# Team 4:

# **Supply Chain Data Engineering & Analytics on Azure Cloud**

# **INDEX**

**1. Software Requirement Specification (SRS) ………………………………………….…………..2**

1.1 Introduction ……………………………………………………………………………………………...2

1.2 Overall Description……………………………………………………………………………………..3

1.3 Functional Requirements…………………….……………………………………………………...3

1.4 Non-Functional Requirements…………………….…………………….………………………...4

1.5 Assumptions & Dependencies……………………………………………………..……………...4

**2. High-Level Design (HLD) …………………….……………………………………………….………… 5**

2.1 System Modules…………………….…………………………………………………………..…..… 5

2.2 Data Flow with Azure Services…………………….……………………………………..……….. 5

2.3 HLD Diagram…………………….………………………………………………………..……………..6

**3. UML Diagrams…………………….…………………………………………………..……….….……… 7**

3.1 Use Case Diagram…………………….……………………………………………………….……… 7

3.2 Class Diagram…………………….………………………………………………………….………… 7

3.3 Activity Diagram…………………….……………………………………………….………………… 8

### 1. Software Requirement Specification (SRS)

**1.1 Introduction**

* **Purpose**  
  The purpose of this system is to develop a **cloud-based supply chain analytics platform** that enables real-time visibility into order movement, delivery status, claims, and inventory health.
* **Scope**  
  The system consolidates data from multiple sources — shipments, vendors, inventory, delivery logs, and claims — and provides analytics, dashboards, and APIs. The platform will:
  + Ingest multi-source data into **Azure Data Lake**.
  + Process and transform using **Azure Data Factory** and Python/SQL pipelines.
  + Expose **FastAPI microservices** for real-time insights.
  + Deliver Python **dashboards** for KPIs like shipment delays, inventory health status, and claims trends.
* **Definitions & Acronyms**
  + **ADLS** – Azure Data Lake Storage
  + **ADF** – Azure Data Factory
  + **KPI** – Key Performance Indicator
  + **API** – Application Programming Interface
* **References**
  + Project Specification (Capstone Project 3 1.pdf)
  + GFG Websites
  + You tube
  + Python Organization documentation
  + Azure service documentation.

**1.2 Overall Description**

* **Product Perspective**  
  The platform simulates an enterprise cloud-native supply chain data hub, replacing fragmented systems with a single integrated data and analytics solution.
* **Product Features**
  + Shipment Tracker: monitors shipment status and delays.
  + Claims Monitor: records, summarizes, and resolves claims.
  + Vendor Inventory Viewer: tracks stock levels and reorder requirements.
  + Dashboards: deliver real-time supply chain visibility.
* **User Characteristics**
  + **Admin**: manages ingestion pipelines, data uploads, API configs.
  + **Business Analyst**: consumes dashboards for decision-making.
  + **Vendor/Manager**: monitors shipments, stock, and claims.
* **Constraints**
  + Cloud-first deployment (Azure).
  + Must support real-time API queries.
  + Secure role-based access.

**1.3 Functional Requirements**

* **Shipment Tracking**
  + Track delivery timelines and calculate delays.
  + Provide freight cost summaries by city/carrier.
* **Claims Summarization**
  + Maintain lifecycle of claims (filed, pending, resolved).
  + Provide claim percentages per carrier.
* **Inventory Monitoring**
  + Detect reorder needs (stock vs. thresholds).
  + Show inventory aging and restock schedules.
* **Vendor Management**
  + Track vendor performance based on ratings and delivery history.
  + Monitor contract validity and compliance.

**1.4 Non-Functional Requirements**

* **Performance**: APIs respond without delay.
* **Scalability**: Capable of scaling for millions of shipment records.
* **Security**: Data should be encrypted.
* **Reliability**: It should be reliable.

**1.5 Assumptions & Dependencies**

* All data sources (shipments, vendors, claims) available in CSV format.
* Azure subscription with access to **ADLS, ADF, Synapse, Python Visualization**.
* Users have access credentials for APIs and dashboards.

