NEXUS DATALENS
Version 2.0

An End-to-End Automated Data Analytics Pipeline on AWS

Automated Serverless Data Analytics Pipeline on AWS

Prepared by
Ruksana Shaikh

Date: September 9, 2025

Abstract

Nexus DataLens 2.0 *is a cloud-native, end-to-end automated data analytics pipeline built on Amazon Web Services (AWS). It addresses the challenge of transforming raw CSV data into meaningful insights without manual processing or technical expertise. Users upload CSV files through a web interface, after which the system autonomously performs ingestion into Amazon S3, cleaning with AWS Glue, schema cataloging through AWS Glue Crawlers, and visualization using Amazon QuickSight or Power BI. The result is a scalable, secure, and cost-effective solution that accelerates data-to-decision workflows with minimal effort, showcasing the power of serverless, event-driven architectures in modern data engineering. This solution is applicable to SMEs, research institutions, and enterprises looking for automated analytics*

Table of Contents

1. Introduction

1.1 Background

In today's data-driven environment, organizations collect large volumes of raw data from various sources. However, transforming this data into actionable insights typically requires technical expertise, manual cleaning, and lengthy processes. There is a growing need for scalable, automated solutions that can streamline data ingestion, processing, and visualization particularly for non-technical users. Nexus DataLens addresses this gap by providing a fully automated pipeline built on reliable AWS services.

1.2 Motivation

This project is motivated by the need to simplify and democratize data analytics. Manual workflows often introduce delays, errors, and scalability issues. With increasing data volumes, it becomes essential to implement a system that automates every step—from file ingestion to dashboard generation—thereby saving time, reducing effort, and enhancing accessibility for all users, regardless of technical background.

1.3 Scope of the Project

This project focuses on building an end-to-end data analytics solution capable of:

- Ingest CSV files via a web-based frontend
- Process and clean data automatically using AWS Glue and DataBrew
- Create and update Athena tables for SQL querying
- Visualize data dynamically in Amazon QuickSight and optionally Power BI
- Enables seamless, near real-time insights without human intervention

The scope is limited to CSV-based structured data and primarily uses AWS services under the free tier wherever possible.

1.4 Technologies Used

The Nexus DataLens project integrates a diverse range of AWS services and technologies to deliver a seamless, automated data analytics workflow. The technologies used are grouped below into **Frontend**, **Backend**, and **Support Services** for clarity:

Frontend Technologies

- **HTML, CSS, JavaScript** – Developed a clean, responsive user interface for CSV upload
- **Amazon S3 (Static Website Hosting)** – Hosts the frontend as a static web application
- **AWS Cognito** – Provides secure user authentication

- **Upload Path** – Files are uploaded directly to:
  s3://nexus-web-frontend1/data/filename.csv

Backend & Data Processing Technologies

- **Amazon S3** – Stores both raw uploaded files and cleaned outputs
- **AWS Lambda** – Automates Glue Crawlers and Glue Jobs upon file upload
- **AWS Glue Crawler** – Detects schema from new data files and updates Glue Catalog
- **AWS Glue (Spark-based Jobs)** – Cleans and transforms CSV data efficiently
- **AWS Glue DataBrew** – Provides no-code UI-based transformation for advanced cleaning
- **Amazon Athena** – Performs SQL queries on cleaned datasets stored in S3
- **Amazon QuickSight** – Creates dynamic, interactive dashboards from Athena datasets
- **Power BI** *(optional)* – Enterprise-level reporting via Athena ODBC integration
- **Python** – Used in AWS Glue Jobs and Lambda scripts for logic implementation

Security, Monitoring & Integration Services

- **AWS IAM** – Role-based access control for services and users
- **AWS KMS** – Key management for data encryption at rest
- **AWS CloudWatch** – Real-time logging and monitoring of Lambda and Glue operations
- **AWS API Gateway** – Exposes RESTful endpoints if integrating external tools
- **Amazon CloudFront** – Content delivery network (CDN) for global access to the frontend
- **AWS Cognito** – Adds user pool-based authentication to restrict uploads

2. Problem Statement

In many organizations, data analysts and business users often spend significant time manually processing and cleaning uploaded data for reporting purposes. This traditional approach is not only time-consuming but also prone to human error, lacks scalability, and delays decision-making. There is a growing need for an automated mechanism that can reliably process raw user-uploaded CSV files and generate meaningful insights with minimal technical effort.

3. Solution Overview

Nexus DataLens addresses this challenge by implementing a fully automated, event-driven data analytics pipeline built on AWS. The architecture eliminates manual intervention and delivers real-time insights using the following key services:

- **Amazon S3** – for secure and scalable file storage
- **AWS Lambda** – for automating workflow execution upon file upload
- **AWS Glue (Crawler + Job)** – to infer schema and perform data cleaning and transformation
- **Amazon Athena** – to query structured, cleaned datasets using standard SQL
- **Amazon QuickSight** *(and optionally Microsoft Power BI)* – to build rich, interactive dashboards for end users

The entire pipeline is cloud-native, scalable, and designed for seamless integration, ensuring data moves from ingestion to visualization without any manual processing.

3.1. Objectives

The primary objectives of the Nexus DataLens project are as follows:

- To build a robust and scalable platform using AWS services for automated CSV data ingestion
- To clean and transform datasets using AWS Glue with no manual effort
- To enable seamless data visualization through **Amazon QuickSight** and **Power BI**
- To create a frontend user interface that allows non-technical users to upload datasets and receive insights effortlessly

Note: This project supports visualization through both Amazon QuickSight and Microsoft Power BI. Both platforms were successfully integrated and tested during implementation.

4. System Architecture

The architecture of **Nexus DataLens** is designed for **end-to-end automation**, **real-time data processing**, and **scalable analytics** — all built on a **serverless AWS stack**.

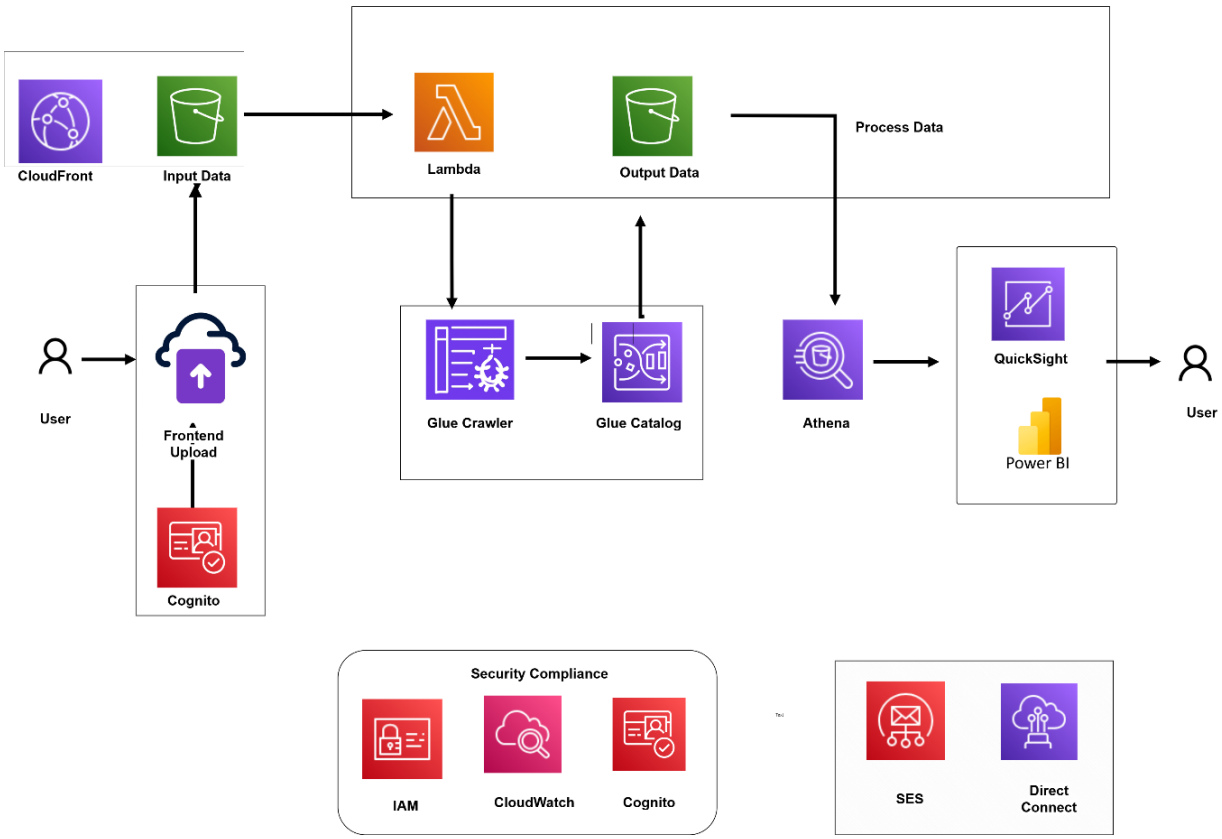


Figure 1: Nexus DataLens – End-to-End Serverless Architecture

Data Flow Explanation

This architecture enables seamless data ingestion, processing, and visualization using AWS services. Here's how the components interact:

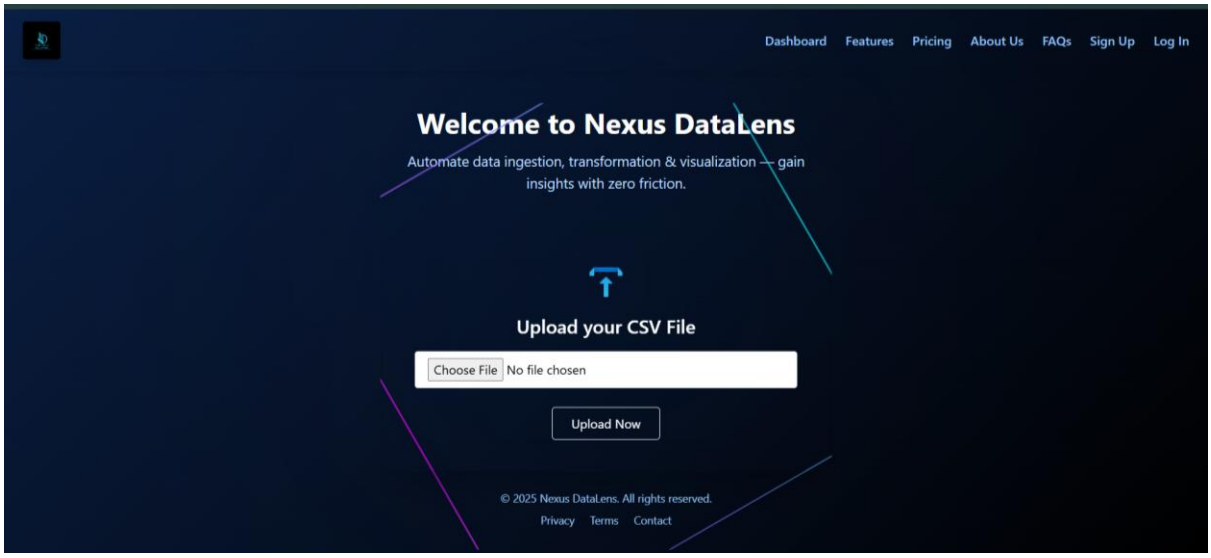
1. **Frontend Upload & Authentication**
   - Users upload CSV files through a web-based frontend.
   - **Amazon Cognito** secures access with optional user authentication.
   - Files are stored in the S3 input folder and served via **CloudFront**.
2. **AWS Lambda Orchestration**
   - The S3 upload triggers **AWS Lambda**, which initiates:
     - A **Glue Crawler** to scan and update the **Glue Catalog**
     - A **Glue ETL Job** to clean and format the data

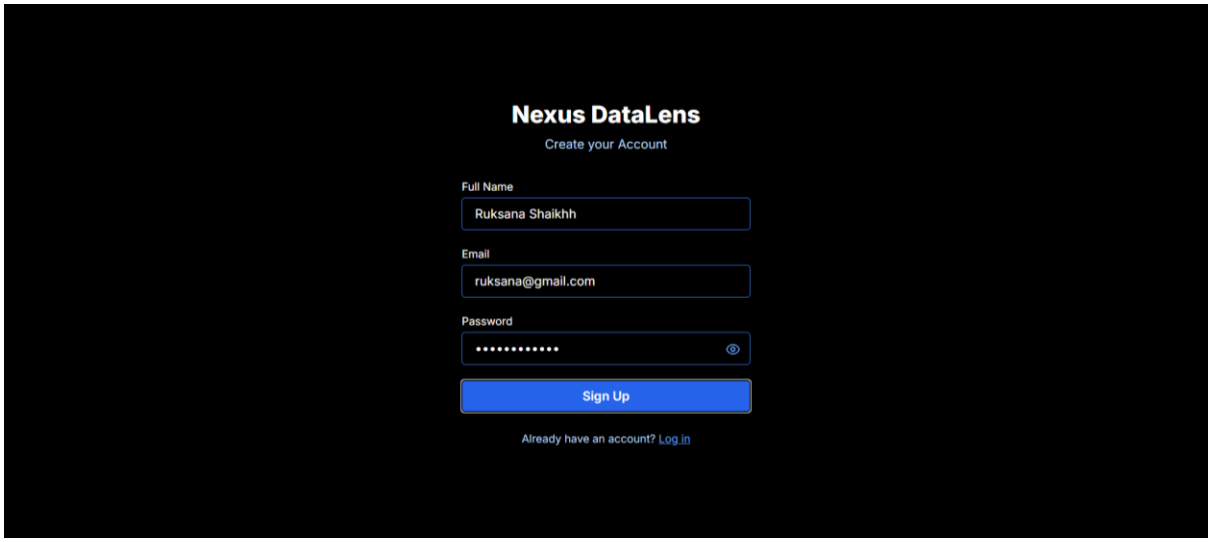
   - Cleaned data is written to the cleaned-output/ folder in S3.
3. **Athena & Data Querying**
   - Cleaned data is queried using **Amazon Athena**.
   - Athena integrates with both **QuickSight** and **Power BI** for dashboarding.
4. **Visualization & Reporting**
   - Users can visualize the data using **Amazon QuickSight** or **Microsoft Power BI** dashboards.
5. **Security & Compliance**
   - **IAM** ensures access control.
   - **CloudWatch** handles monitoring and alerts.
   - **Cognito** adds secure authentication for both the frontend and backend services.
6. **Optional Enterprise Features**
   - **Amazon SES** can send automated email reports/alerts.
   - **AWS Direct Connect** can be used in enterprise setups for faster, private network access.

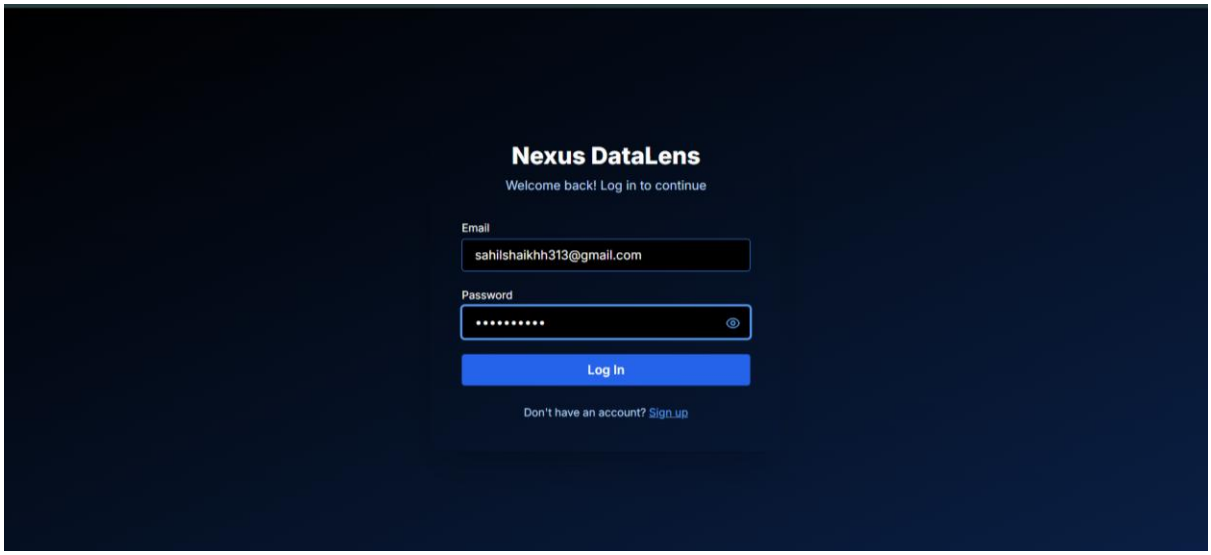
5. Frontend Application

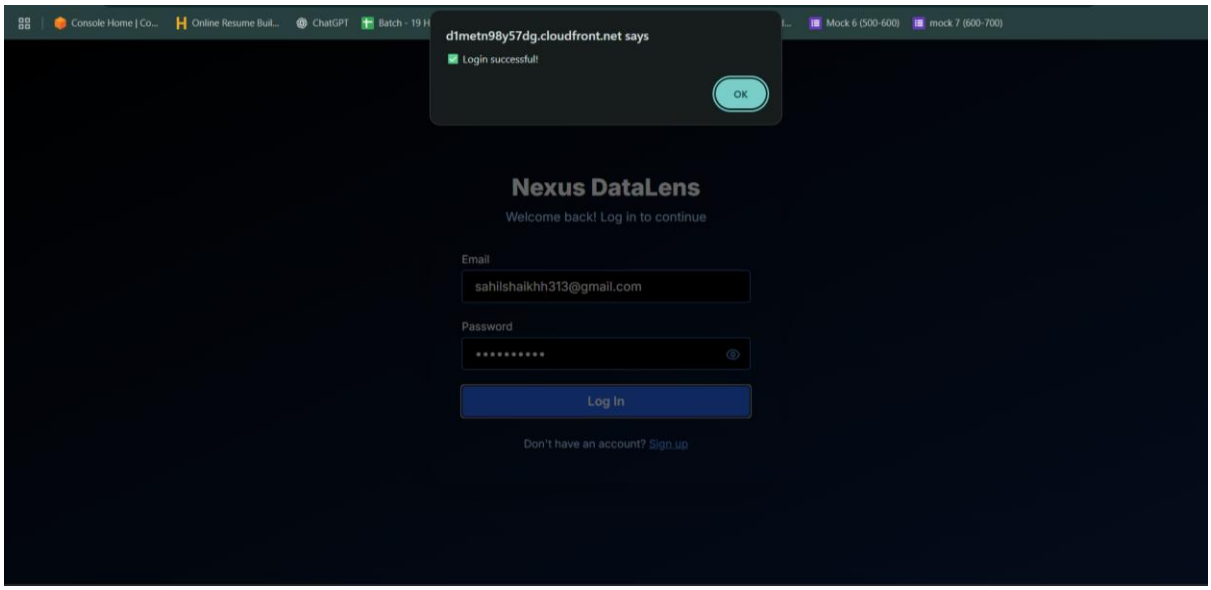
5.1 Upload Interface

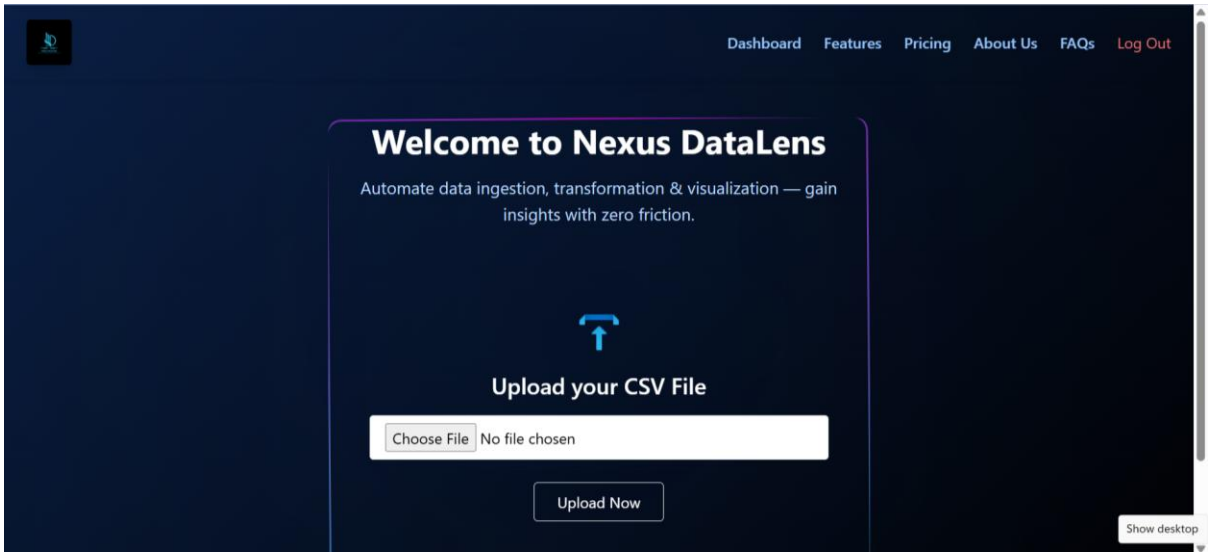
This section covers the CSV file upload mechanism built using a simple HTML/CSS/JavaScript interface hosted on Amazon S3.

