



AN
NIIT
VENTURE

Capstone Project:

Supply Chain Data Engineering & Analytics on Azure Cloud

Business Scenario

Project Overview

Today's global supply chains are complex, involving multiple vendors, warehouses, transporters, and systems operating across regions. Delays, stockouts, and claim disputes can disrupt operations and escalate costs. Enterprises need cloud-first data platforms that consolidate and process data from shipments, inventory, vendors, delivery logs, and claims to drive visibility and performance.

This capstone simulates the development of a **cloud-native supply chain analytics platform** that manages order movement, inventory health, delivery status, and claims. You will build structured pipelines, serve REST APIs for real-time visibility, and integrate Azure services to simulate enterprise deployment.

Project Objectives

1. Define system design and data documentation using SDLC, SRS, HLD, and UML artifacts.
2. Ingest multi-source supply chain data (shipments, inventory, claims, etc.).
3. Apply SQL and Pandas for data cleaning, transformation, and summarization.
4. Build FastAPI microservices to expose key metrics like delivery status and claim history.
5. Use Azure Data Lake, Azure Data Factory, and simulated Synapse pipelines for cloud integration.
6. Visualize KPIs such as delivery duration trends and inventory health via dashboards (Power BI or Python).

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

Capstone Phases with learner Tasks & Deliverables

Phase 1: SDLC Documentation & System Design (SRS, HLD, UML)

Tasks:

- Draft **SRS** for the cloud-based supply chain analytics platform.
- Document **business and functional requirements** like shipment tracking and claim summarization.
- Create **HLD** showing system modules:
 - Shipment Tracker
 - Claims Monitor
 - Vendor Inventory Viewer
- Design **UML Diagrams**:
 - **Use Case Diagram**: Initiate shipment, file a claim, check inventory.
 - **Class Diagram**: Shipment, Vendor, Inventory entities.
 - **Activity Diagram**: Claim resolution workflow.

Phase 2: Data Engineering with Python & SQL

Tasks:

- Use **Python with Pandas** to:
 - Clean and merge shipment, delivery, and claim data.
 - Calculate delay duration, reorder flags, claim aging.
- Write **SQL scripts** for:
 - Joins across shipment–delivery–claim tables.
 - Aggregate freight costs by city, average claim amounts.
 - Grouped summaries by carrier or vendor.
- Use **Views and Subqueries** for reporting stock health or unresolved claims.

Phase 3: FastAPI Development for Supply Chain Metrics

Tasks:

- Develop **FastAPI microservices**:
 - GET /claims-summary → Return claim percentages per carrier.
 - GET /inventory-health → Return stock and reorder status.
 - POST /log-shipment → Add a new shipment record.
- Integrate **SQLAlchemy + Pydantic** models.
- Enable **file upload APIs** to import new delivery logs.
- Expose **automatic docs via Swagger/Redoc**.

Phase 4: Azure Cloud Integration (ADLS, ADF)

Tasks:

- **Data Lake Integration**:
 - Upload raw CSVs (Shipments, Vendors, Claims) to **ADLS Gen2**.
- **ADF Pipelines**:
 - Ingest data into bronze/silver zones.
 - Schedule periodic ingestion of inventory and delivery updates.
- **SQL**:
 - Perform SQL queries over structured supply chain data.
- **Dashboarding**:
 - Use **or Python** to visualize:
 - Stock level vs reorder threshold
 - Claim status across delivery carriers
 - Shipment delay trends over time