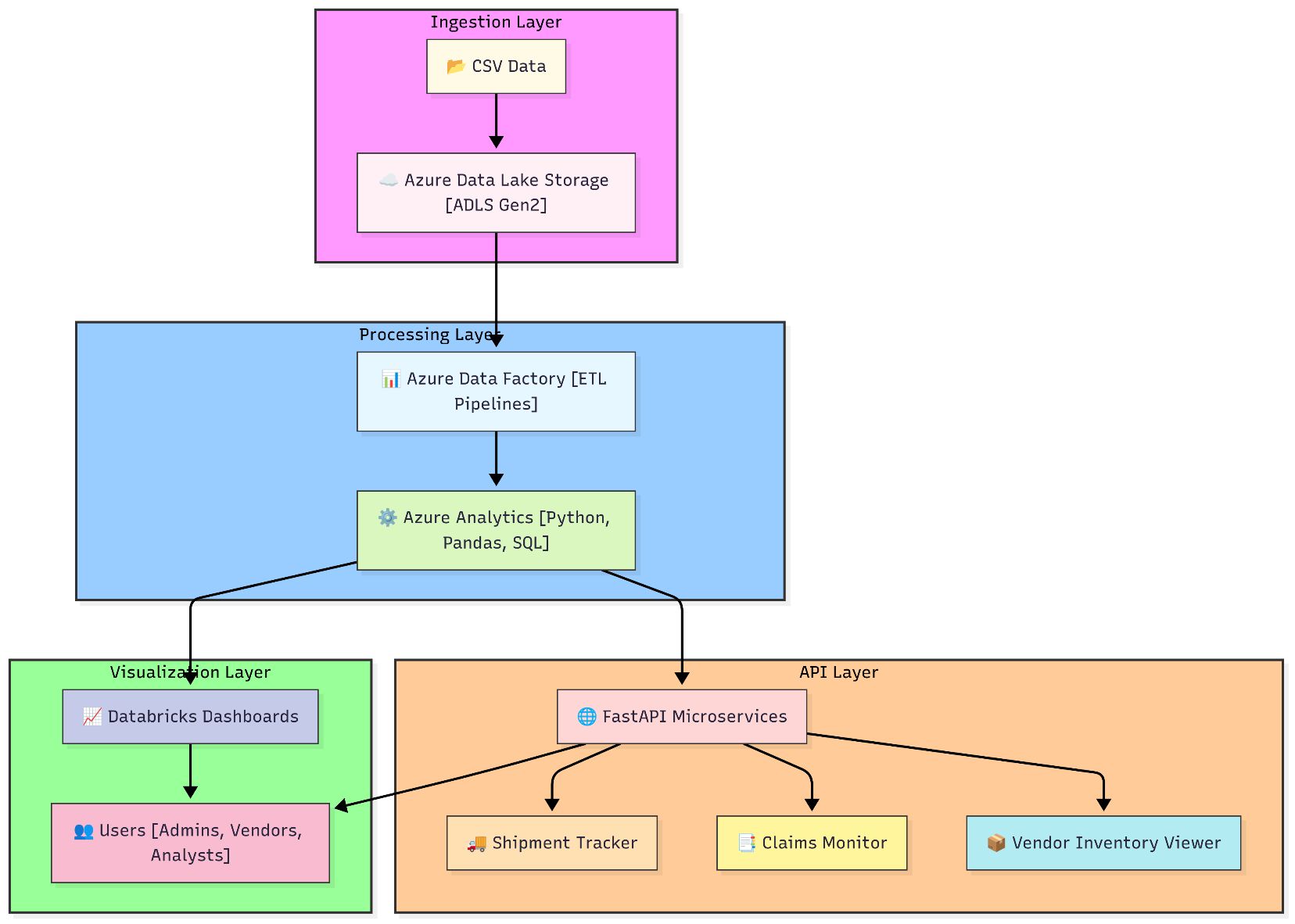
### 2. High-Level Design (HLD)

**2.1 System Modules**

* **Shipment Tracker**: Processes shipment and delivery logs, computes KPIs like delay duration.
* **Claims Monitor**: Processes claims linked to delivery logs, generates claim summaries.
* **Vendor Inventory Viewer**: Tracks stock levels, reorder needs, and vendor reliability.

