAPI).
5. Deploy complete pipelines and dashboards using Azure (ADF, Synapse, Databricks).

6. Apply governance and security best practices for enterprise readiness.

Project Dataset

1. Shipments Table
shipment_id, origin_warehouse, destination_city, ship_date, delivery_date, product_id, quantity, freight_cost

2. Vendors Table
vendor_id, vendor_name, product_id, contract_start, contract_end, vendor_rating, country

3. Inventory Table
warehouse_id, product_id, stock_level, reorder_threshold, last_restock_date, next_restock_due

4. Delivery Logs Table
delivery_id, shipment_id, carrier, status, delivery_duration_days, damage_flag, proof_of_delivery_status

5. Claims Table
claim_id, delivery_id, reason, amount_claimed, claim_status, claim_date, resolved_date

Relationships:

- shipment_id links Shipments and Delivery Logs
- delivery_id links Delivery Logs and Claims
- product_id links Vendors, Shipments, and Inventory

Solution Outline

1. **Data Collection & Preparation**: Load shipment, vendor, inventory, delivery, and claims data from flat files or simulated APIs.
2. **Data Modeling & Storage**: Design a relational schema using MySQL; apply constraints and indexes.
3. **Data Engineering & Transformation**: Build pipelines in Python & Pandas; normalize delivery durations, detect damaged shipments, enrich inventory status.
4. **Analytics & ML**:
    - Predict delivery delays
    - Segment vendors based on performance
    - Flag suspicious claims using classification

5. **API Development**: Expose pipelines and predictions using FastAPI.
6. **Cloud Deployment**: Deploy using Azure Data Factory, Synapse, and Databricks; visualize in Power BI.
7. **Governance & Security**: Apply RBAC, Key Vault, Purview for traceability and compliance.

Capstone Phases with learner Tasks & Deliverables

Stage 1: Algorithmic Thinking, Python & SDLC

Objective: Establish programming and problem-solving foundation.
Tasks:

- Write pseudocode for:
  - Freight cost calculation per shipment
  - Stockout detection logic
  - Vendor reliability score computation
- Implement Python scripts for:
  - Data cleaning and transformation
  - Basic aggregation (avg delay, claim %)
- Document SDLC artifacts (BRS, SRS, HLD)
- Participate in Python Hackathon on supply chain problems
  Deliverables:
- Flowcharts, pseudocode, Python notebooks, SDLC docs

Stage 2: Data Structures, UML & MySQL

Objective: Efficient modeling and querying of supply chain data.
Tasks:

- Implement data structures:
  - Product-stock mapping
  - Warehouse-product matrix
  - Vendor-claim risk scores
- UML diagrams:
  - Use case: Delivery & Claims
  - Class diagram: Inventory lifecycle
  - Activity: Vendor contract renewal
- MySQL:
  - Schema creation, normalization
  - Joins: vendor delays, claim % by carrier
- SQL Hackathon
  Deliverables:

- UML diagrams, MySQL schema/scripts, SQL reports, hackathon solution

Stage 3: Data Engineering & ML with Python

Objective: Build pipelines, run ML models, automate workflows.
Tasks:

- ETL pipelines for:
  - Claims enrichment
  - Vendor delay classification
  - Inventory health score
- ML Models:
  - Delivery delay prediction
  - Vendor segmentation (K-Means)
  - Claim fraud classification (LogReg/Tree)
- FastAPI endpoints:
  - GET: latest claim % by carrier
  - POST: new delivery logs

    Deliverables:
- ETL notebooks, ML models (accuracy, ROC, Silhouette), APIs, test cases

Stage 4: Azure Deployment & Visualization

Objective: End-to-end supply chain pipeline on cloud.
Tasks:

- Azure Storage + ADLS Gen2 for ingestion
- ADF pipelines: Vendor, Shipment, Claim ingestion
- Synapse queries: delay patterns, fraud hotspots
- Databricks notebooks for ML & prep
- Power BI dashboards:
  - Inventory health
  - Vendor performance
  - Shipment trends & delays

    Deliverables:
- Azure resources config, ADF pipelines, Synapse SQL, Power BI reports

Capstone Learning Schedule (Aligned to Topics)

Week 1

- Algorithm & Flowcharts, Git, SDLC, Python Basics, Exception Handling

- Deliverable: Flowcharts, Python cleaning scripts, SRS/HLD docs, Python Hackathon

Week 2

- Data Structures (List, Set, Dict, LinkedList), UML (Use Case, Class, Activity), MySQL
- Deliverable: UML diagrams, Normalized MySQL schema, SQL reports, DB Hackathon

Week 3

- NumPy, Pandas, SQL (JOIN, GROUP BY, Views), ML Intro (classification/clustering)
- Deliverable: Exploratory analysis scripts, SQL insights, clustering model

Week 4

- Spark (PySpark, RDD, DataFrames), FastAPI (CRUD, validation), Azure (ADF, Databricks, Power BI)
- Deliverable: ETL pipelines, ML models, REST APIs, dashboards, final deployment report

---

This capstone blends real-world data workflows with classroom learning across four weeks—guiding learners from data prep to ML deployment on the cloud.