Descriptive Analysis 1

[Project Description: 1](#_Toc1430229208)

[Project Title: 1](#_Toc1649424798)

[Objective: 1](#_Toc1985794350)

[Dataset: 1](#_Toc253398588)

[Methodology: 1](#_Toc1805645163)

Tools and Technologies: 3

[Deliverables: 3](#_Toc1995117632)

[AWS Deployment and Service Models 4](#_Toc914194637)

[AWS Cost Analysis 7](#_Toc2032893621)

[AWS Global infrastructure 10](#_Toc1070462277)

[AWS IAM 11](#_Toc1138330697)

[AWS VPC 12](#_Toc158185041)

AWS Lambda 12

[AWS EBS 13](#_Toc1021751728)

# **Descriptive Analysis**

**Project Description:** Cloud-Based Data Pipeline for Financial Data Management

**Project Title:** Design and Implementation of a Cloud-Based ETL Workflow for Financial Data at University Canada West

**Objective**: The primary goal of this project is to design, evaluate, and implement a cloud-based data pipeline for financial data. The focus is on understanding data ingestion costs, performing cleaning and profiling, and developing a scalable ETL (Extract, Transform, Load) system using AWS services. This project emphasizes practical application of cloud computing concepts to enhance data quality, cost-efficiency, and system reliability.

**Dataset:** The dataset originates from a finance group assignment and includes sample business and transactional data. Key features include:

* **Business Questions:** Contextual queries related to financial analysis
* **Raw Financial Data:** Used for ingestion, cleaning, and profiling
* **Metadata:** Structure and relationships within the financial datalake
* **Profiling Attributes:** Null values, duplicates, data types, and inconsistencies

**Methodology:**

**1. Business Analysis and Initial System Design (Week 2)**

* + **Business Question Analysis:**  
    Interpreted Business Problem 3 from the finance domain to identify data and system requirements.
  + **Root Cause Identification:**  
    Created a Fishbone diagram to explore contributing factors to the data challenge.
  + **Datalake Design:**
  + Drafted an Excel-based datalake structure.
  + Visualized the architecture in draw.io for a clear system blueprint.
  + **Infrastructure Setup:**  
    Deployed an EC2 instance and configured a VPC and security group in AWS to simulate the environment.

**2. Cost Evaluation and Data Cleaning Design (Week 3)**

* + **Cost Analysis of Data Ingestion:**  
    Used the AWS Pricing Calculator to evaluate data transfer and storage costs.
  + **Data Cleaning Blueprint:**
  + Listed common data issues and resolution strategies in Excel.
  + Visualized the cleaning workflow in draw.io (handling nulls, types, duplicates).
  + **Implementation of Cleaning:**  
    Conducted profiling and applied cleaning techniques to improve data quality using scripts and tools.

**3. Dataset Profiling and ETL Development (Week 4)**

* + **Profiling Cost Evaluation:**  
    Estimated resource costs for profiling and cleaning tasks.
  + **ETL Workflow Design:**  
    Defined an end-to-end ETL system integrating previously cleaned data.
  + **ETL Implementation:**  
    Developed and tested ETL scripts to automate data movement, transformation, and loading.