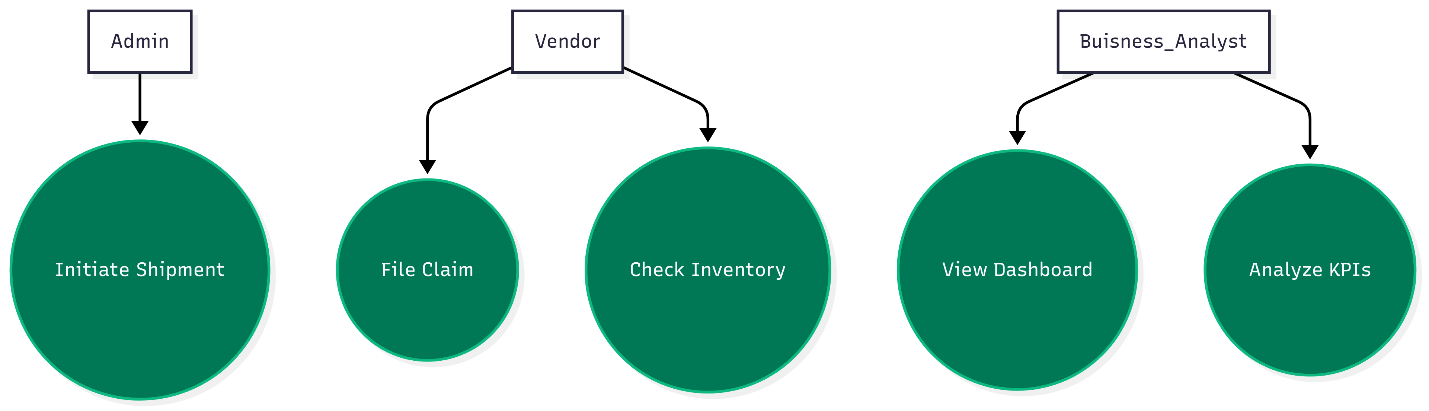
**2.2 Data Flow with Azure Services**

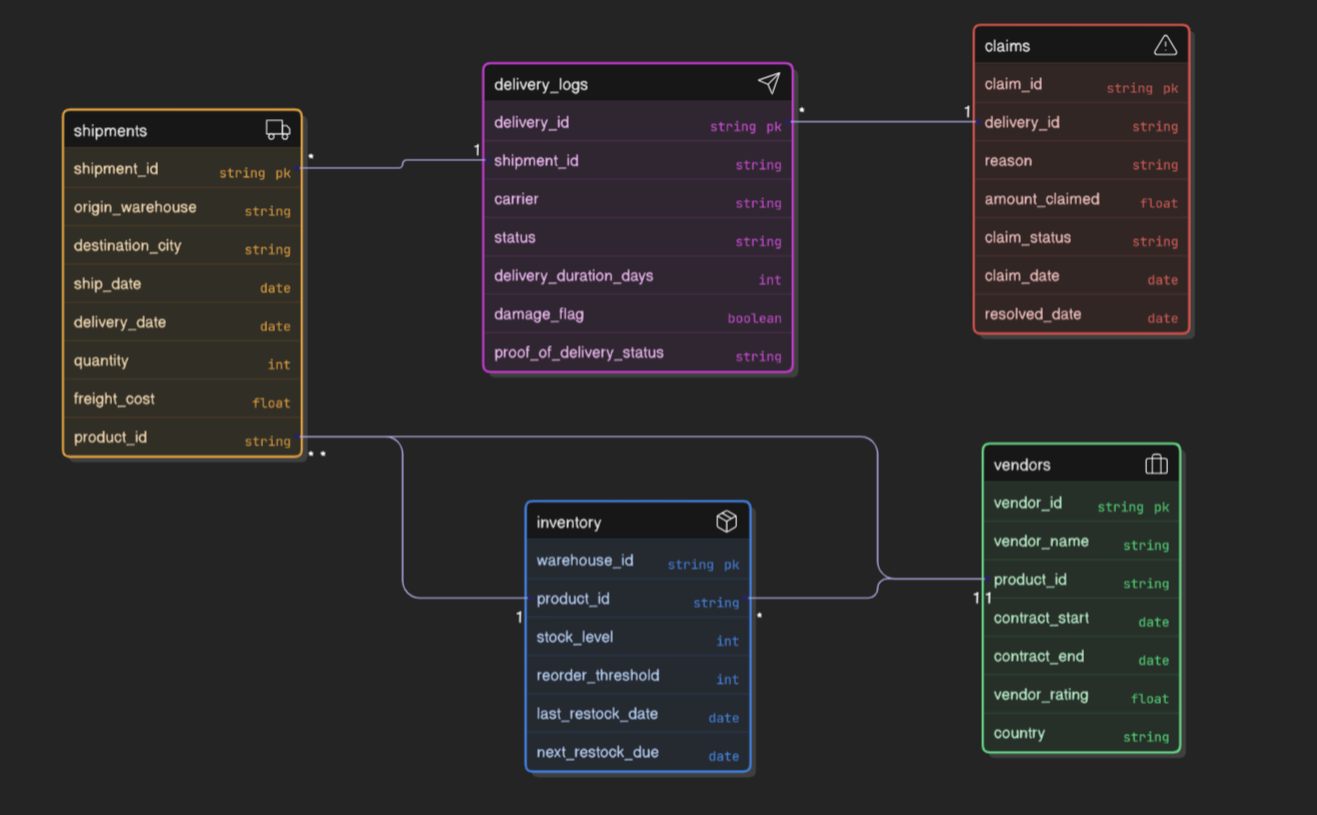
1. **Data Ingestion**
   * Raw shipment, vendor, inventory, claims data uploaded to **ADLS Gen2**.
   * **ADF pipelines** ingest and clean into bronze/silver/gold zones.
2. **Processing & Storage**
   * Data transformation with Python (Pandas) and SQL.
   * Processed datasets stored in **Azure Synapse Analytics**.
3. **API Layer**
   * **FastAPI microservices** expose endpoints:
     + /claims-summary
     + /inventory-health
     + /log-shipment
     + /Vendor Inventory viewer
4. **Visualization**
   * **Data Bricks visualization**
   * **FastAPI Microservices**
5. **HLD Diagram**