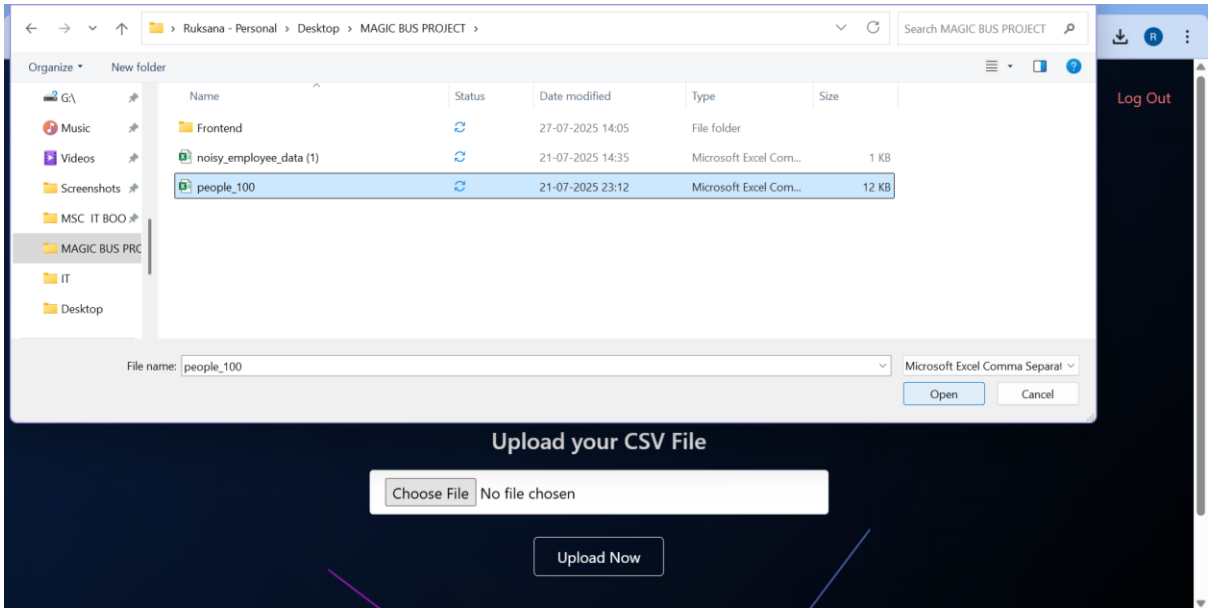


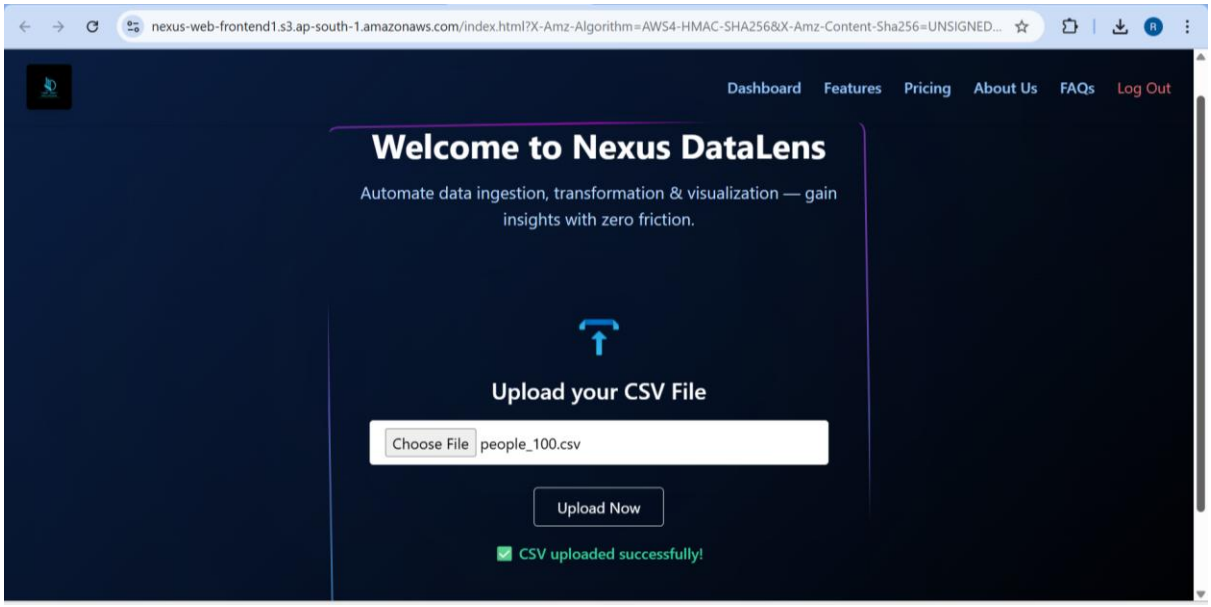


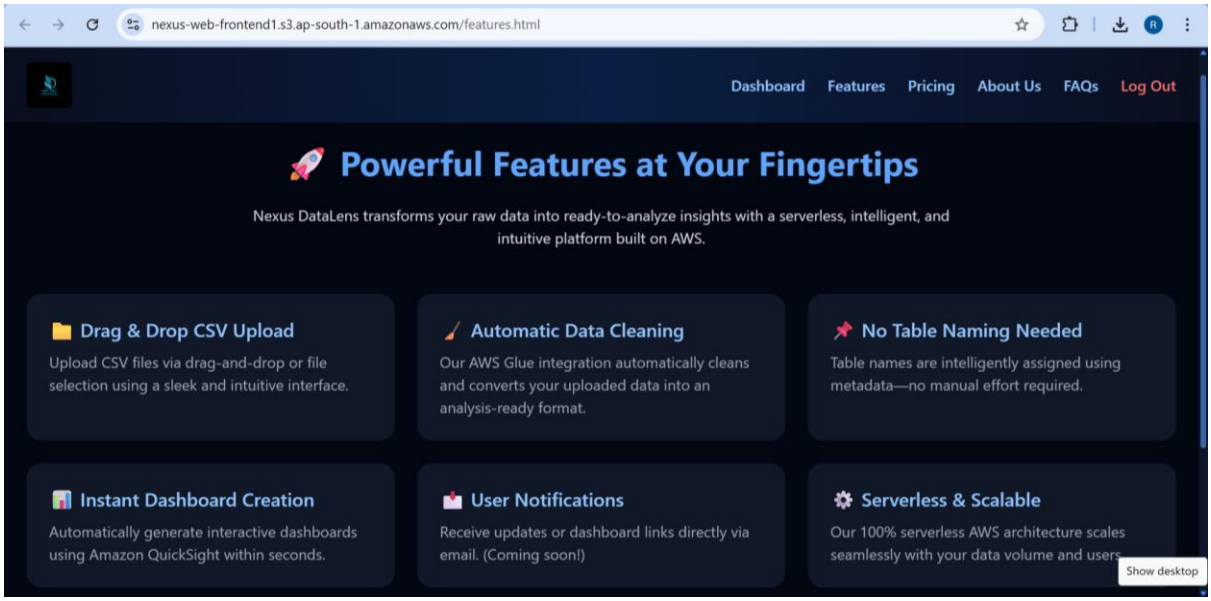


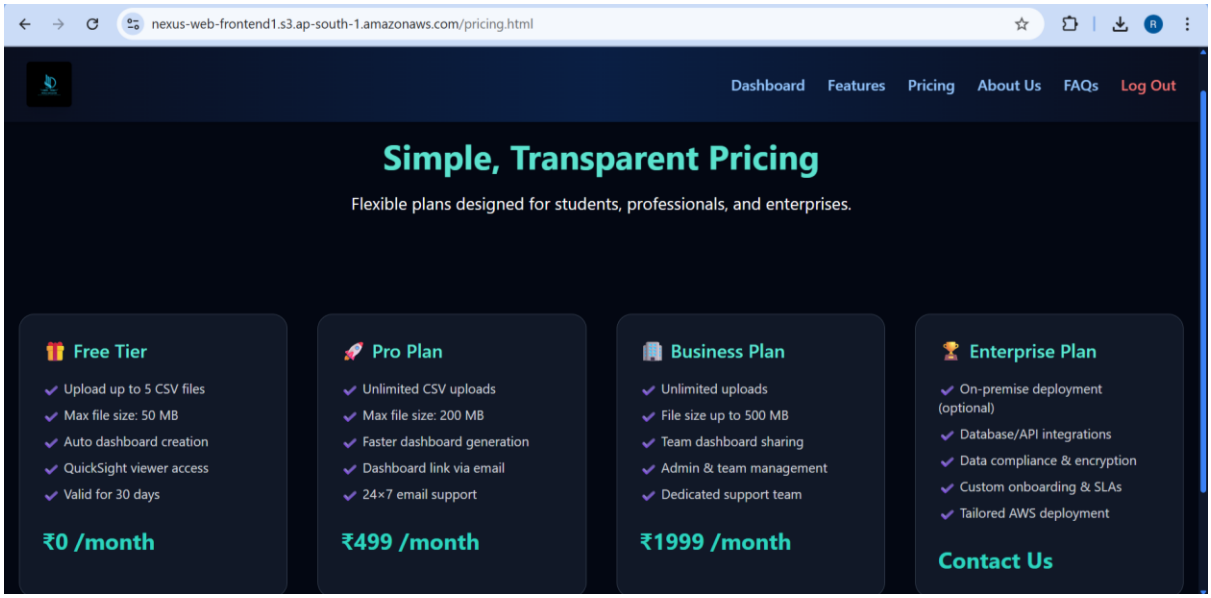


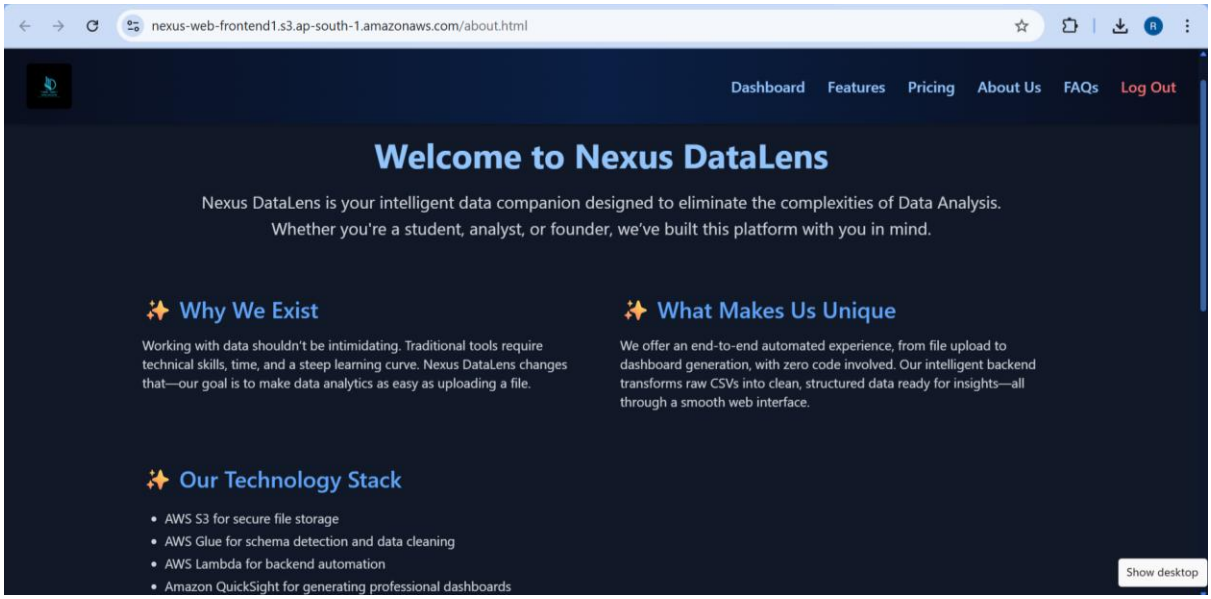


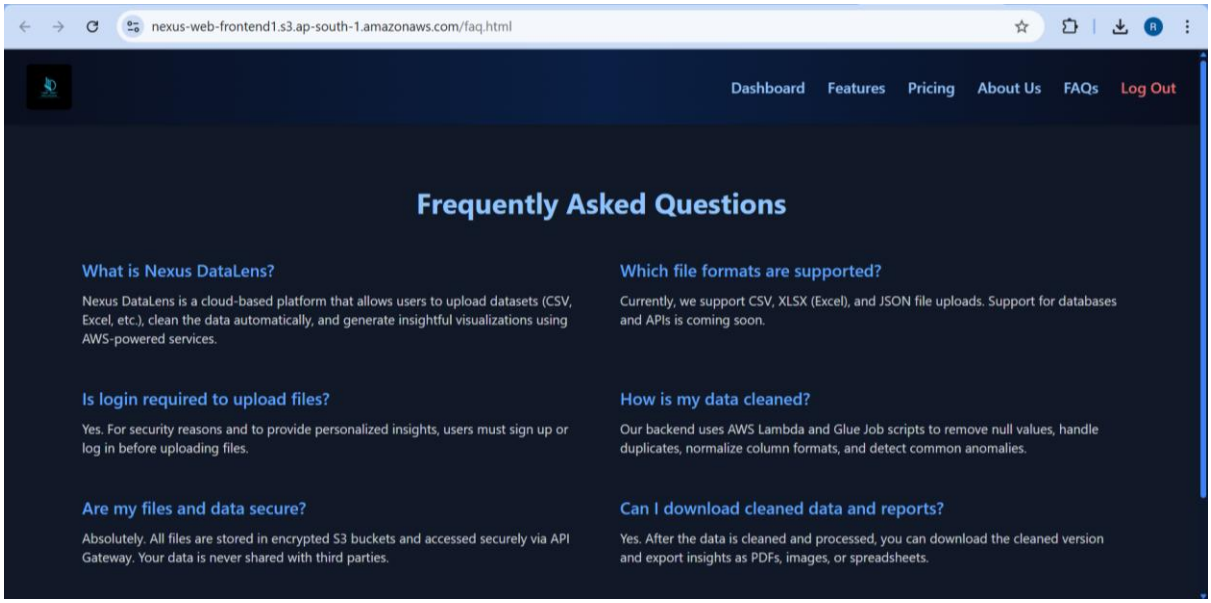


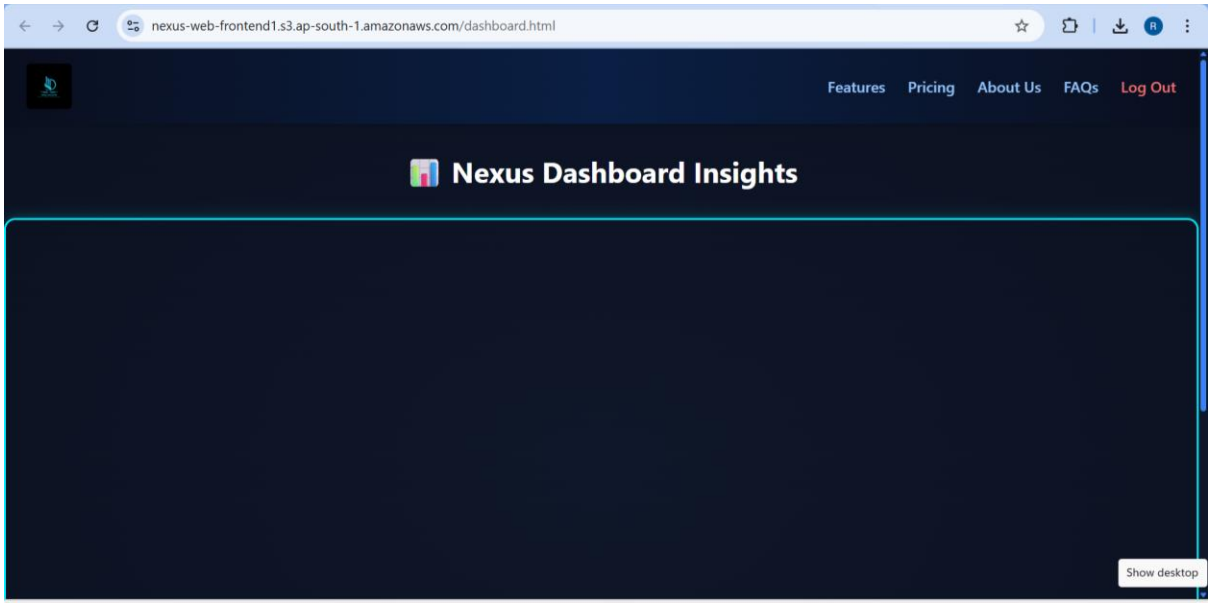


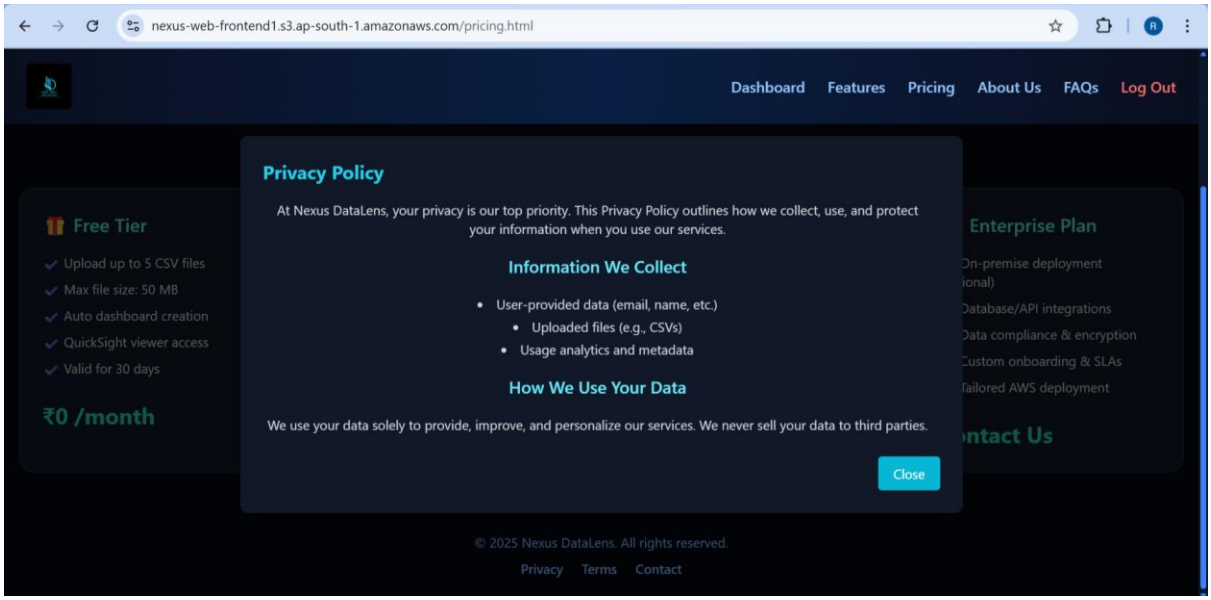


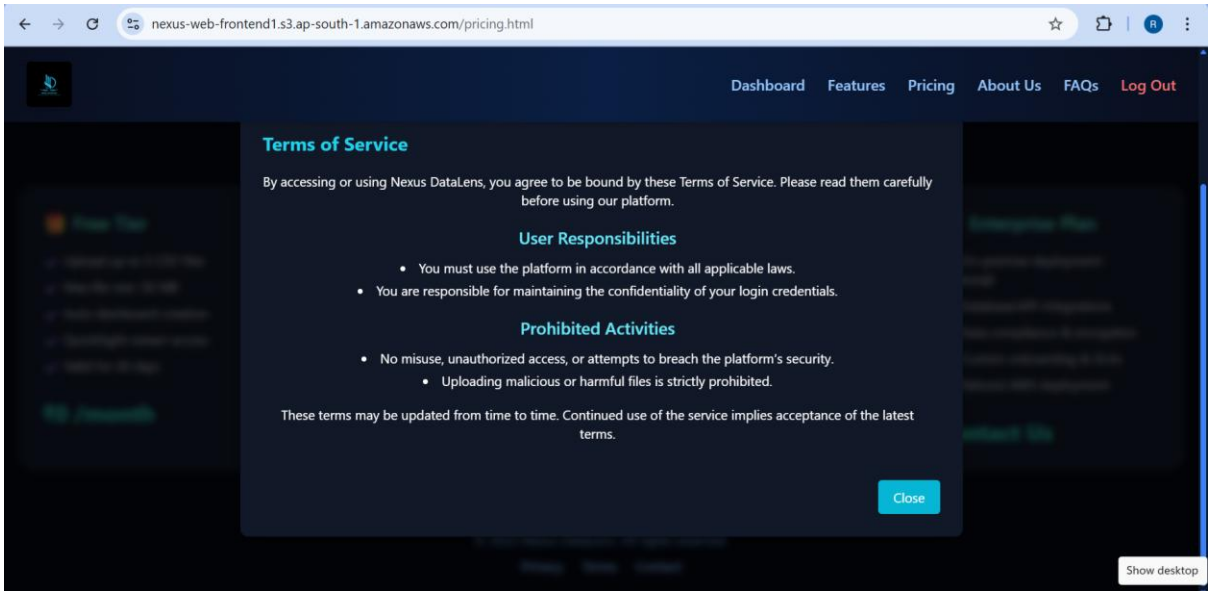


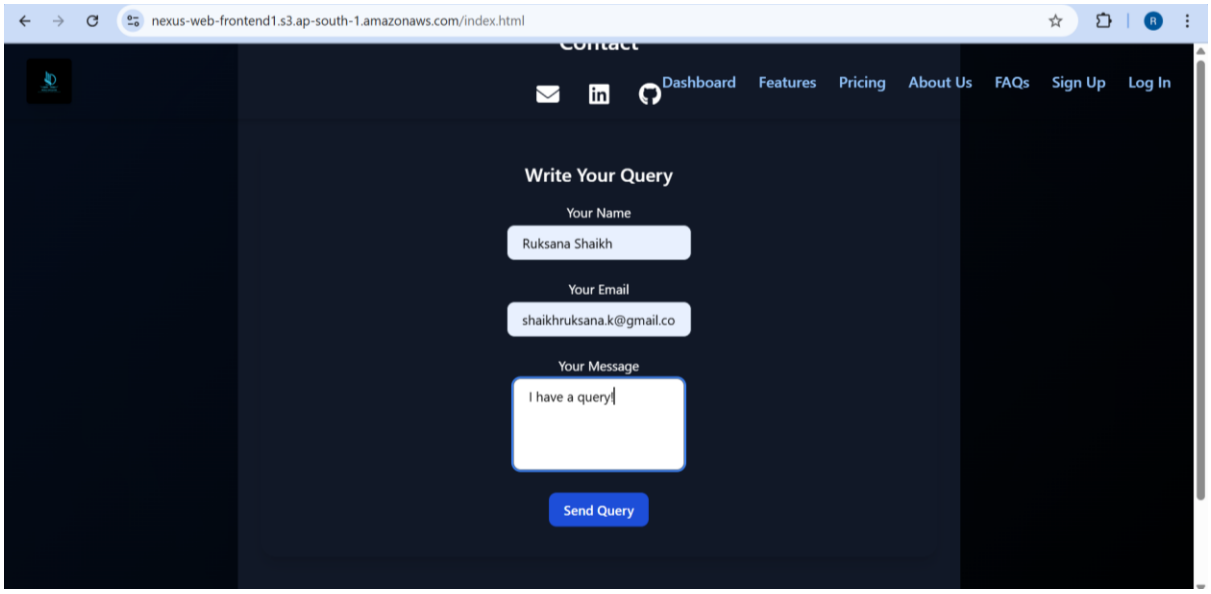






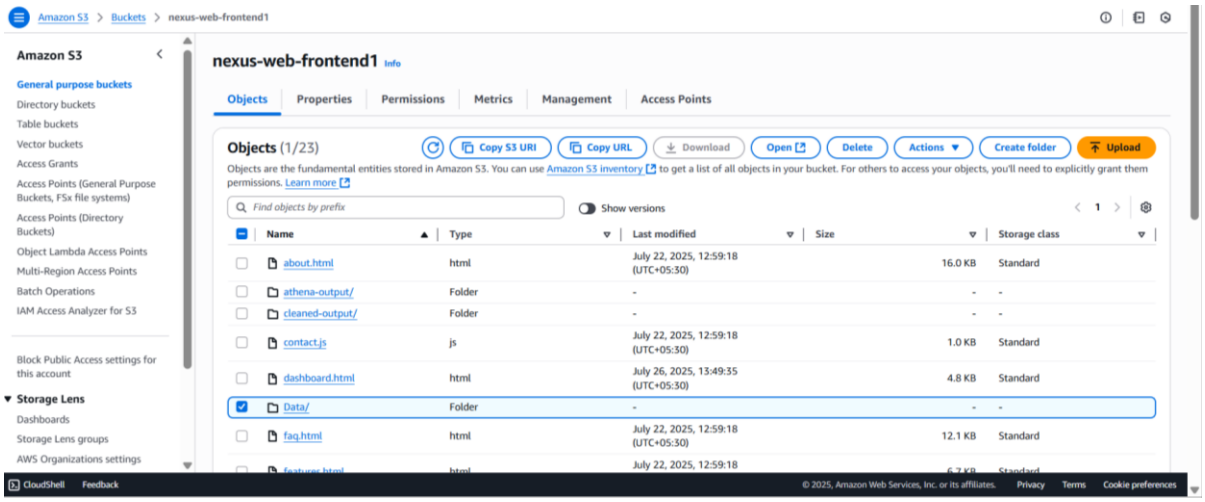


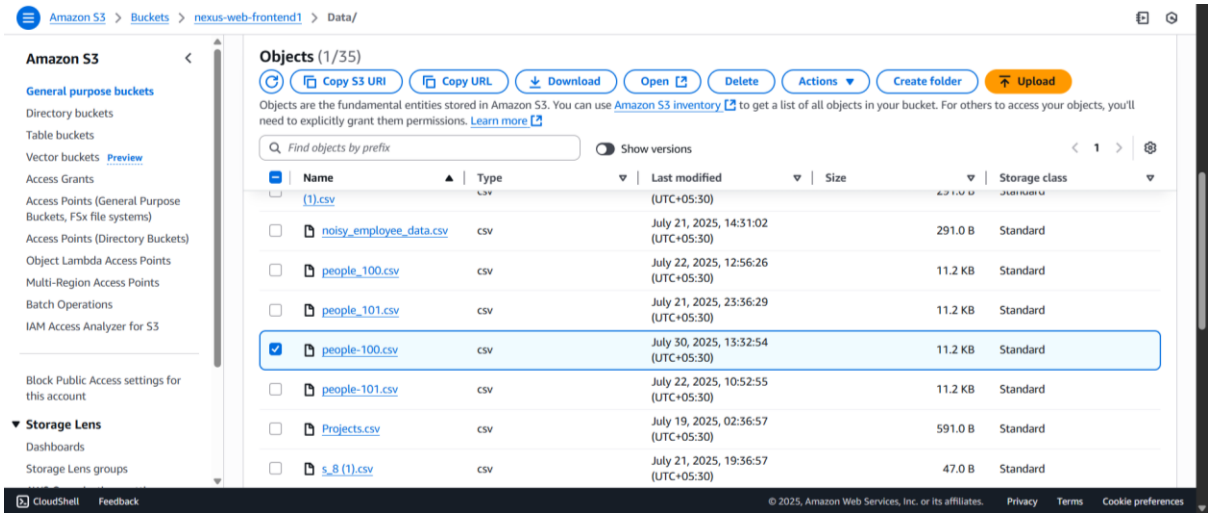




5.2 S3 Bucket Structure

Overview of the S3 bucket layout including folders for raw and cleaned data.





5.3 Frontend File Upload and Hosting

The frontend is a **React-based web application**, hosted on **Amazon S3 (Static Website Hosting)** where users interact to upload .csv files.

- Uploaded files are securely stored in:
- s3://nexus-web-frontend1/data/

- **Amazon CloudFront** is used as a CDN to deliver the frontend globally with low latency.
- **AWS Cognito** enables **secure user authentication**, ensuring that only authorized users can upload CSV files.

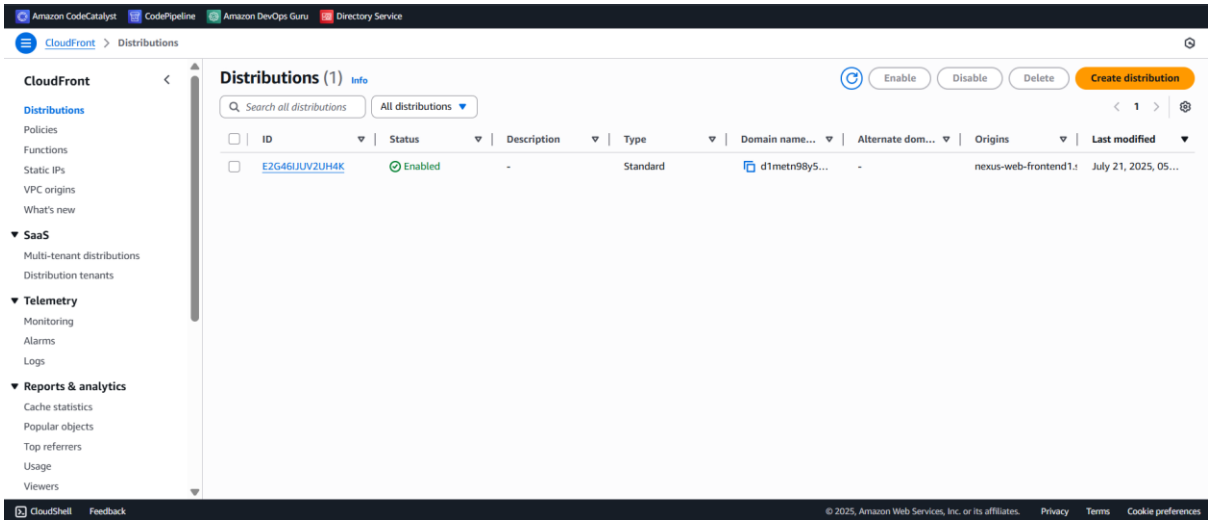
🛠 **Technologies Used**:

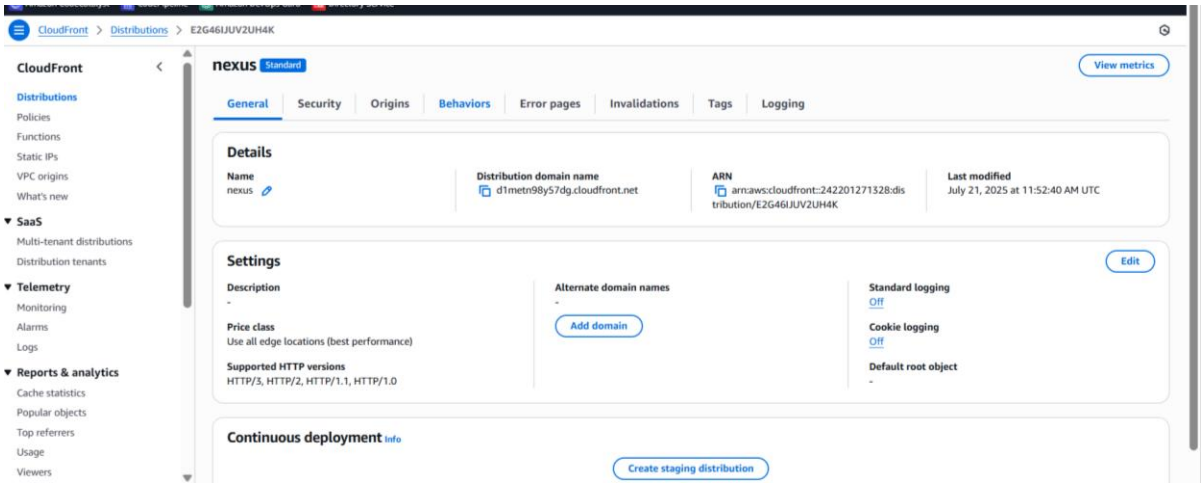
- Amazon S3 (Website Hosting)

