**Case Study: Customer Analytics Data Platform**

**Problem Statement**

A retail company wants to build a centralized **Customer Analytics Data Platform** to consolidate data from multiple sources, including transactional databases, web logs, and customer interaction systems. The goal is to enable data-driven insights for marketing campaigns, customer segmentation, and real-time recommendations. The existing process is fragmented, slow, and error-prone due to manual data handling and lack of automation.

**Skill Tower Developed in the Project**

This project will develop skills across multiple layers of data engineering:

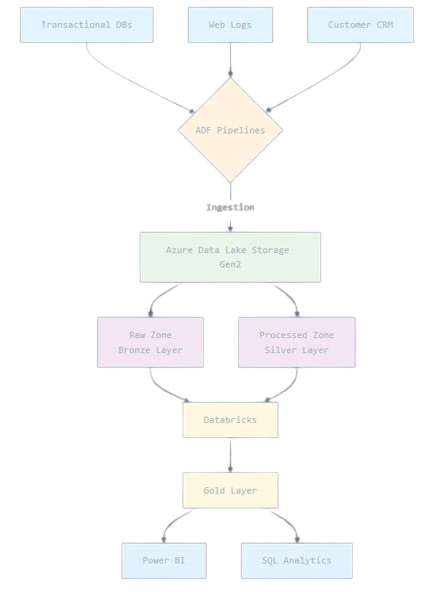
1. **Cloud Fundamentals**: Azure environment setup, resource hierarchy, cost management, Azure CLI & PowerShell usage.
2. **Data Storage & Management**: ADLS Gen2, Blob storage, Table storage, Queue storage, storage security & lifecycle policies, data redundancy.
3. **Data Ingestion & ETL**: Azure Data Factory pipelines, datasets, linked services, copy activities, incremental loads, error handling.
4. **Data Transformation**: Mapping and Wrangling Data Flows, transformations (Join, Aggregate, Conditional Split, Derived Column).
5. **Data Analytics & Lakehouse**: Databricks workspace, cluster management, notebooks (Python, SQL), Spark DataFrame transformations, Medallion Architecture.
6. **DevOps & Governance**: CI/CD integration with Git, monitoring, logging, RBAC, SAS/AD authentication.