## 

## **Tools and Technologies:**

 **AWS Cloud Platform:** EC2, VPC, Security Groups

 **Excel:** Data structure planning and issue tracking

 **Draw.io:** Architecture and process visualization

 **AWS Pricing Calculator:** Cost estimation

 **Scripting Tools:** For cleaning and ETL execution

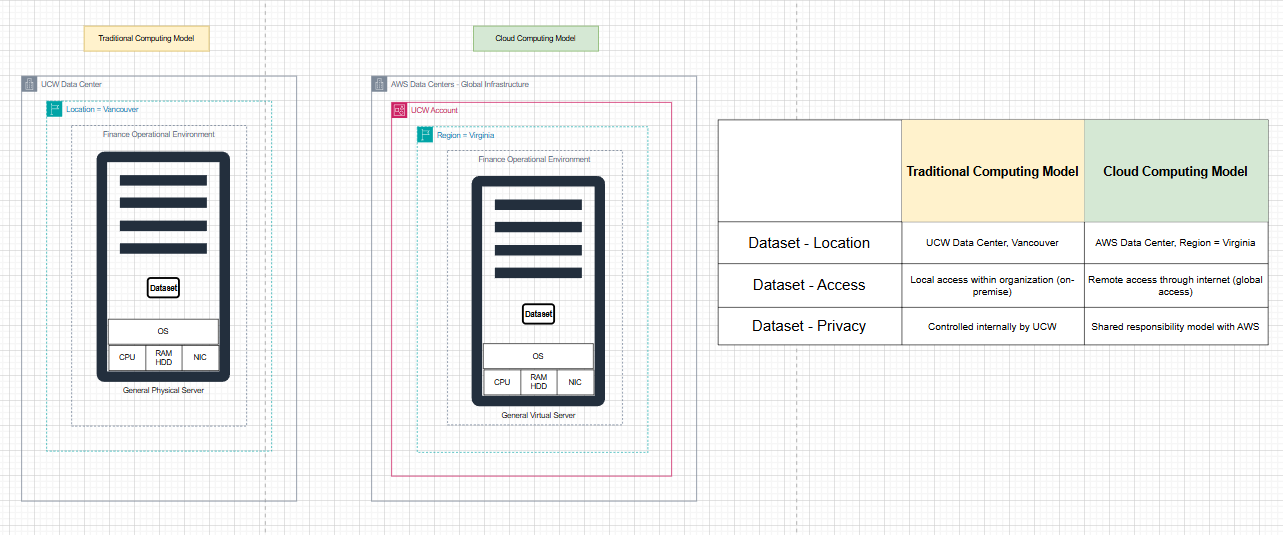
## **Deliverables:**

* Detailed weekly progress reports and screenshots (Weeks 2–4)
* Visual models for Fishbone, Datalake Design, and Cleaning Workflow
* Cost estimation documents for ingestion and profiling stages
* Working ETL prototype and documentation
* Summary report integrating all weekly activities into a cohesive project outcome

# 

# **AWS Deployment and Service Models**

Case Study #1: Traditional Computing Model vs Cloud Computing Model



A diagram of a cloud computing model

AI-generated content may be incorrect.

In the traditional computing model, UCW stores data locally in its Vancouver data center, with limited internal access and full control over privacy. In the cloud computing model, data is stored in an AWS data center (Virginia), accessible remotely via the internet, with privacy managed jointly by AWS and UCW under a shared responsibility model. Cloud offers more flexibility and scalability, while traditional systems provide full internal control.

Case Study #2: Cloud Deployment Models

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a computer

AI-generated content may be incorrect.

In a Public Cloud, data is stored in a provider's region (like AWS Virginia), accessed via the internet, and privacy is managed through a shared responsibility model. A Private Cloud keeps data within UCW or a dedicated environment, with restricted access and full privacy control by the organization. Hybrid Cloud combines both public and private setups, with shared access and privacy responsibilities. Multi-Cloud uses multiple cloud providers, requiring coordinated access and strong governance to ensure consistent privacy and compliance.

Case Study #3: Cloud Service Models

A screen shot of a computer

AI-generated content may be incorrect.

A chart with text and images

AI-generated content may be incorrect.

This table compares IaaS, PaaS, and SaaS based on data location, access, and privacy responsibilities. In IaaS, the Finance Operation Team provisions and controls all infrastructure in the AWS Virginia region, including storage and operating systems, and manages privacy through tools like IAM and encryption. In PaaS, AWS manages the infrastructure, while the Finance Operation Team handles application data and enforces app-level privacy. In SaaS, the AWS Operation Team manages the entire backend and data environment, with the Finance Operation Team only interacting with the application and ensuring compliant use of the data.

# **AWS Cost Analysis**

Case Study #4: Total Cost Of Ownership - Delaware North

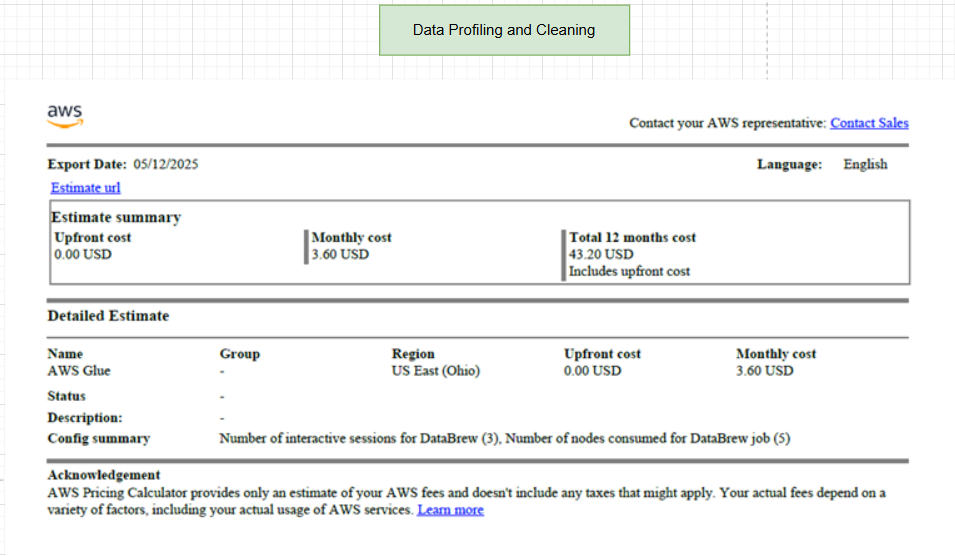
A screenshot of a computer screen

AI-generated content may be incorrect.