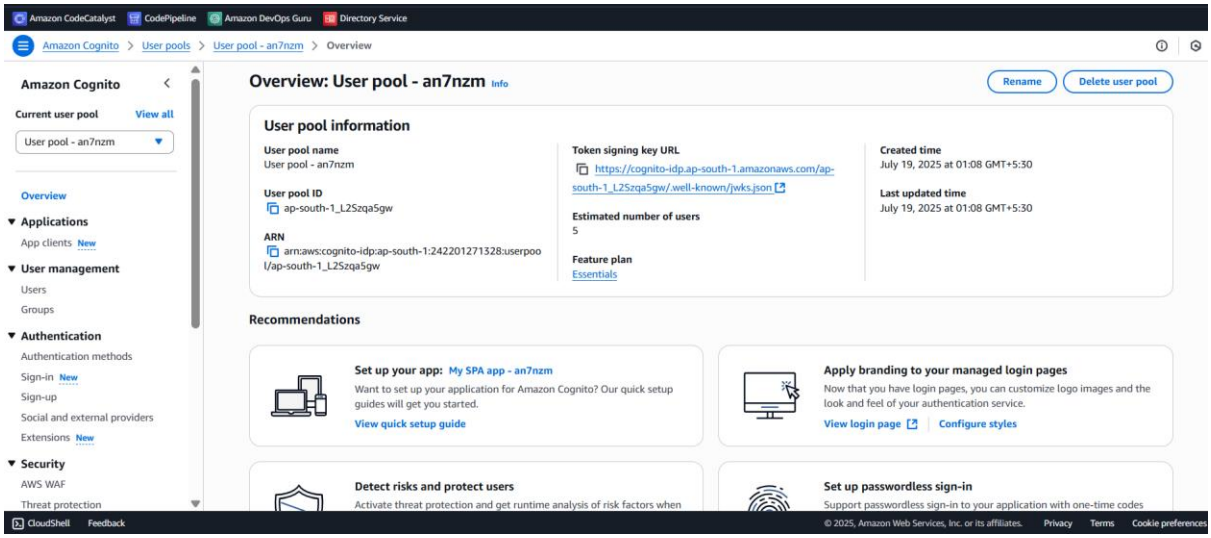
- Amazon CloudFront (CDN)
- AWS Cognito (Authentication)

📸 **Suggested Screenshots**:

- Frontend UI with file upload section
- S3 bucket hosting the static site (console)
- Cognito user pool dashboard







5.4 Secure File Storage in Amazon S3

- Upon upload, CSV files are stored in the raw data folder:

bash

Copy code

s3://nexus-web-frontend1/data/

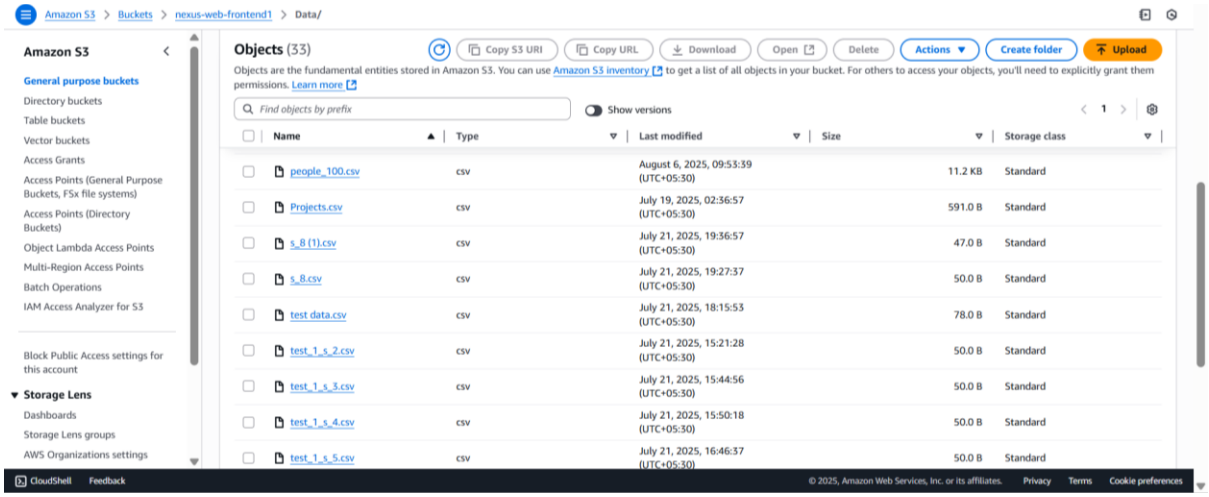
- **AWS KMS (Key Management Service)** ensures encryption at rest.
- Fine-grained access is controlled using **AWS IAM roles and policies**.

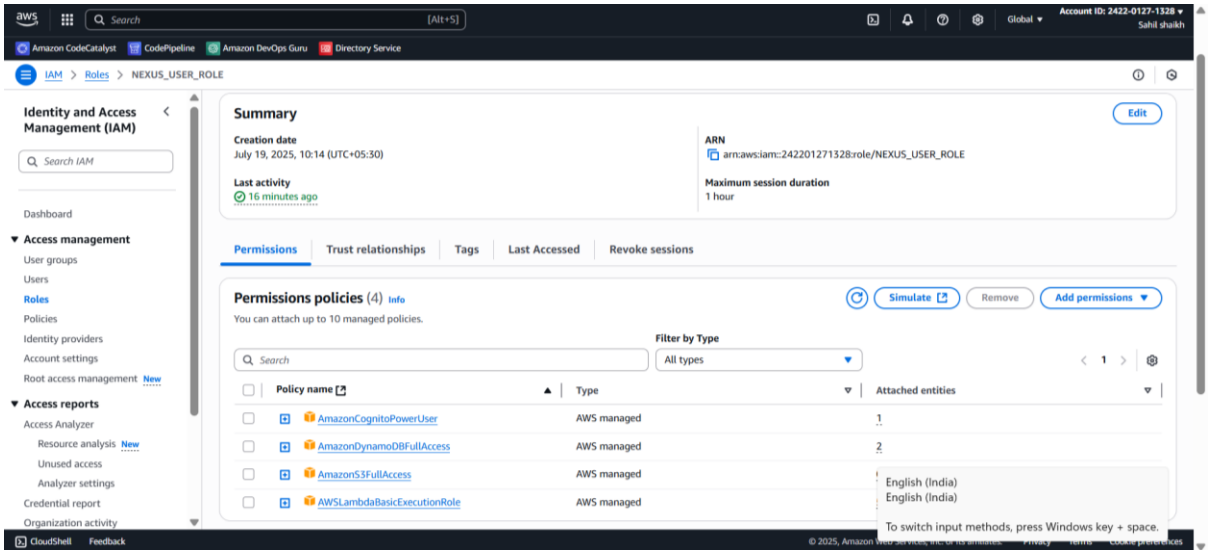
🛠 **Technologies Used**:

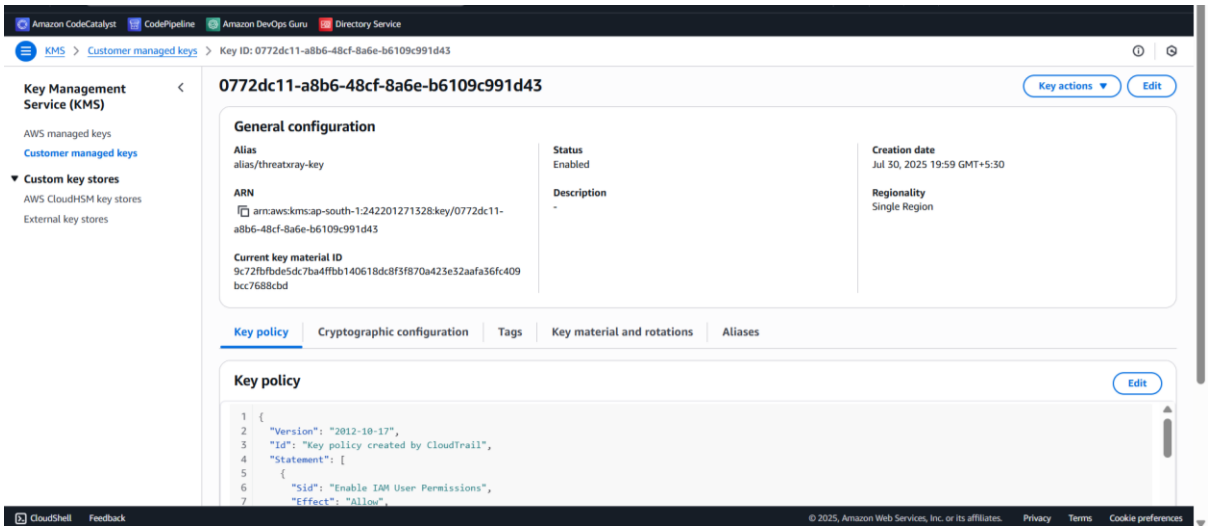
- Amazon S3 (Data Storage)
- AWS IAM (Access Control)
- AWS KMS (Encryption)

📸 **Suggested Screenshots**:

- S3 bucket structure with uploaded CSV in data/
- IAM role used for S3 access
- KMS key policy for encryption







Project Workflow – Step-by-Step Breakdown

The **Nexus DataLens** project leverages a robust serverless architecture to automate the entire data lifecycle — from file upload to advanced business visualizations. Below is a comprehensive breakdown of how each AWS service integrates into the workflow:

6. Automation Using AWS Lambda

6.1 Trigger Mechanism

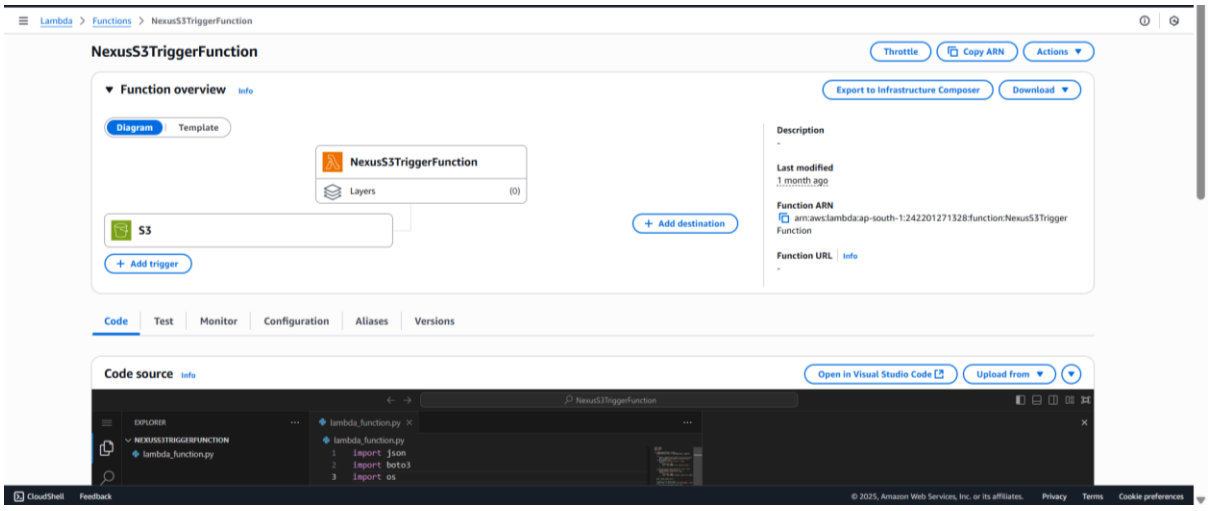
- An S3 PUT event notification automatically triggers the LambdaTriggerQuicksight Lambda function. It performs the following operations:
    - Invokes **Glue Crawler** to scan the raw file
    - Waits for crawler completion via polling logic
    - Runs the **Glue Job** for cleaning

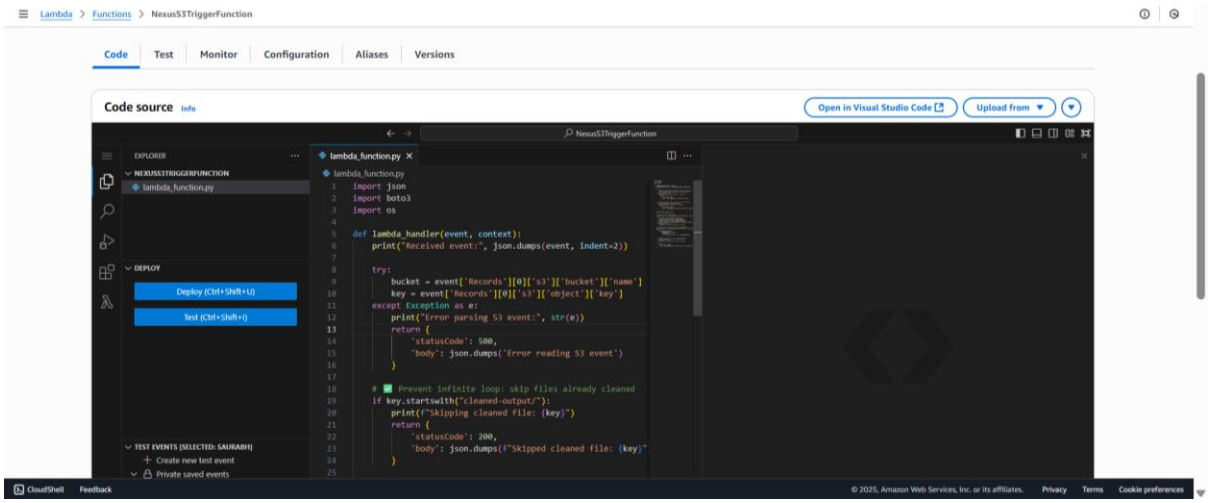
🛠 **Technologies Used**:

- AWS Lambda
- Amazon S3 Event Notifications
- AWS CloudWatch (Monitoring)

📸 **Suggested Screenshots**:

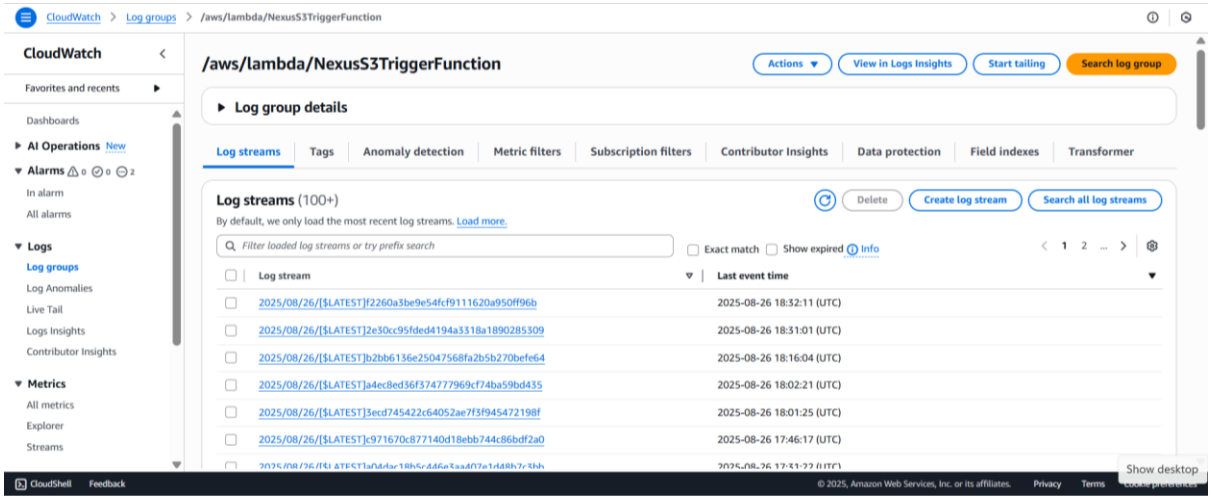
- Lambda trigger config (S3 event → Lambda)
- Lambda code snippet (Python)
- CloudWatch logs showing process

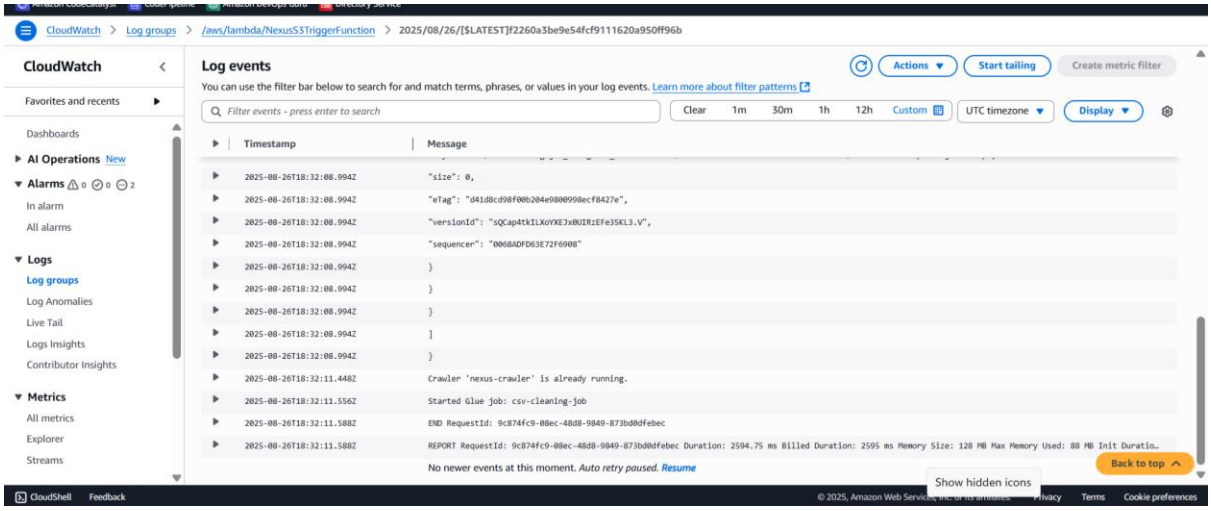


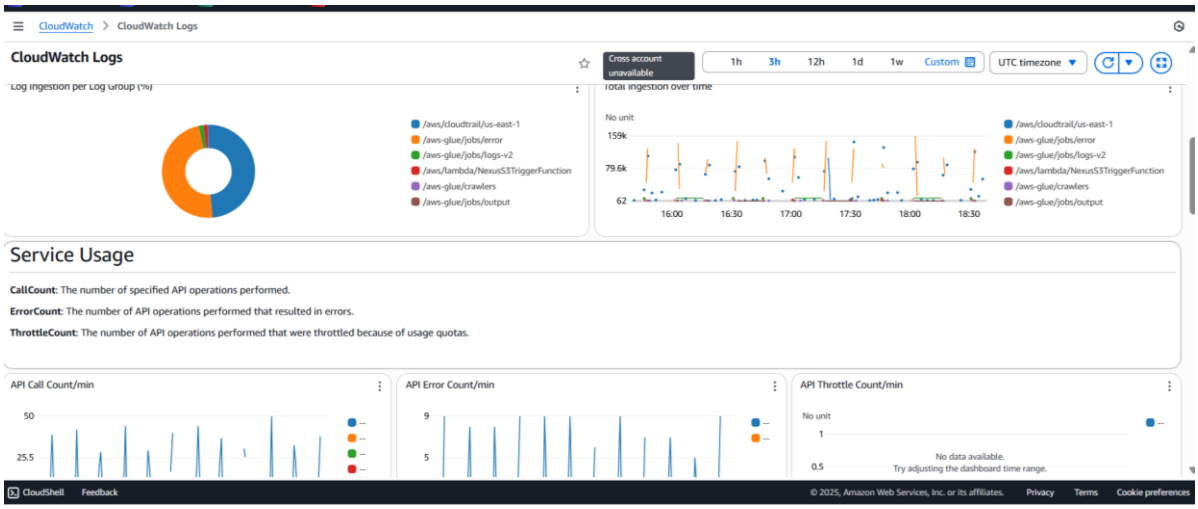


6.2 Lambda Trigger Function

Lambda Trigger Function 2 → Automatically creates Athena-based datasets and QuickSight dashboards after Glue crawler finishes.
It ensures user-specific datasets are generated and visualized in QuickSight dynamically.







7. Data Processing Pipeline

7.1 Schema Detection with AWS Glue Crawler

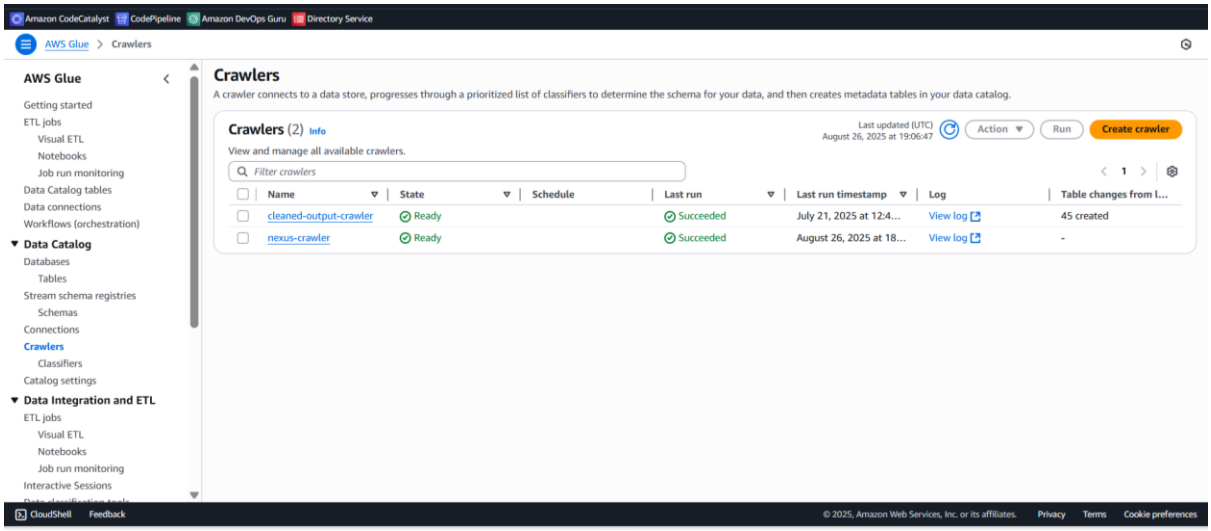
- The Glue Crawler scans the uploaded file in the data/ folder, detects the schema, and updates the **AWS Glue Data Catalog.**
- The resulting table is used by **Athena** for SQL querying**.**

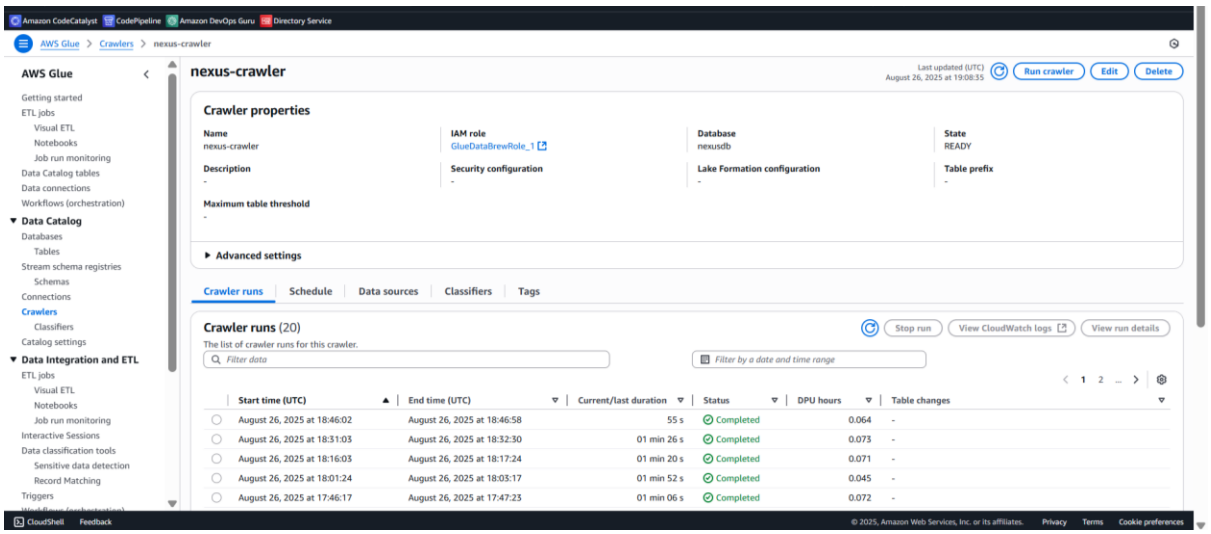
🛠 **Technologies Used:**

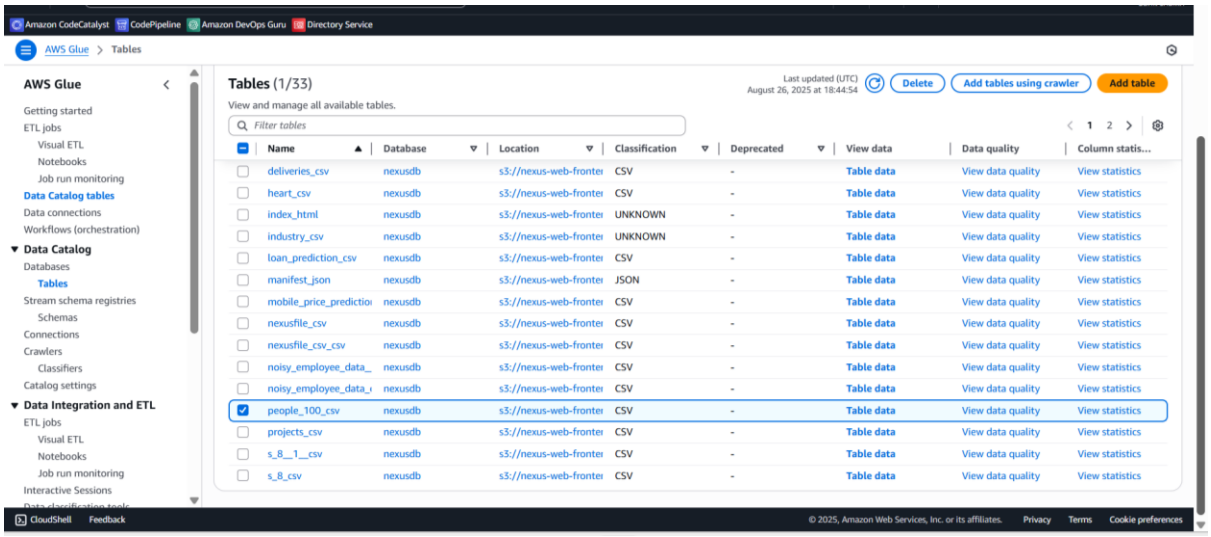
- **AWS Glue Crawler**
- **AWS Glue Catalog**

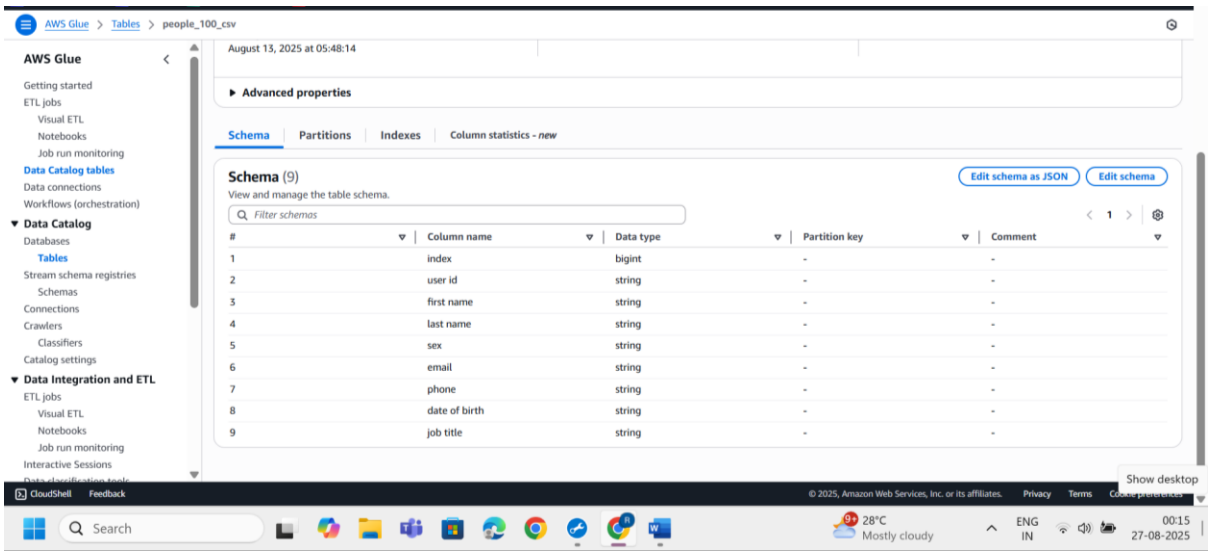
📸 *Screenshot Suggestions:*

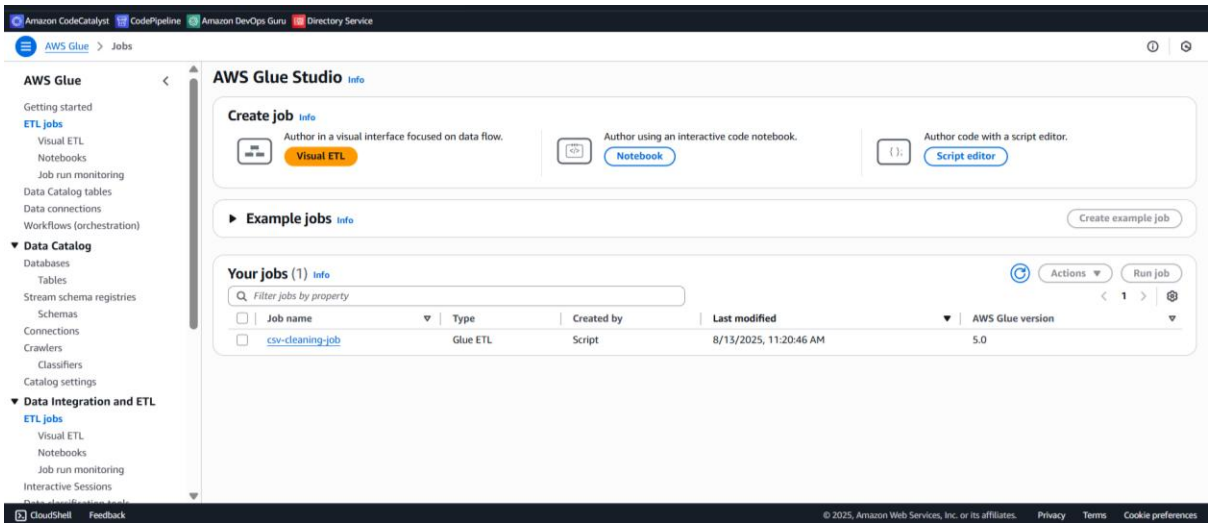
- Glue Crawler configuration page
- Table in Glue Catalog pointing to data/ S3 path











