Building a Real-Time Weather Data Pipeline for Weather Analytics

Real-Time Weather Data Pipeline – AWS Kinesis, Lambda, Redshift
[Bootcamp – AWS Data Engineering Project – 3]

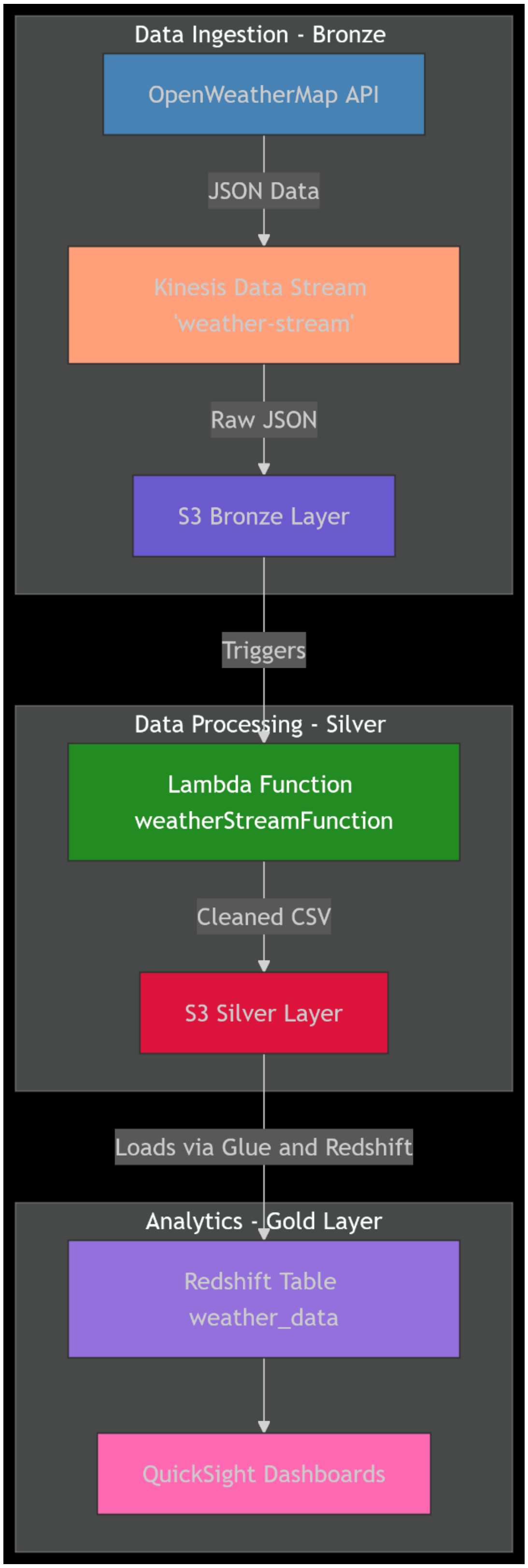
Objective

The objective of this project is to design and implement a **real-time weather data pipeline** using AWS serverless technologies. The pipeline ingests, processes, stores, and visualizes weather data to support:

- **Real-time monitoring** of weather conditions.
- **Historical trend analysis** for forecasting.
- **Automated data processing** with minimal latency.
- **Data-driven decision-making** via dashboards.

Key outcomes include:
- 90% reduction in manual data processing time.
- Scalable, cost-efficient architecture using AWS Kinesis, Lambda, Redshift, and QuickSight.

System Architecture



Prerequisites

- **AWS Account** with permissions for Kinesis, Lambda, Redshift, S3, and IAM.
- **Python 3.x** for Lambda functions and data simulation.
- **OpenWeatherMap API Key** (for simulated data).
- **Terraform/IaC** (optional for infrastructure automation).

Component Breakdown

A. Data Ingestion (Kinesis)

- **Kinesis Stream:** weather-stream (on-demand capacity mode).
- **Data Producer:** Python script (weather_stream-project-3.py) posts weather data every 60 seconds.

B. Data Processing (Lambda)

- **Lambda Function:** weatherStreamFunction triggers on Kinesis events.
  - Converts Kelvin to Celsius.
  - Validates and flattens JSON into CSV.
  - Stores raw (Bronze) and cleaned (Silver) data in S3.

C. Data Warehouse (Redshift)

- **Table Creation**
- **Workgroup:** Serverless Redshift configured with VPC security groups.

D. Analytics (QuickSight)

  - QuickSight Arena

Design Decisions

Decision	**Rationale**
Kinesis (On-Demand)	Handles unpredictable data spikes without manual shard management.
Lambda for ETL	Serverless scaling, cost-efficient for sporadic data batches.
S3 for Bronze/Silver	Cost-effective storage with lifecycle policies for raw/processed data.
Redshift Serverless	Auto-scales compute for analytical queries; no cluster management.
QuickSight	Integrated with AWS, supports real-time dashboards.

Data Flow

1. **Ingestion:**

    - Python script - Kinesis (weather-stream) - S3 Bronze (raw JSON).

2. **Processing:**
    - Lambda reads Kinesis - Cleans data - S3 Silver (CSV).
3. **Warehousing:**
    - Lambda/Glue loads CSV - Redshift (weather_data).
4. **Visualization:**
    - QuickSight queries Redshift - Dashboards.

Security & Compliance

- **Encryption:**
    - Kinesis/KMS (data in transit/at rest).
    - S3 buckets with SSE-S3.
- **IAM Roles:** Least-privilege access for Lambda (e.g., LambdaRoleProject3).
- **VPC Isolation:** Redshift deployed in a private subnet with security groups.

Monitoring & Quality

- **CloudWatch:** Logs for Lambda/Kinesis errors.
- **Data Validation:**
    - Lambda checks for empty/invalid JSON.
    - Unit conversion (Kelvin - Celsius) validated.
- **Redshift Query Monitoring:** Track performance via Redshift console.