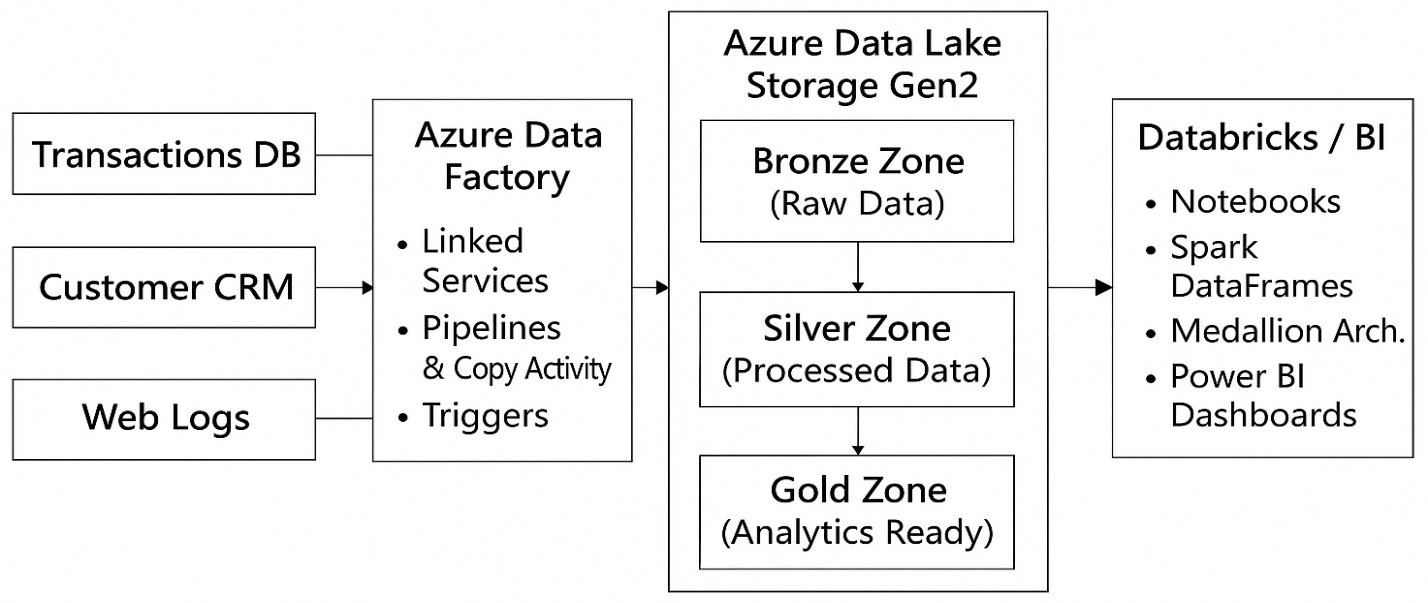
**Use Case / Architecture Diagram**

**Architecture Overview:**

**Key Components:**

* **Raw Layer (Bronze)**: Stores raw ingestion from various sources.
* **Processed Layer (Silver)**: Cleansed and transformed datasets.
* **Analytics Layer (Gold)**: Aggregated, ready-for-analytics tables for reporting.





**User Stories**

1. **As a data engineer**, I want to create an Azure Data Lake with proper hierarchical structure so that data is organized efficiently.
2. **As a data engineer**, I want to build ADF pipelines to copy data from source systems to ADLS so that raw data is ingested automatically.
3. **As a data engineer**, I want to implement mapping data flows to clean, transform, and aggregate data for analytics purposes.
4. **As a data analyst**, I want to query curated datasets in Databricks so that I can generate insights and dashboards.
5. **As a DevOps engineer**, I want to set up Git integration for ADF pipelines so that changes are version-controlled and deployable.
6. **As a security engineer**, I want to implement RBAC and SAS token access so that storage access is secure and auditable.

**Expected Deliverables**

1. **Azure Environment Setup**: Resource groups, subscriptions, storage accounts, access policies.
2. **Data Lake Structure**: Bronze, Silver, and Gold zones in ADLS Gen2.
3. **ADF Pipelines**: Full ingestion pipelines with copy activities, data flows, triggers, and monitoring.
4. **Databricks Workspace**: Clusters configured, notebooks implemented for data transformation and Medallion Architecture.
5. **Documentation**: User guide, architecture diagram, and operational runbook.
6. **Demo Reports**: Sample dashboards or visualizations using transformed datasets.

**Milestones and Duration**

| **Milestone** | **Duration** | **Description** |
| --- | --- | --- |
| Environment Setup | 1 hr | Provision Azure resources, storage accounts, configure RBAC. |
| Data Ingestion | 2 hr | Build ADF pipelines and copy raw data into ADLS. |
| Data Transformation | 3 hr | Implement Mapping/Wrangling Data Flows, Databricks transformations. |
| Analytics & Reporting | 2 hr | Create Gold tables and demo BI dashboards. |
| CI/CD & Monitoring | 1 hr | Git integration, pipeline triggers, monitoring & alerts. |
| Documentation & Review | 1 hr | Prepare architecture diagram, operational manuals, and deliverables. |

**Implementation Notes**

* **Azure Storage**: Use LRS for development and RA-GRS for production critical data.
* **ADF Pipelines**: Include retry policies and incremental data load strategy for efficient ETL.
* **Databricks**: Use Python notebooks for transformations; integrate Delta Lake for data versioning.
* **Monitoring**: Enable ADF metrics, pipeline logging, and Azure Storage diagnostics.
* **Security**: Use RBAC for users; prefer SAS tokens for temporary access; integrate with Azure AD where possible.
* **Cost Optimization**: Implement storage lifecycle policies to move older data to cool or archive tiers.

**Evaluation Rubrics**

| **Criteria** | **Weight** | **Evaluation** |
| --- | --- | --- |
| **ADF Pipeline Implementation** | 25% | Correct ingestion, transformations, triggers, error handling. |
| **Databricks Transformation & Lakehouse** | 25% | Bronze/Silver/Gold layer implementation, Spark transformations, Medallion architecture adherence. |
| **Azure Environment & Security** | 15% | Proper resource organization, RBAC, SAS/AD integration. |
| **Data Governance & CI/CD** | 15% | Git integration, versioning, logging, monitoring setup. |
| **Documentation & User Stories Fulfilment** | 10% | Clear architecture diagrams, runbooks, user story coverage. |
| **Reporting & Analytics** | 10% | Correct aggregation, insights generation, sample dashboards. |