7.2 Glue Job – Data Cleaning

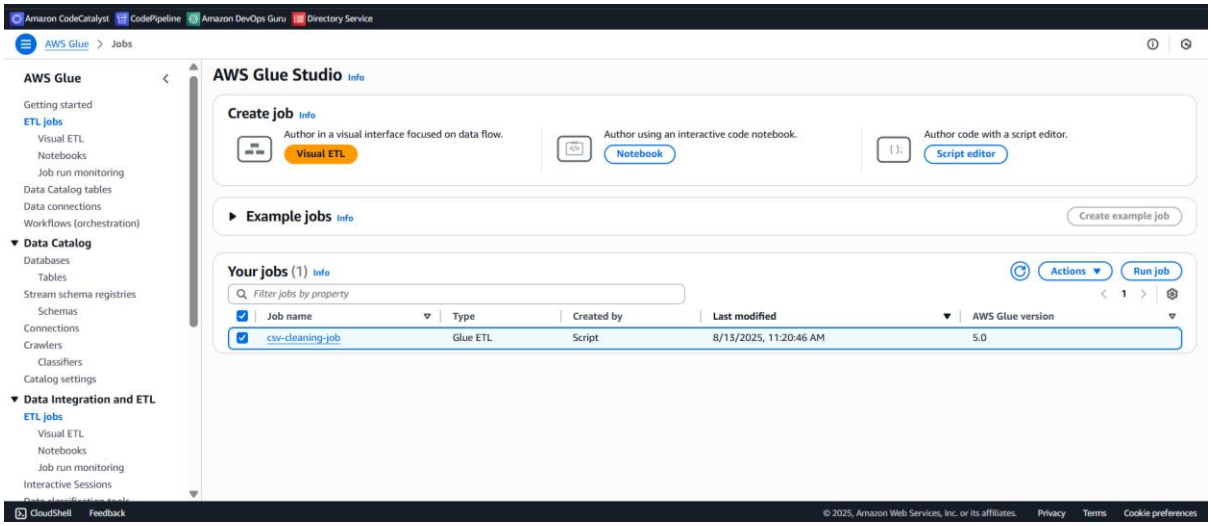
The Spark-based job csv-cleaning-job processes the raw data:

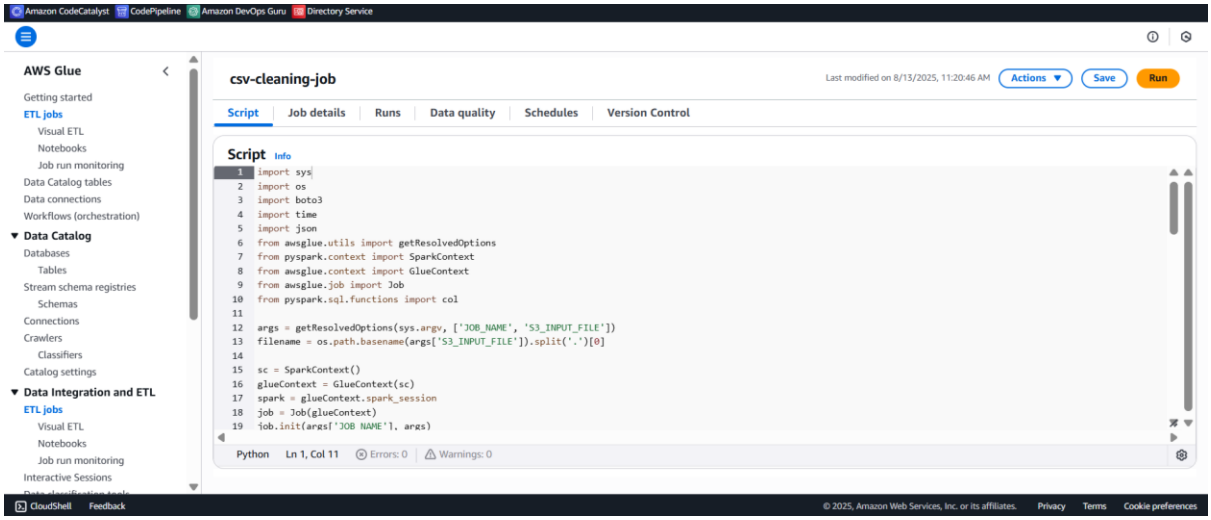
- Drops rows and columns that are entirely null
- Filters out invalid values
- Standardizes column headers
- Repartitions to a single flat file

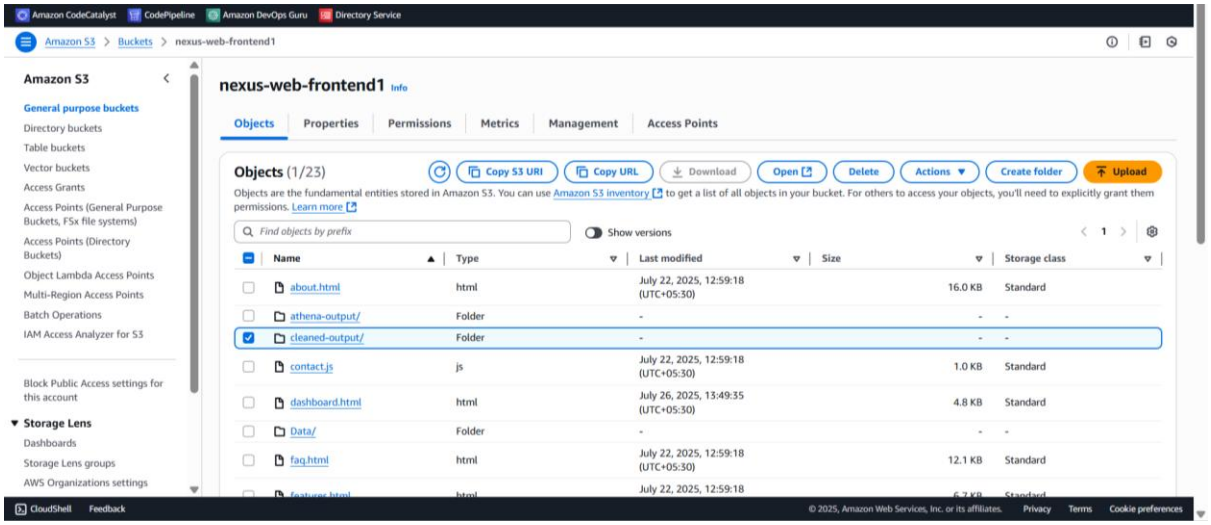
Cleaned output is written to:
s3://nexus-web-frontend1/cleaned-output/<filename>.csv

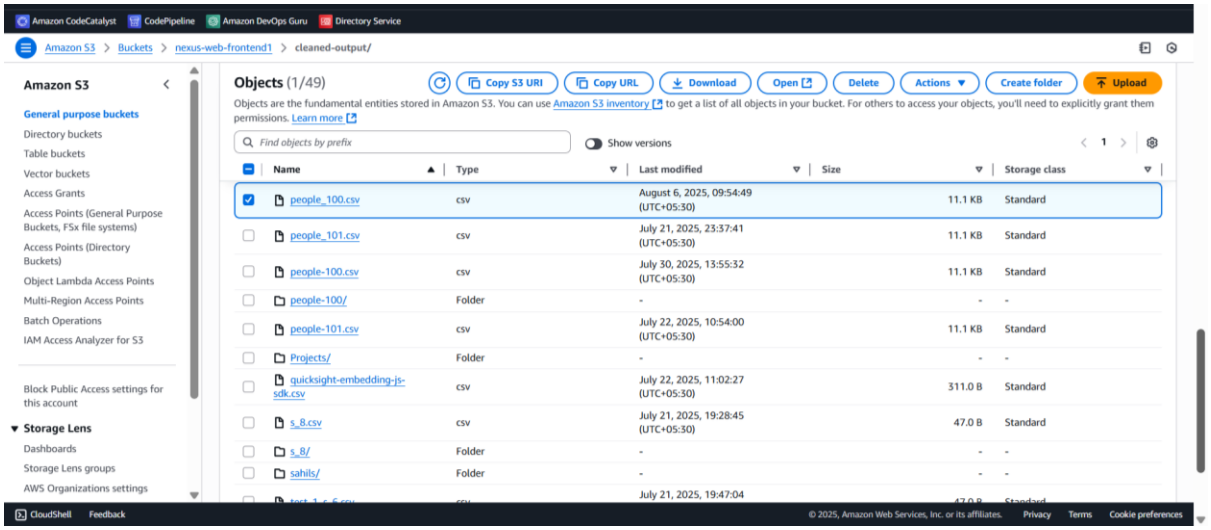
Screenshot Suggestions:

- Glue job script code
- Sample cleaned output file in S3
- Before vs After comparison of the CSV (raw vs cleaned)









7.3 Advanced Cleaning with Glue DataBrew *(Optional)*

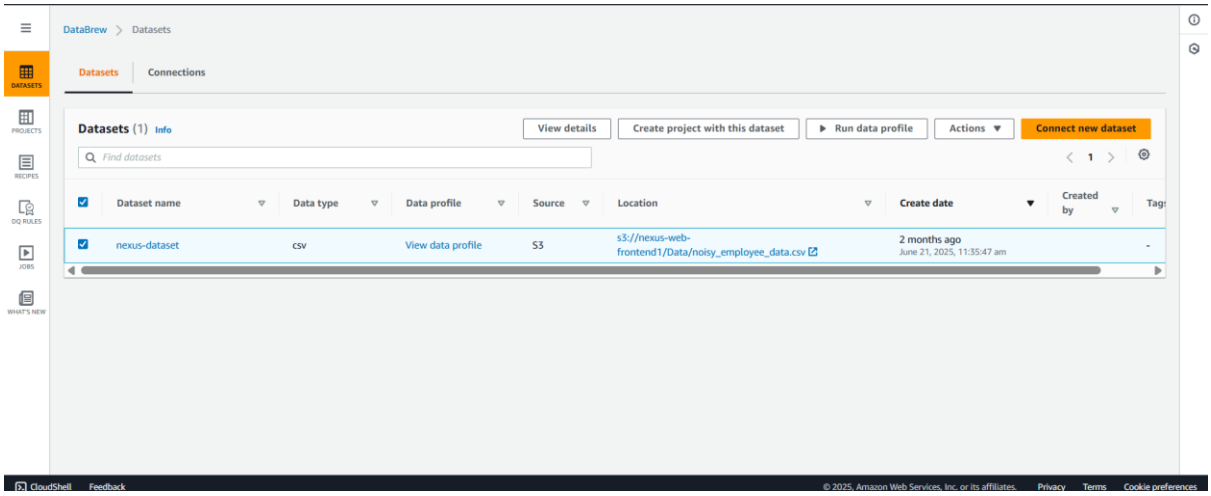
- For more complex or no-code transformations, **AWS Glue DataBrew** jobs can be triggered.
- It enables:
  - Data profiling
  - Pattern matching
  - Custom transformations

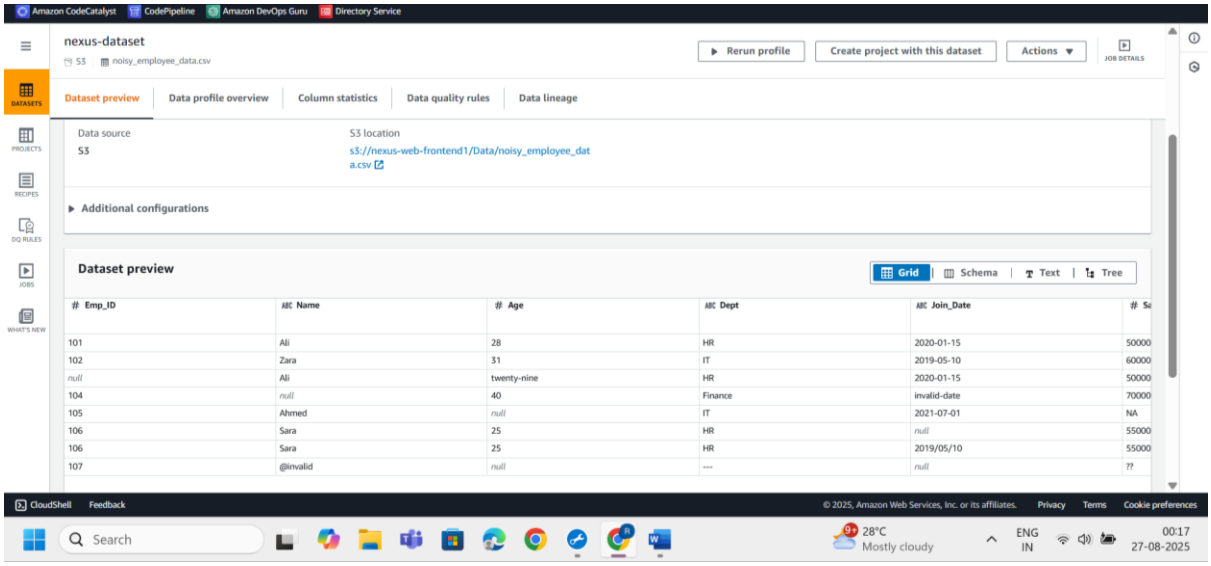
🛠 **Technologies Used**:

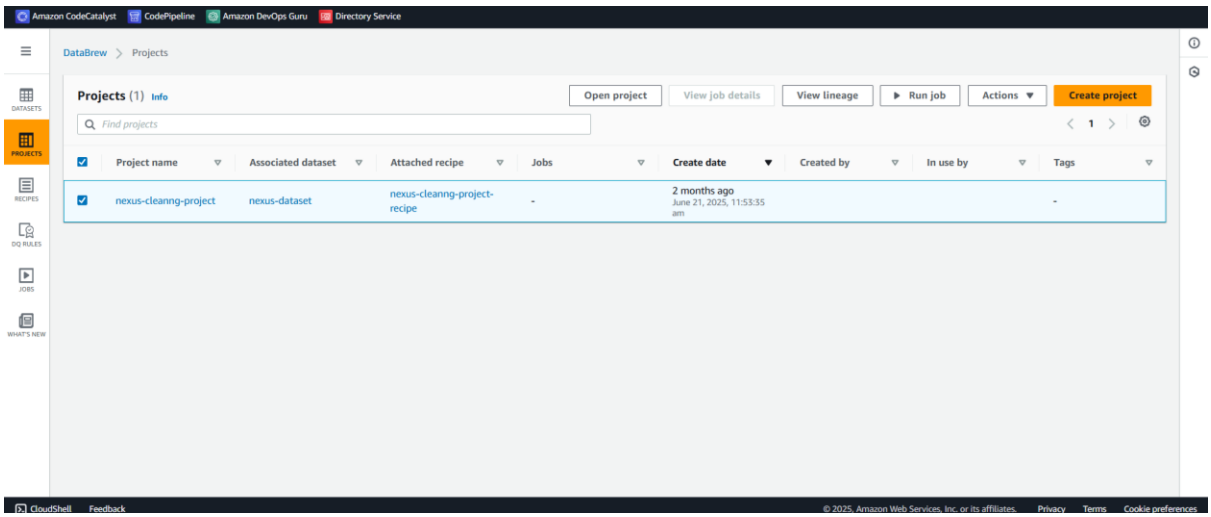
- AWS Glue DataBrew

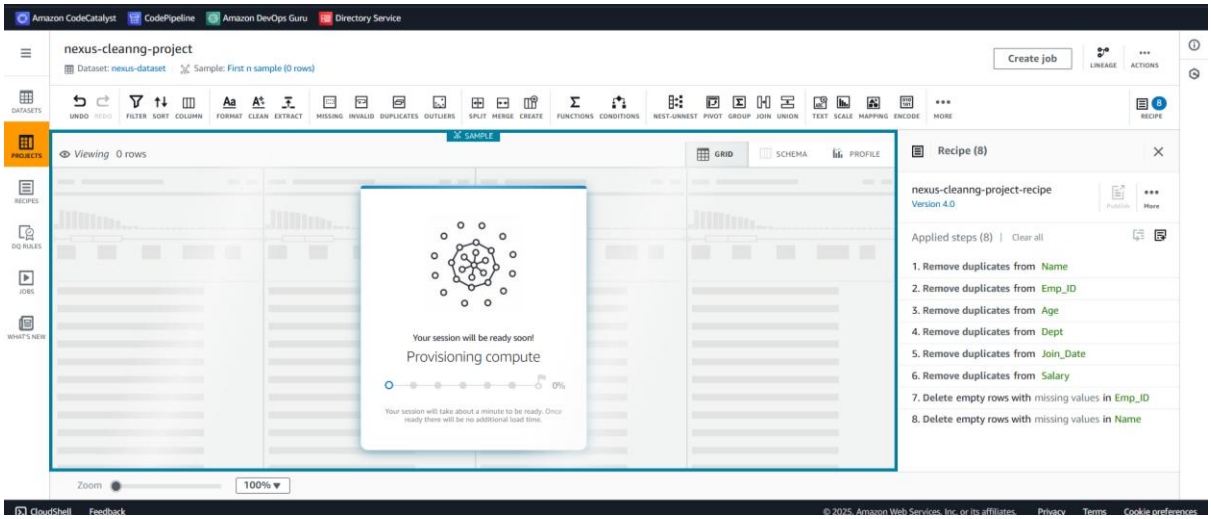
📸 **Suggested Screenshots**:

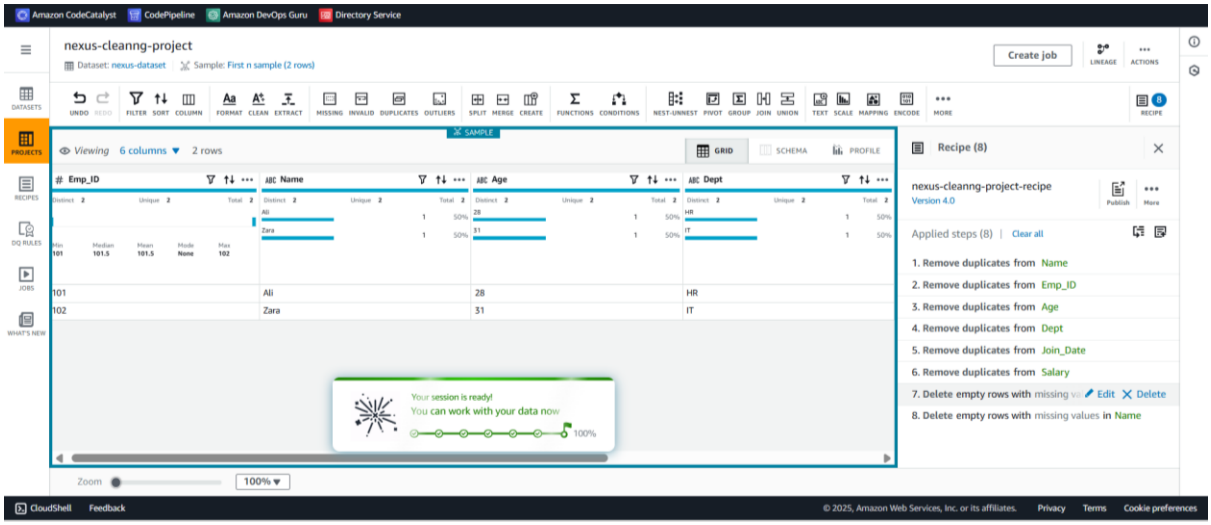
- DataBrew project and recipe
- Preview of cleaned results
- Profile summary (data quality insights)