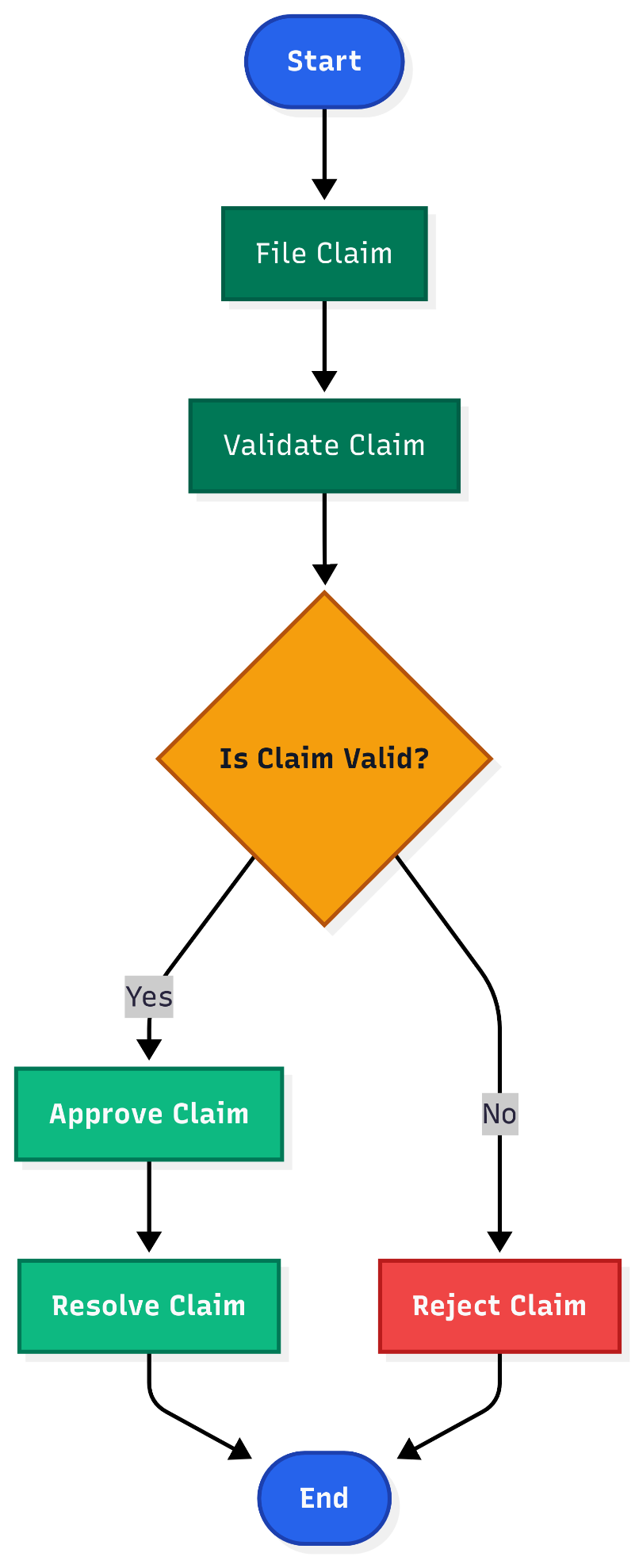
### 3. UML Diagrams

**3.1 Use Case Diagram**



**3.2 Class Diagram**

**3.3 Activity Diagram**

****