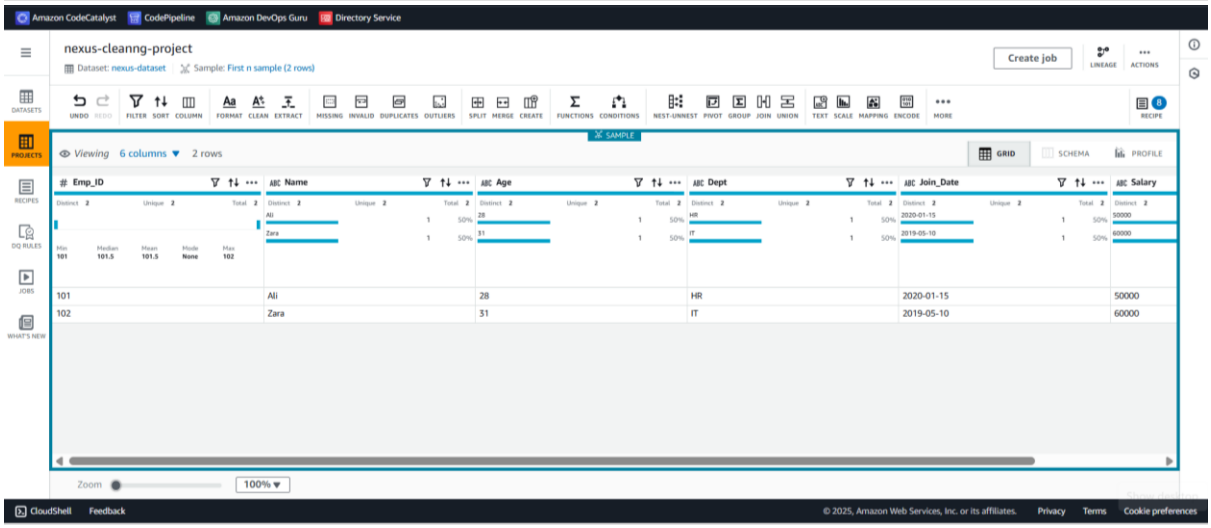


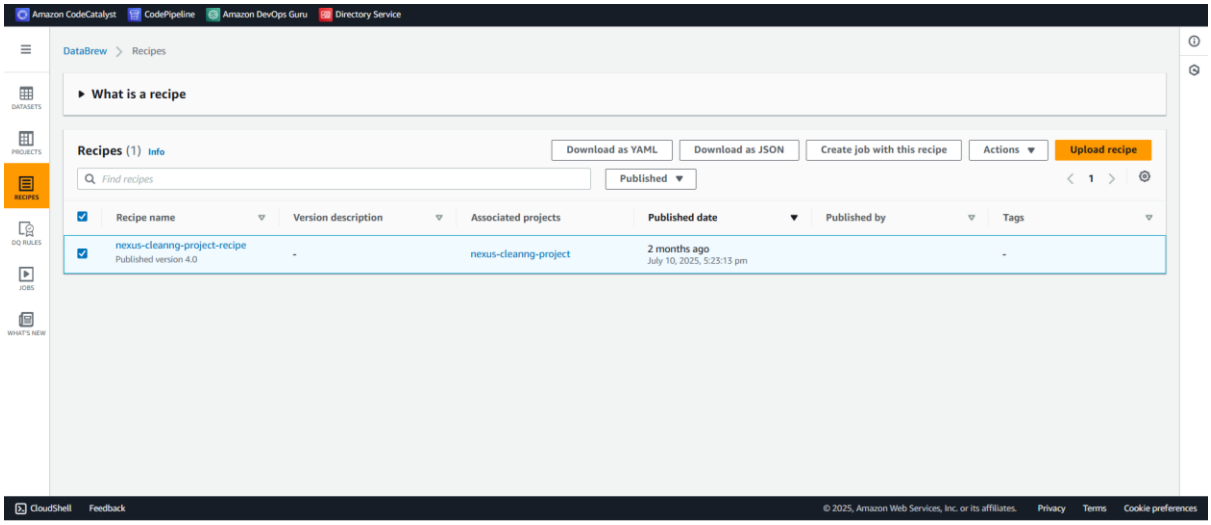


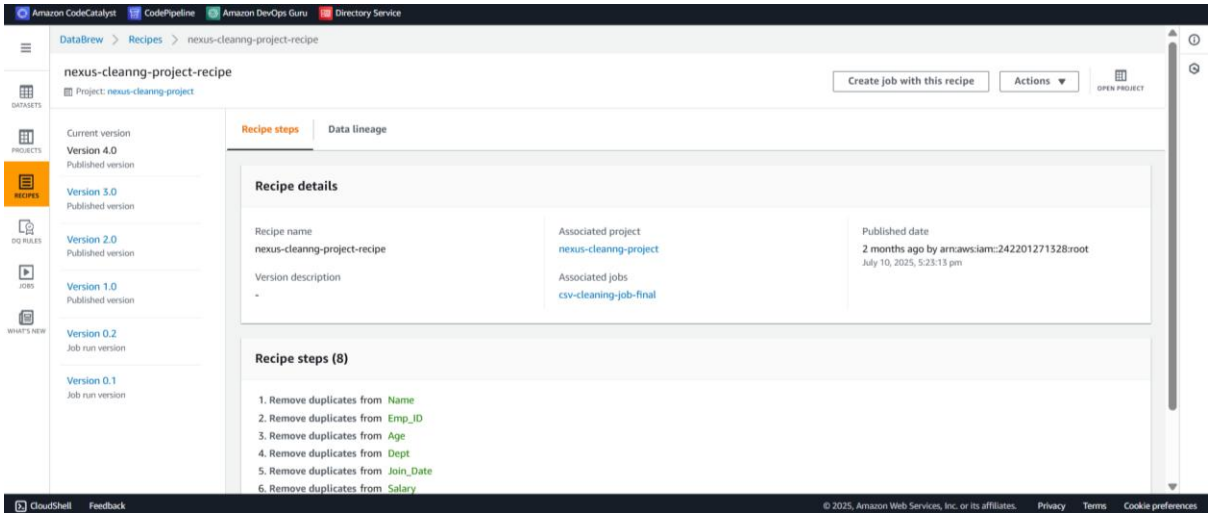


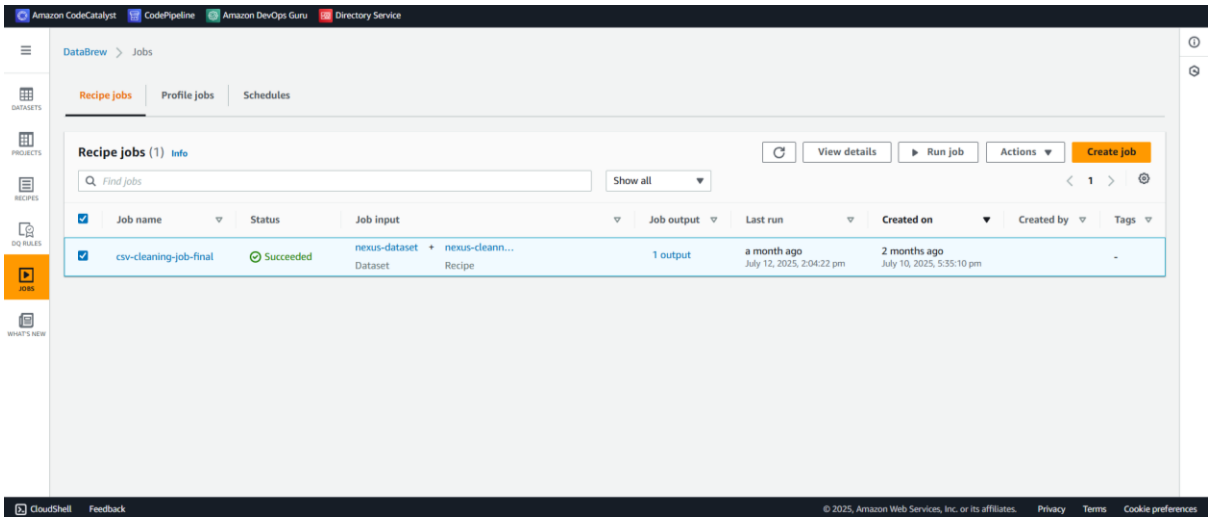


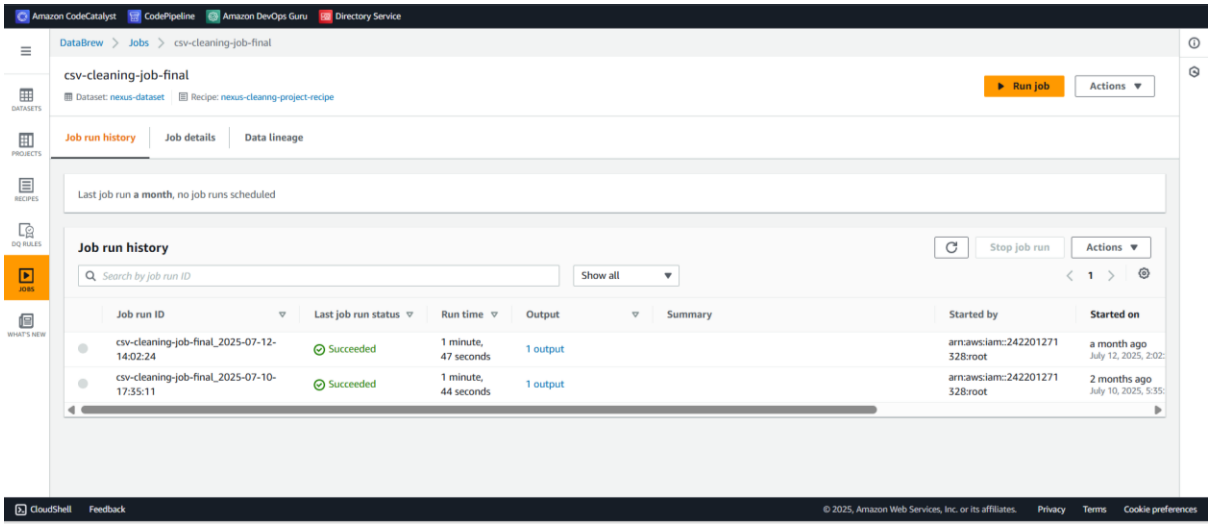












7.4. Second Lambda Invocation

Once the Glue Job completes, it **invokes a second Lambda function** to automate the following:

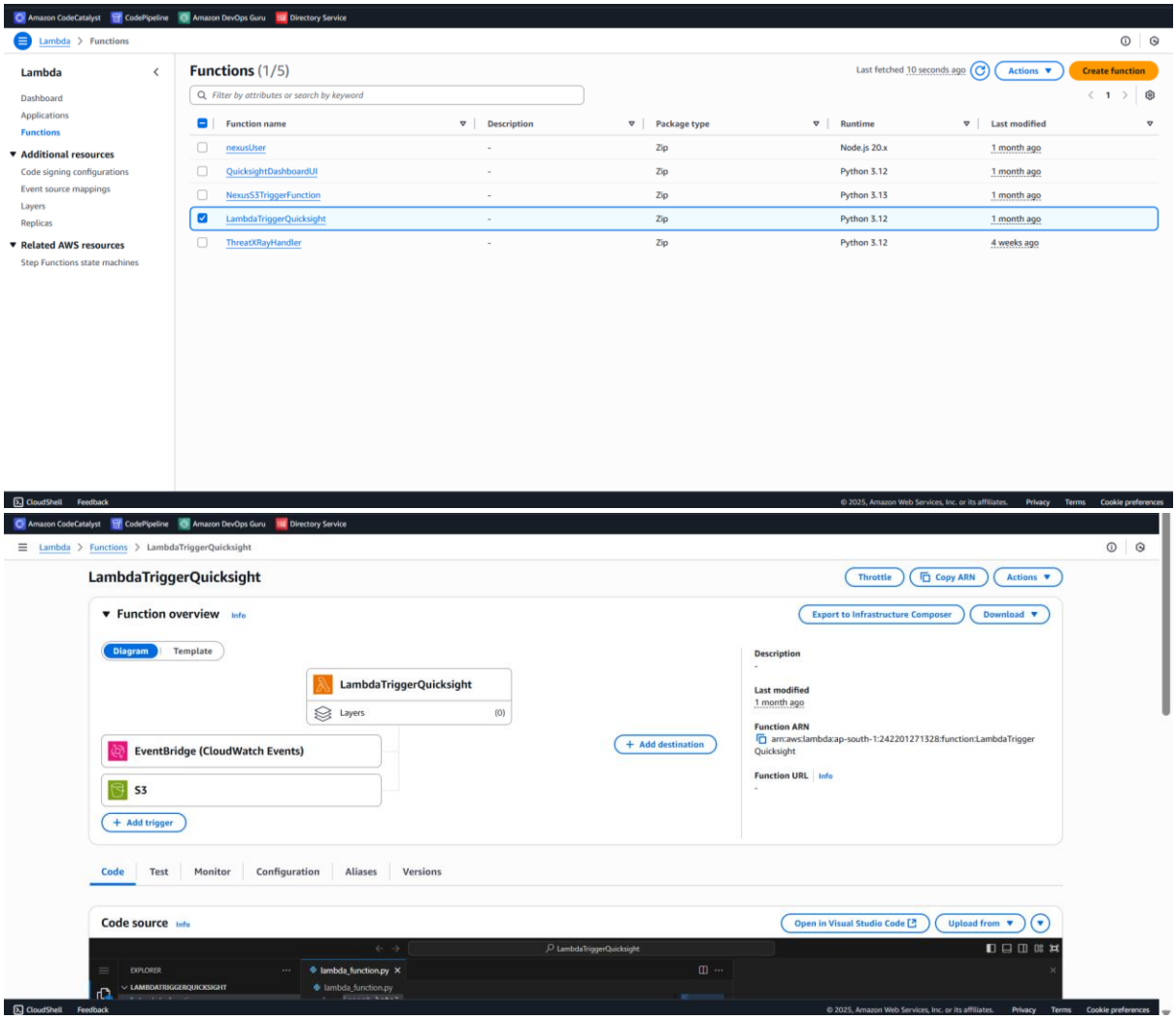
- Trigger a second Glue Crawler for the cleaned data
- Configure Athena dataset
- Automatically generate a QuickSight dashboard using a predefined template

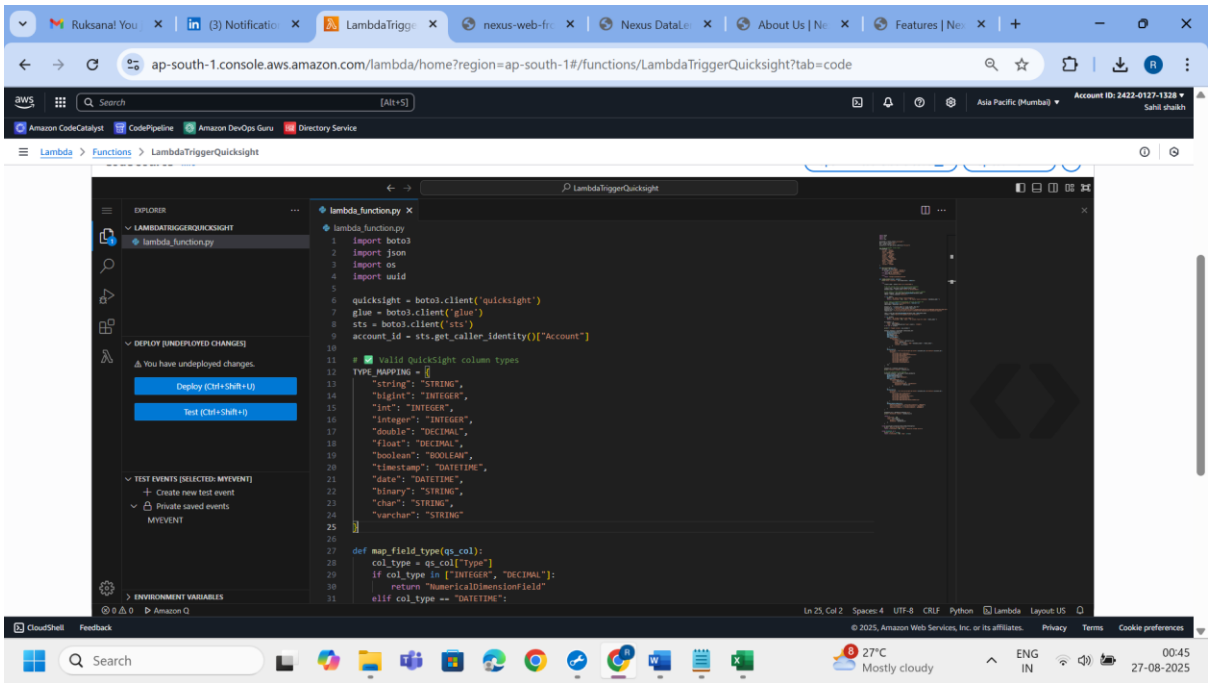
🛠 **Technologies Used**:

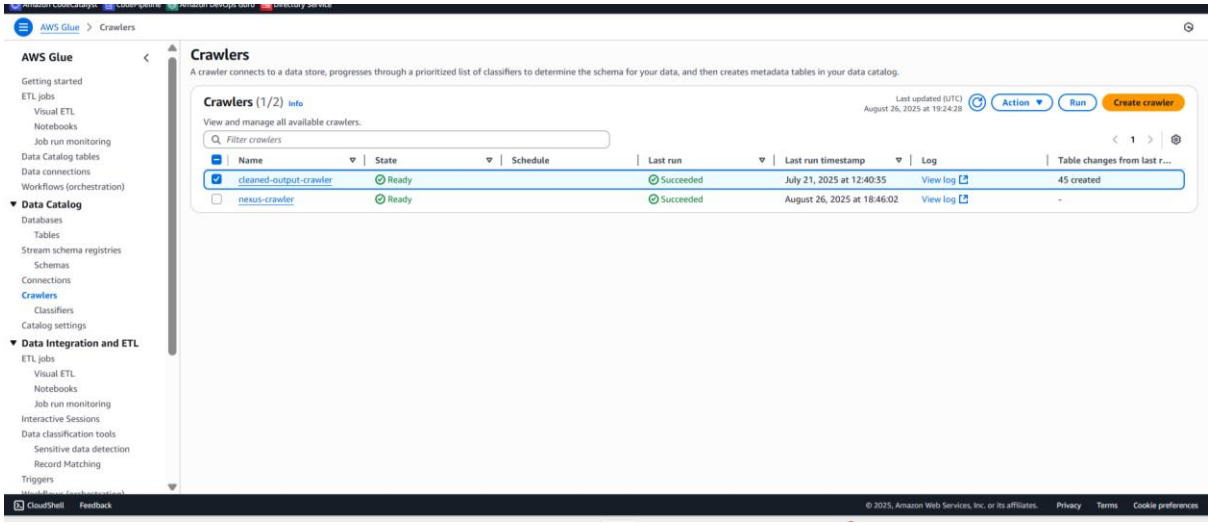
- AWS Lambda
- AWS Glue Crawler
- Athena dataset creation via Boto3
- QuickSight automation via SDK

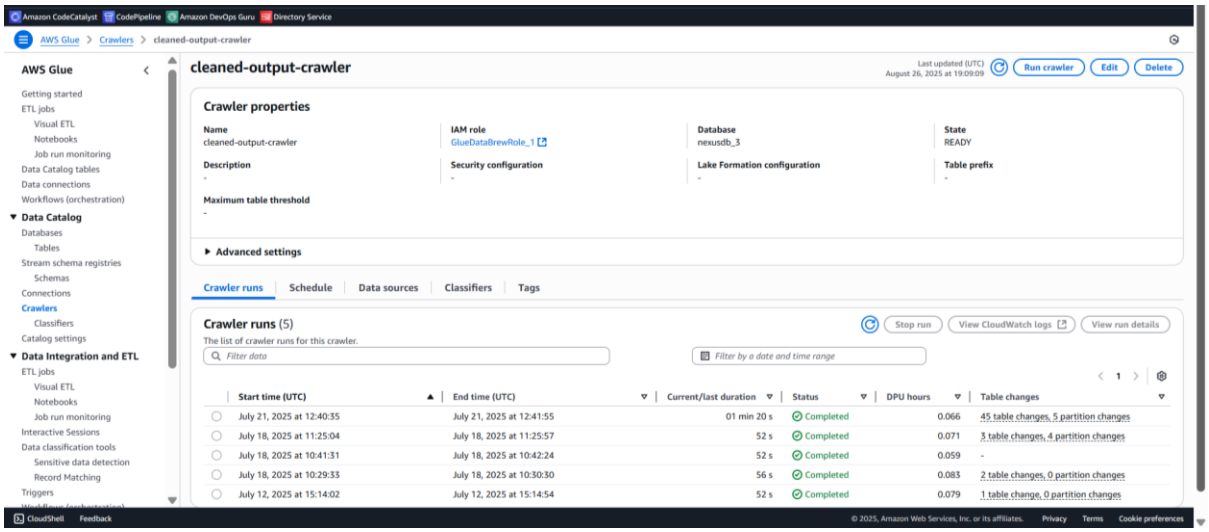
📸 *Screenshot Suggestions:*

- Lambda code snippet invoking dashboard generation
- Glue Crawler for cleaned output



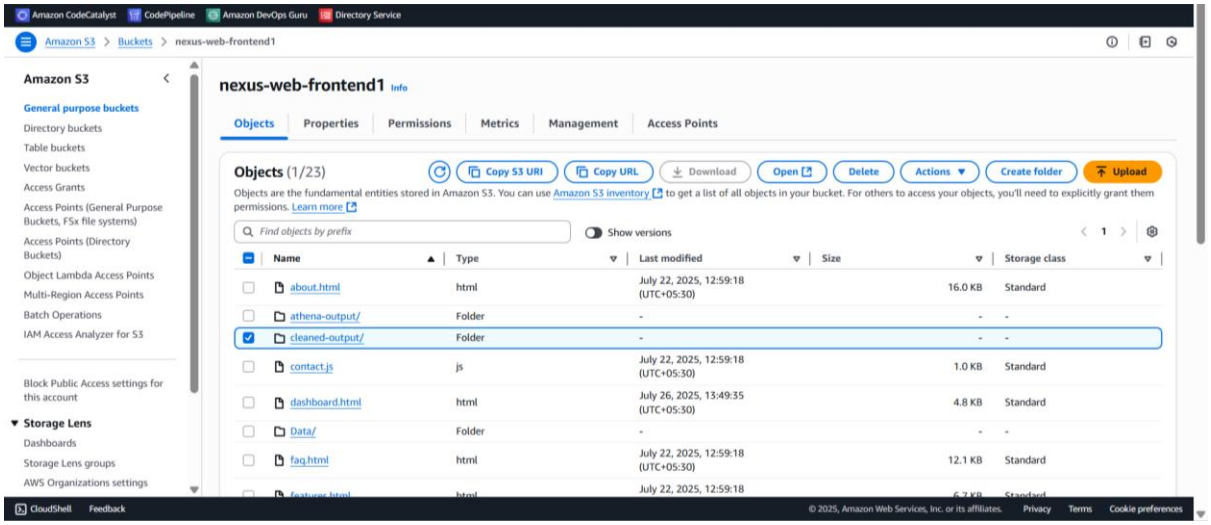


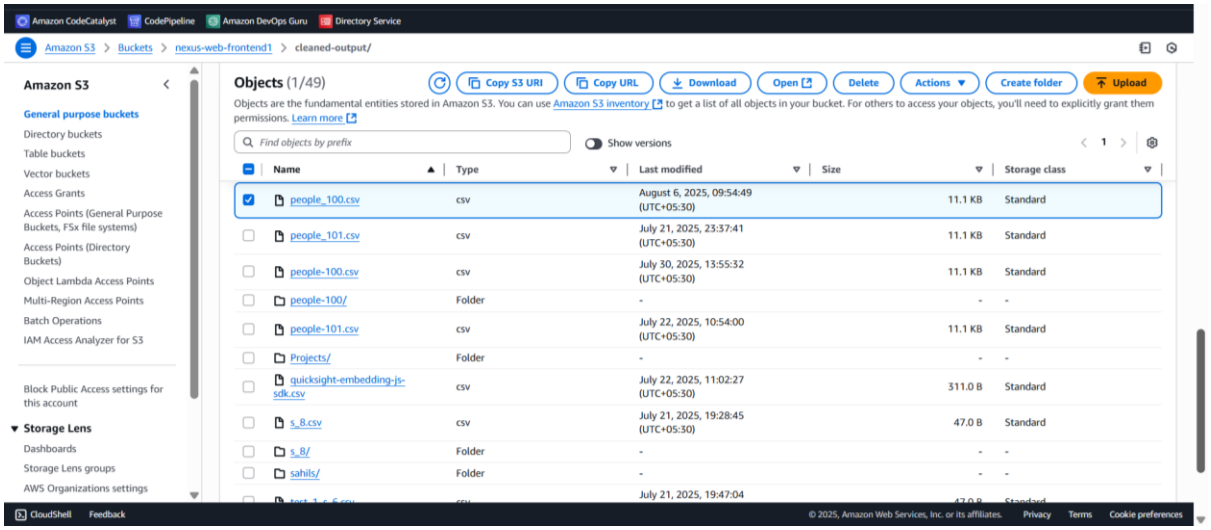




8. Data Storage & Querying

8.1 Cleaned Data Output in S3





8.2 Querying via Athena

- Athena enables serverless querying using standard SQL.
- Cleaned data in S3 is queried via the Glue Catalog tables.

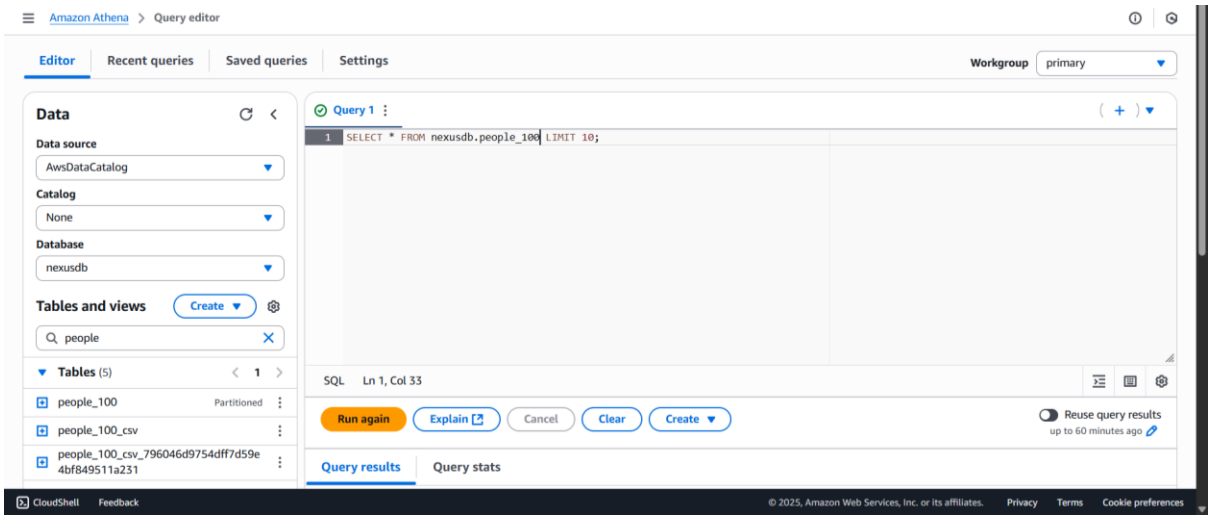
Example query:

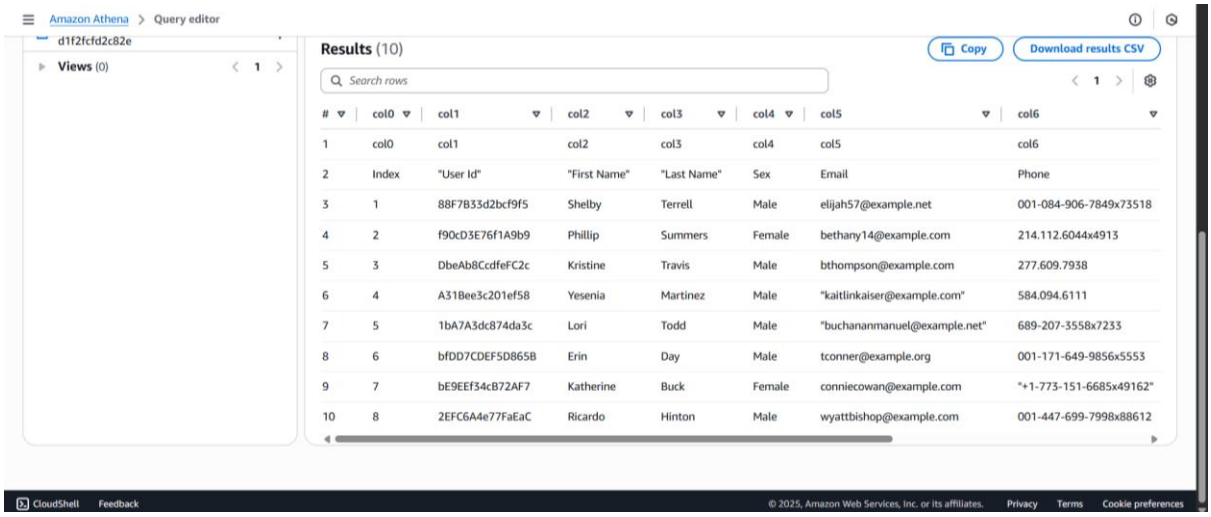
SELECT * FROM nexusdb.cleaned_<filename> LIMIT 10;

📸 *Screenshot Suggestions:*

- Athena query editor with sample query

- Preview of query results





9. Visualization and Reporting

9.1 Data Visualization – QuickSight & Power BI

- In **Amazon QuickSight**, dashboards are generated automatically using Athena datasets.
- In **Microsoft Power BI**, users can connect via the **Athena ODBC driver** for custom reporting.

🛠 **Technologies Used**:

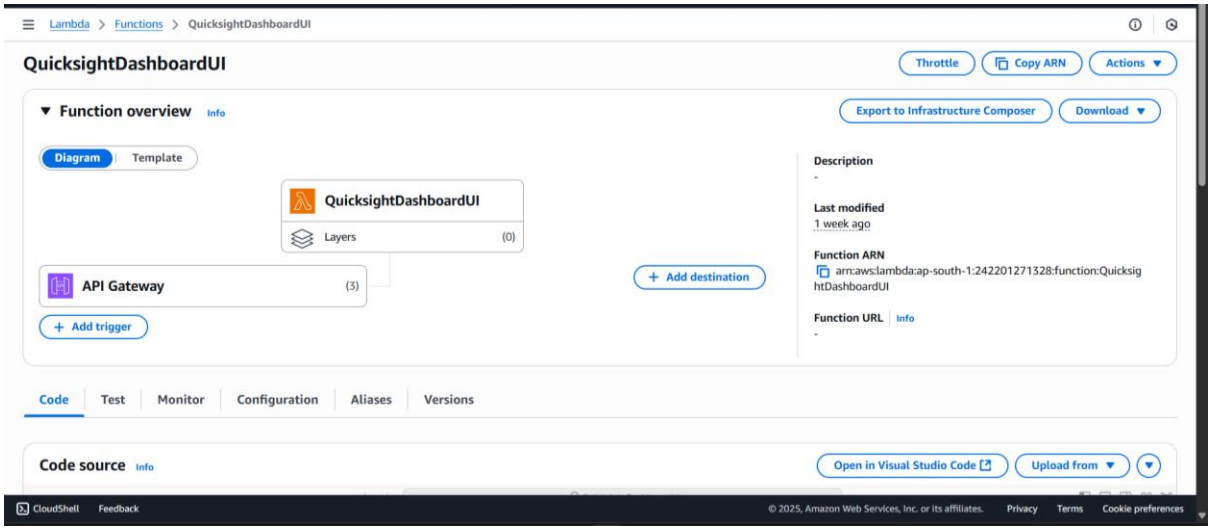
- Amazon QuickSight
- Microsoft Power BI (Optional)
- Athena ODBC Connector

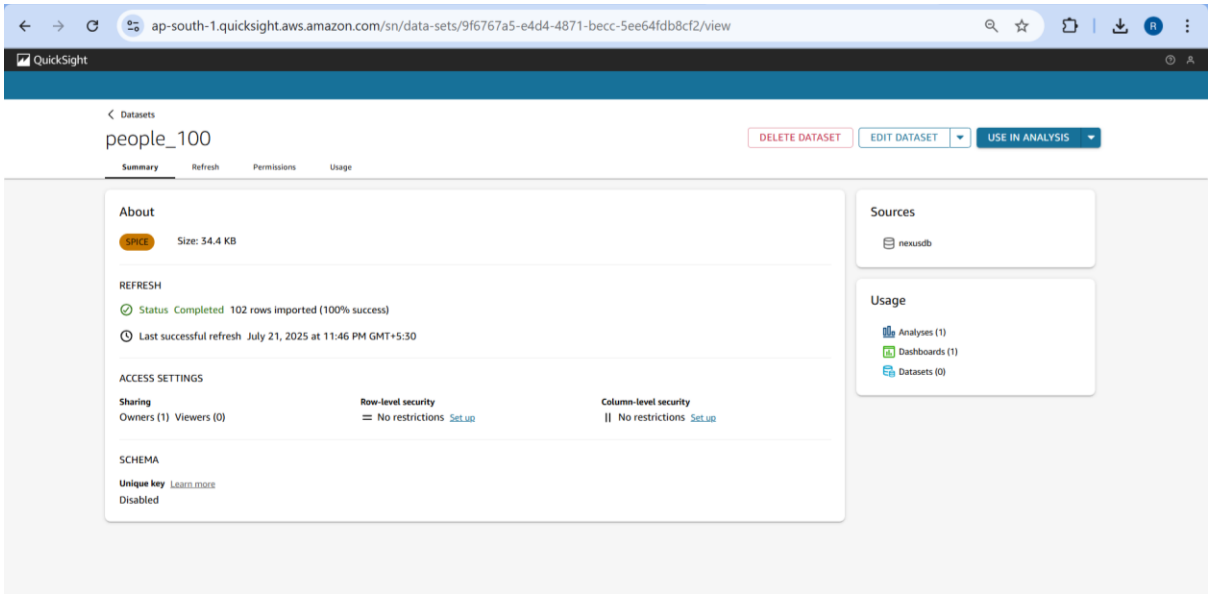
📸 **Suggested Screenshots**:

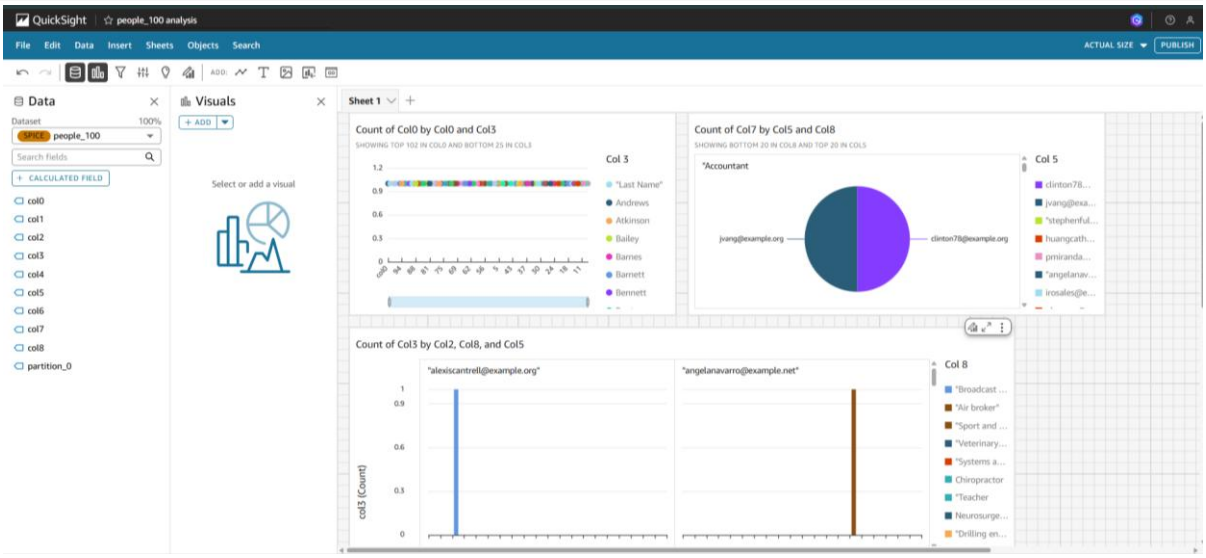
- QuickSight dashboards (bar, pie, line)
- Power BI report preview (same dataset)
- ODBC config panel in Power BI

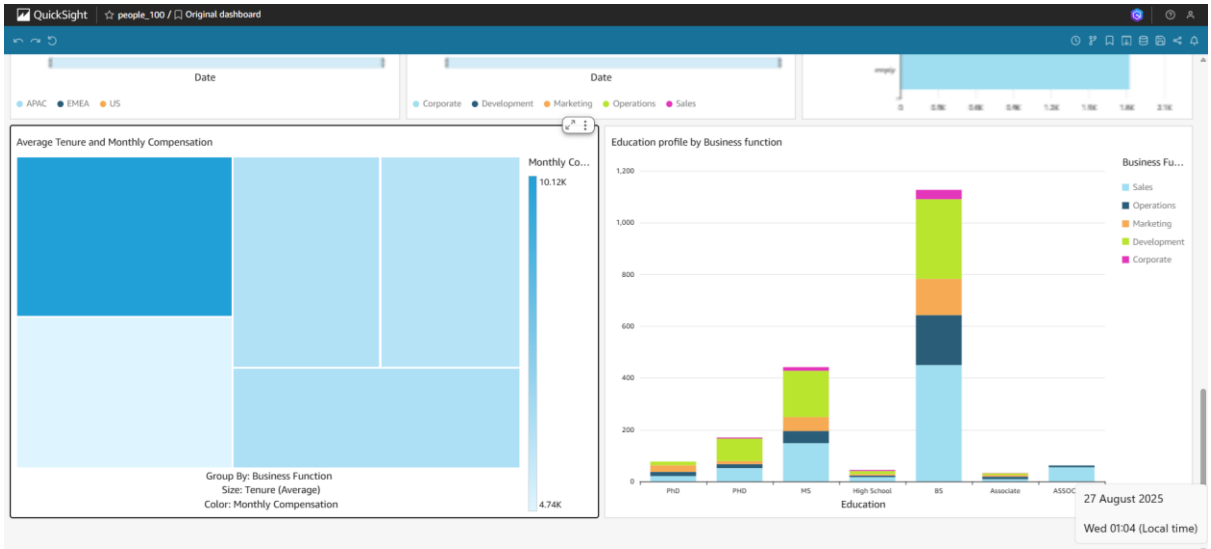
📸 *Screenshot Suggestions:*

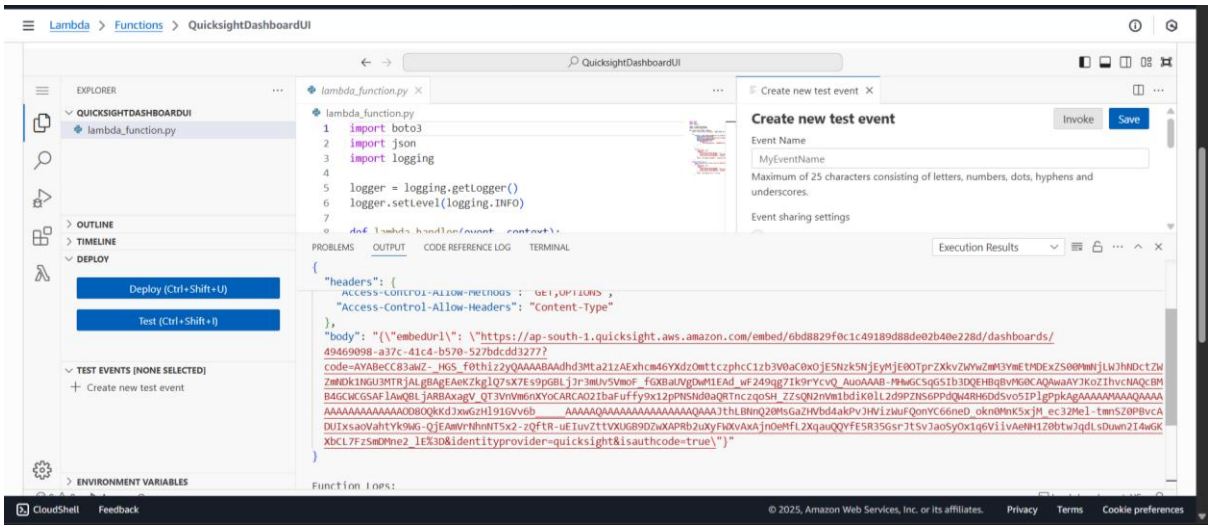
- QuickSight dashboard with graphs/charts
- Power BI connected to Athena with a sample visual



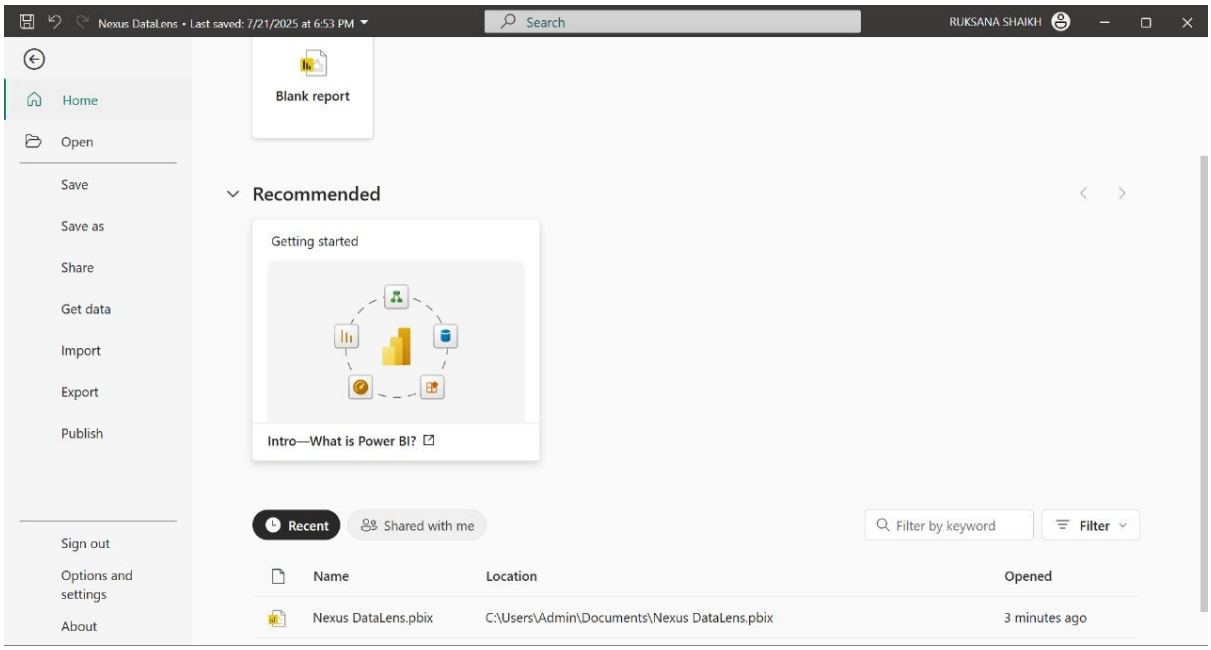


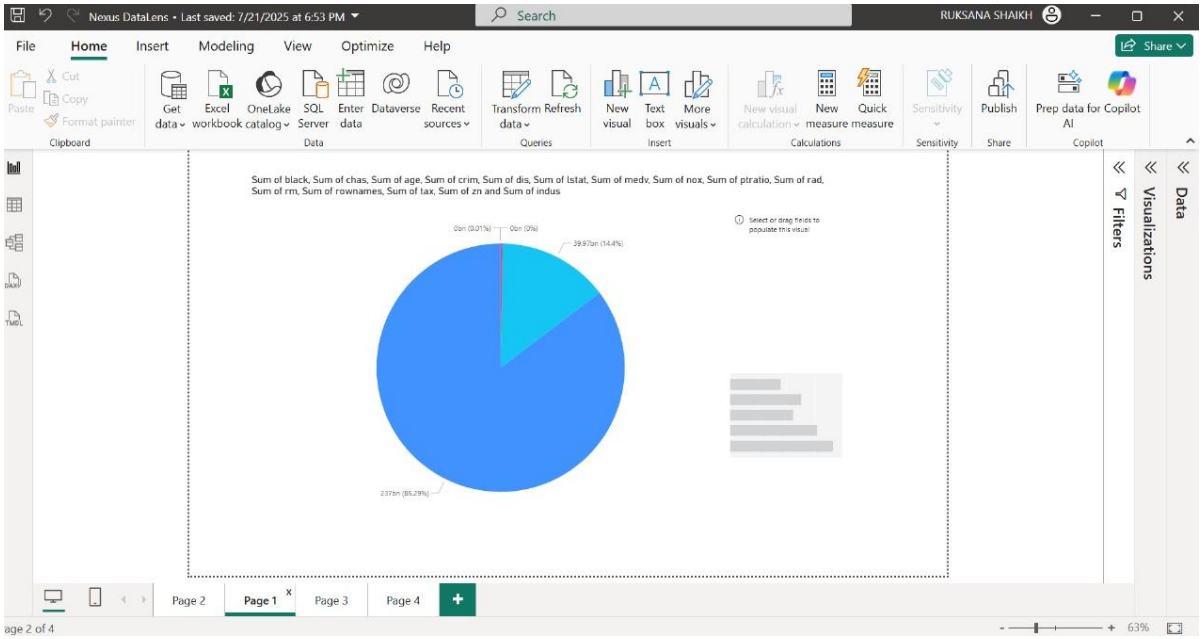


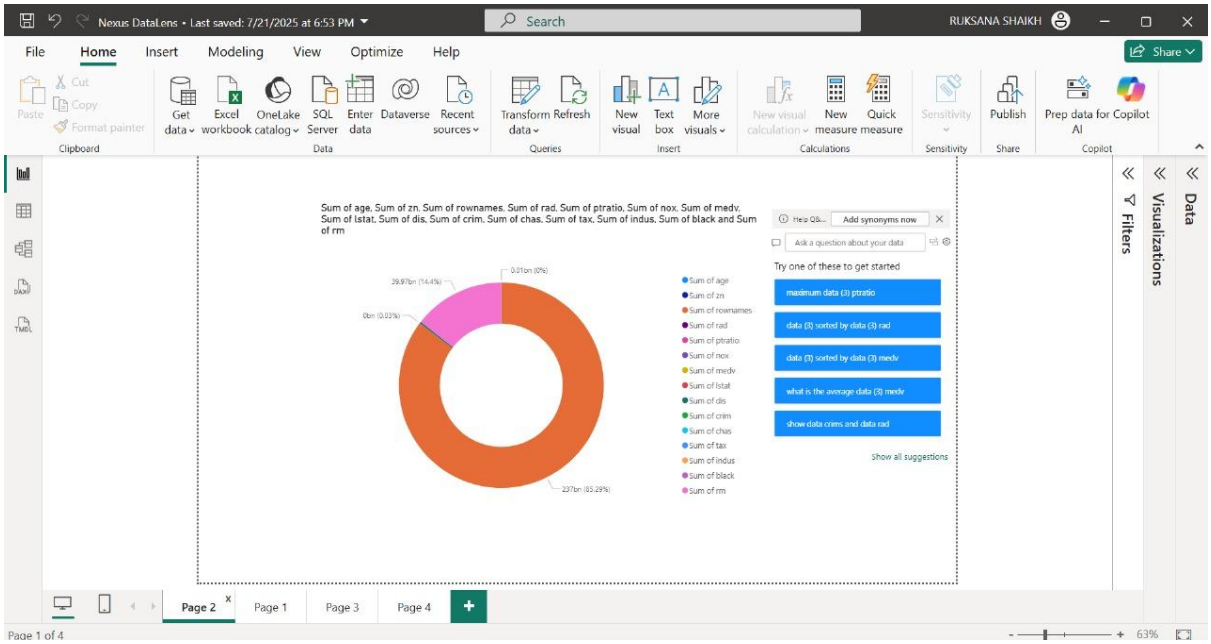


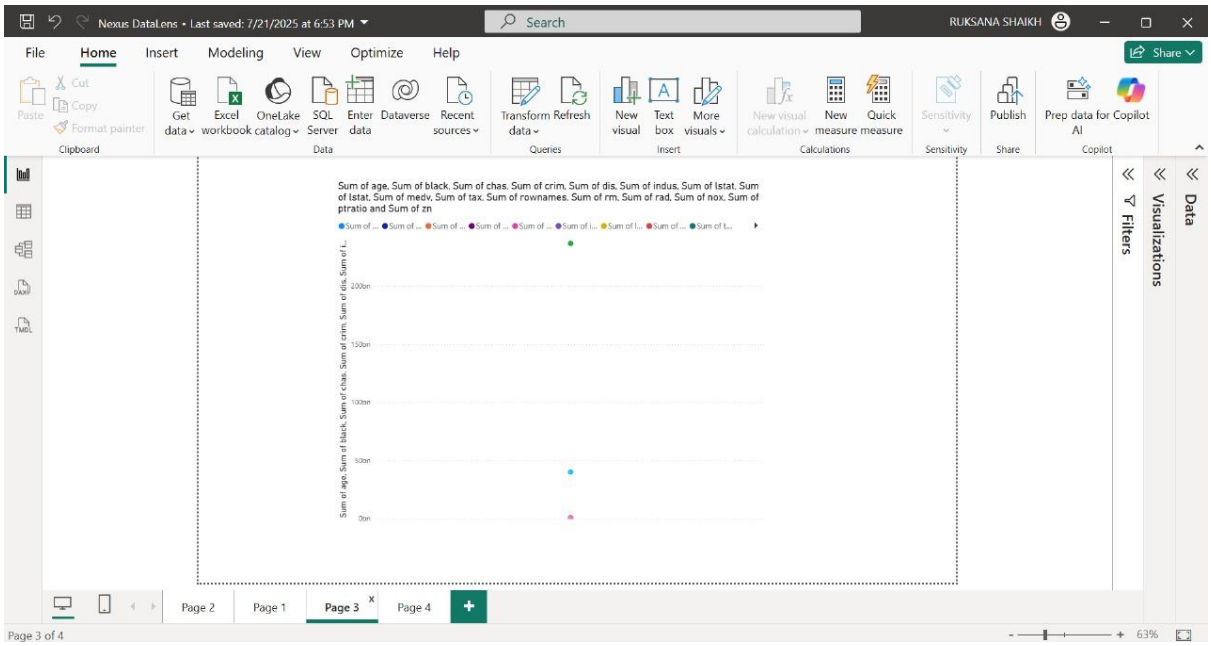


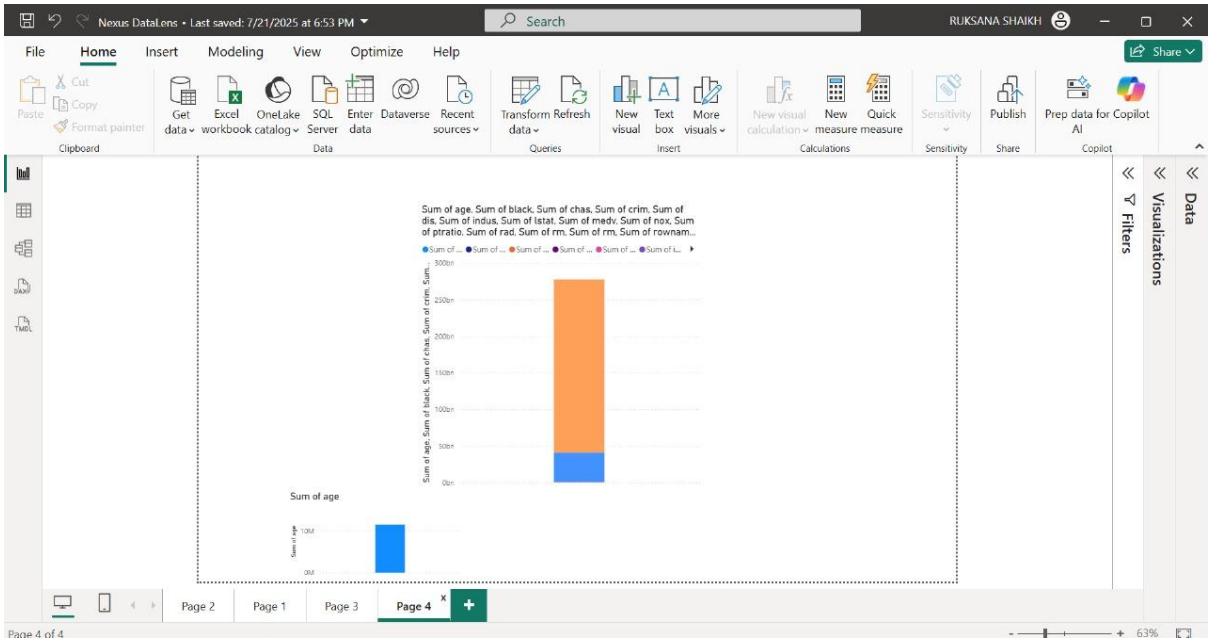
9.2 Power BI Integration (Optional)











10. Challenges and Resolutions

Throughout the development of *Nexus DataLens*, several technical and architectural challenges were encountered. Each of these obstacles presented an opportunity to strengthen the system's robustness and reliability. Below is a summary of the key challenges faced and the corresponding solutions implemented.

1. Incorrect S3 Path in Glue Crawler Configuration

Challenge:

Initially, the AWS Glue Crawler was configured to scan an outdated S3 bucket (nexus-web-front), which resulted in the creation of metadata tables pointing to obsolete data locations. This caused downstream services like Athena and QuickSight to display empty or outdated datasets.

Resolution:

The crawler configuration was updated to target the correct S3 path (nexus-web-frontend1/data/). Existing Glue tables linked to the old location were manually deleted from the Data Catalog, and a fresh crawl was initiated to reflect accurate schema metadata in Athena.

2. Frontend Upload Failure Due to S3 Bucket Policy

Challenge:

CSV file uploads from the frontend to S3 were initially unsuccessful due to restrictive CORS policies and insufficient S3 permissions.

Resolution:

The S3 bucket policy was carefully updated to allow secure PUT access from the frontend domain. Cross-Origin Resource Sharing (CORS) rules were configured to allow the appropriate HTTP methods. This enabled seamless file transfers while maintaining security boundaries.

3. Lambda Execution Timeout During Large File Processing

Challenge:

When handling larger datasets, the Lambda function occasionally exceeded its default timeout, disrupting the automated trigger for Glue Crawlers and Jobs.

Resolution:

The function timeout setting was increased within AWS Lambda to accommodate longer processing durations. Logging and exception handling were also enhanced to improve reliability and traceability for future executions.

4. Duplicate Schema Entries in Glue Catalog

Challenge:

Multiple crawls on the same S3 location led to the creation of duplicate tables in the Glue Catalog, causing confusion in downstream Athena queries and QuickSight reports.

Resolution:

To resolve this, a naming convention and scheduled crawler execution strategy were implemented. Old tables were cleaned up, and table creation settings were restricted to avoid duplication unless schema changes were detected.

5. QuickSight Dataset Not Refreshing Automatically

Challenge:
After successful data cleaning and schema updates, QuickSight dashboards were not reflecting the latest data due to the absence of automatic dataset refresh.
Resolution:
A manual refresh was triggered initially. Later, this was resolved by integrating QuickSight dataset update logic via Lambda post Glue Job execution. This ensured the dashboard always presents the most recent data.

11. Key Features & Benefits

Nexus DataLens offers a robust set of features designed to simplify the data analytics lifecycle while ensuring scalability, automation, and enterprise-grade security.

Key Features:

- **End-to-End Automation –** From CSV upload to dashboard generation, the entire pipeline runs automatically without manual intervention.
- **Serverless Architecture –** Built entirely on AWS managed services (S3, Lambda, Glue, Athena, QuickSight), ensuring scalability and cost-efficiency.
- **User-Friendly Frontend –** A simple web interface for non-technical users to upload CSV files securely.
- **Flexible Data Cleaning –** Supports both code-based cleaning via AWS Glue Jobs and no-code transformations using AWS Glue DataBrew.
- **Seamless Visualization –** Provides dynamic dashboards in Amazon QuickSight with optional integration into Microsoft Power BI.
- Enterprise-Grade Security – Implements AWS IAM for access control, KMS for encryption, and CloudWatch for monitoring.

Benefits:

- **Reduced Effort & Time –** Eliminates manual data preparation, accelerating insights.
- **Scalability on Demand –** Automatically adapts to different workloads without infrastructure management.
- **Improved Data Accuracy** – Automated cleaning minimizes human error and ensures consistent datasets.
- **Accessibility for All Users –** Enables non-technical users to generate insights effortlessly.
- **Cost-Effective Solution –** Pay-as-you-go pricing model of AWS serverless services reduces operational costs.

12. Use Cases of Nexus DataLens

The *Nexus DataLens* platform is designed as a scalable, serverless data analytics solution that caters to a variety of real-world business and analytical needs. Its architecture allows organizations to ingest, clean, query, and visualize data with minimal manual effort. Below are key use cases where Nexus DataLens can deliver significant value:

1. Business Intelligence & Reporting Automation
Problem: Businesses often deal with scattered raw data that requires time-consuming preprocessing before analysis.
Solution: Nexus DataLens enables automated transformation of raw CSV files into clean, queryable datasets, and visualizes them through Amazon QuickSight or Power BI dashboards.
Impact: Saves analyst time, enables faster decision-making, and improves data accuracy.

2. Operational Monitoring & KPI Dashboards
Problem: Monitoring KPIs (Key Performance Indicators) across departments can be difficult without centralized visualization.
Solution: Uploads from various teams can be cleaned and visualized automatically into live dashboards using QuickSight or Power BI.
Impact: Enables department heads to monitor sales, operations, customer satisfaction, or finance in real time.

3. Data Democratization for Non-Technical Users
Problem: Non-technical users struggle to query data or generate insights.
Solution: With a simple frontend upload UI, anyone can upload CSV files and get visualized insights without needing to write code.
Impact: Makes data-driven decision-making accessible to marketing, HR, or sales teams.

4. Academic or Research Data Analysis
Problem: Researchers need a quick way to analyze large CSV-based datasets without complex setups.
Solution: Researchers can upload their raw data, clean it, and immediately query or visualize results using built-in analytics tools.
Impact: Accelerates data-driven research and facilitates reproducibility.

5. Client/Partner Data Integration for Enterprises
Problem: Receiving external data from clients or partners often requires preprocessing before use.
Solution: External partners can upload CSVs via the secure frontend; Nexus DataLens automatically prepares it for analysis.
Impact: Speeds up B2B data integration and fosters seamless collaboration.

6. Government or NGO Reporting Dashboards
Problem: Public agencies and NGOs handle data from multiple sources and require transparent reporting.
Solution: Nexus DataLens provides automated processing and visualization, ideal for transparency

and compliance.
Impact: Facilitates evidence-based reporting to stakeholders and donors.

7. Startup MVP for Analytics-as-a-Service
Problem: Building a custom analytics backend from scratch is time- and resource-intensive.
Solution: Startups can white-label Nexus DataLens as a backend for providing analytics services to their users.
Impact: Reduces go-to-market time and cost for SaaS products.

8. Internal Tool for ETL Process Validation
Problem: Data engineers often need to test ETL transformations before deployment.
Solution: Engineers can use Nexus DataLens to validate ETL logic on sample CSVs and visualize the output.
Impact: Ensures data quality and reduces post-deployment errors.

13. Results and Outcomes

The implementation of **Nexus DataLens** successfully demonstrated the value of an automated, serverless data analytics pipeline on AWS. The project delivered measurable improvements across multiple dimensions:

- **Seamless Automation** – Achieved a fully automated workflow where CSV files uploaded via the frontend are processed, cleaned, and visualized without manual intervention.
- **Data Accuracy** – Cleaning logic in AWS Glue Jobs ensured removal of null values, inconsistent headers, and invalid entries, producing high-quality datasets ready for analysis.
- **Faster Time to Insights** – Reduced the end-to-end process (from file upload to dashboard generation) from hours of manual effort to just a few minutes.
- **Scalability** – Validated that the serverless architecture can handle growing data volumes without additional infrastructure management.
- **Visualization Success** – Generated interactive dashboards in **Amazon QuickSight** and validated **Power BI integration**, confirming flexibility for different BI environments.
- **Enhanced Accessibility** – Enabled non-technical users to generate insights through a simple upload interface, removing dependency on specialized data engineers.

Overall Outcome: Nexus DataLens proved to be a secure, cost-effective, and extensible solution that can be applied across business, research, and organizational use cases to accelerate data-driven decision-making.

14. Conclusion and Future Enhancements

14.1 Recommendations

To further enhance the robustness, usability, and scalability of the Nexus DataLens platform, several strategic improvements are proposed. These recommendations aim to optimize performance, reduce manual overhead, and extend functionality to support enterprise-scale analytics needs.

1. File Validation Prior to Processing
Implementing pre-ingestion validation checks for uploaded files—such as verifying size limits, acceptable formats (e.g., CSV only), and schema conformity—would help prevent processing failures and ensure data consistency.
2. Automated Cleanup of Obsolete Data
Introduce lifecycle management for raw and processed datasets stored in Amazon S3. Automating the deletion of outdated files, logs, and dashboards would not only reduce storage costs but also maintain a clutter-free data environment.
3. Event-Based Notifications
Incorporating Amazon SNS or SES to notify users upon pipeline completion, failure, or anomalies would provide real-time operational transparency. This is particularly useful in multi-user or team environments where timely updates are crucial.
4. Integration with Redshift or RDS
Extending data storage and querying capabilities by integrating with Amazon Redshift or Amazon RDS would enable persistent and structured storage for large-scale data warehousing and advanced analytics.

14.2 Future Scope

Looking ahead, the Nexus DataLens platform can evolve significantly to support broader use cases and provide deeper insights:

- **Support for Additional File Formats**
  Extend compatibility beyond CSV to include formats such as Excel (XLSX) and JSON, thereby enabling a wider range of data sources to be processed seamlessly.
- **Machine Learning-Based Anomaly Detection**
  Integrate Amazon SageMaker or similar ML services to automatically detect anomalies and outliers during the data cleaning process, enabling smarter and more proactive data quality assurance.
- **Role-Based Dashboards and Access Controls**
  Enhance user experience by enabling personalized dashboards based on roles, departments, or teams. AWS Cognito and IAM policies can be used to implement granular access controls.
- **Comprehensive Notification System**
  Expand the notification mechanism to support both email (SES) and SMS-based alerts (SNS), offering flexible communication options tailored to user preferences.

15. References

1. **Amazon S3 – Object Storage Service**
   https://docs.aws.amazon.com/s3

2. **AWS Lambda – Serverless Compute**
   https://docs.aws.amazon.com/lambda

3. **AWS Glue – Data Integration & ETL**
   https://docs.aws.amazon.com/glue

4. **Amazon Athena – Query Service**
   https://docs.aws.amazon.com/athena

5. **Amazon QuickSight – Business Intelligence**
   https://docs.aws.amazon.com/quicksight

6. **Amazon API Gateway – Managed API Service**
   https://docs.aws.amazon.com/apigateway

17. Appendix

17.1 Source Code Links (GitHub)

The complete source code, including frontend files, Lambda functions, and Glue job scripts, is available in the project's GitHub repository:
👉 GitHub Repository – Nexus DataLens 2.0

17.2 Sample Input/Output

- **Sample Input (Raw CSV):** A noisy employee dataset with missing values, inconsistent headers, and placeholder entries.
- **Sample Output (Cleaned CSV):** The processed dataset generated by the Glue Job, containing standardized headers, filtered values, and a single flat CSV file stored in s3://nexus-web-frontend1/cleaned-output/.

17.3 Additional Screenshots

Below are supplementary screenshots to illustrate the end-to-end pipeline:

- Frontend upload interface
- S3 bucket structure (raw data vs. cleaned output)
- AWS Lambda configuration and CloudWatch logs
- Glue Crawler and Glue Job setup
- Athena query editor with sample SQL results
- Amazon QuickSight dashboards (bar, line, and pie charts)
- Power BI report (optional integration demo)