Case Study #5: AWS Pricing Calculator

A screenshot of a computer

AI-generated content may be incorrect.



A screenshot of a computer

AI-generated content may be incorrect.

Case Study #6: Support Plan

A close-up of a document

AI-generated content may be incorrect.

# **AWS Global infrastructure**

Case Study #7: AWS Global Infrastruture

A screenshot of a computer

AI-generated content may be incorrect.

A screenshot of a graph

AI-generated content may be incorrect.

# 

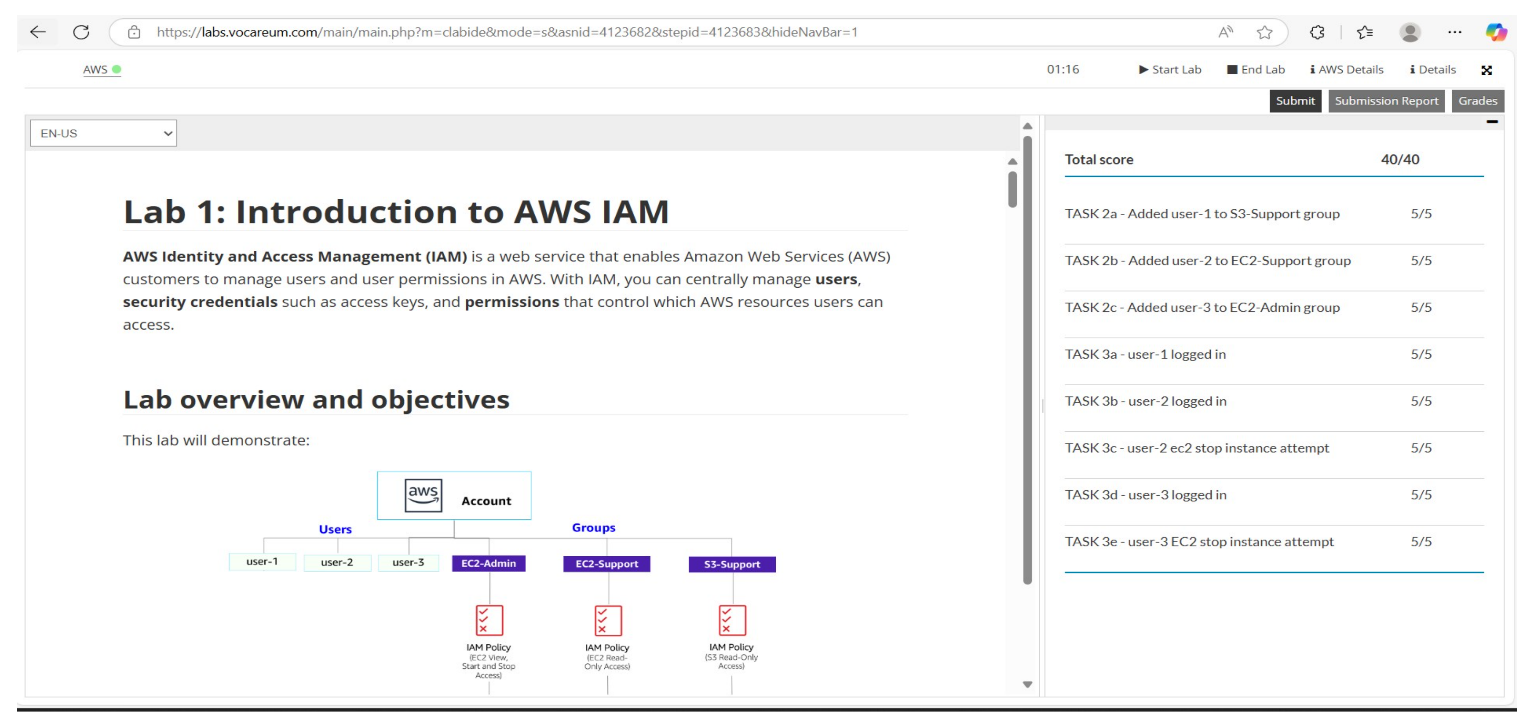
# **AWS IAM**

Case Study #8: Who is responsible

A screenshot of a computer

AI-generated content may be incorrect.

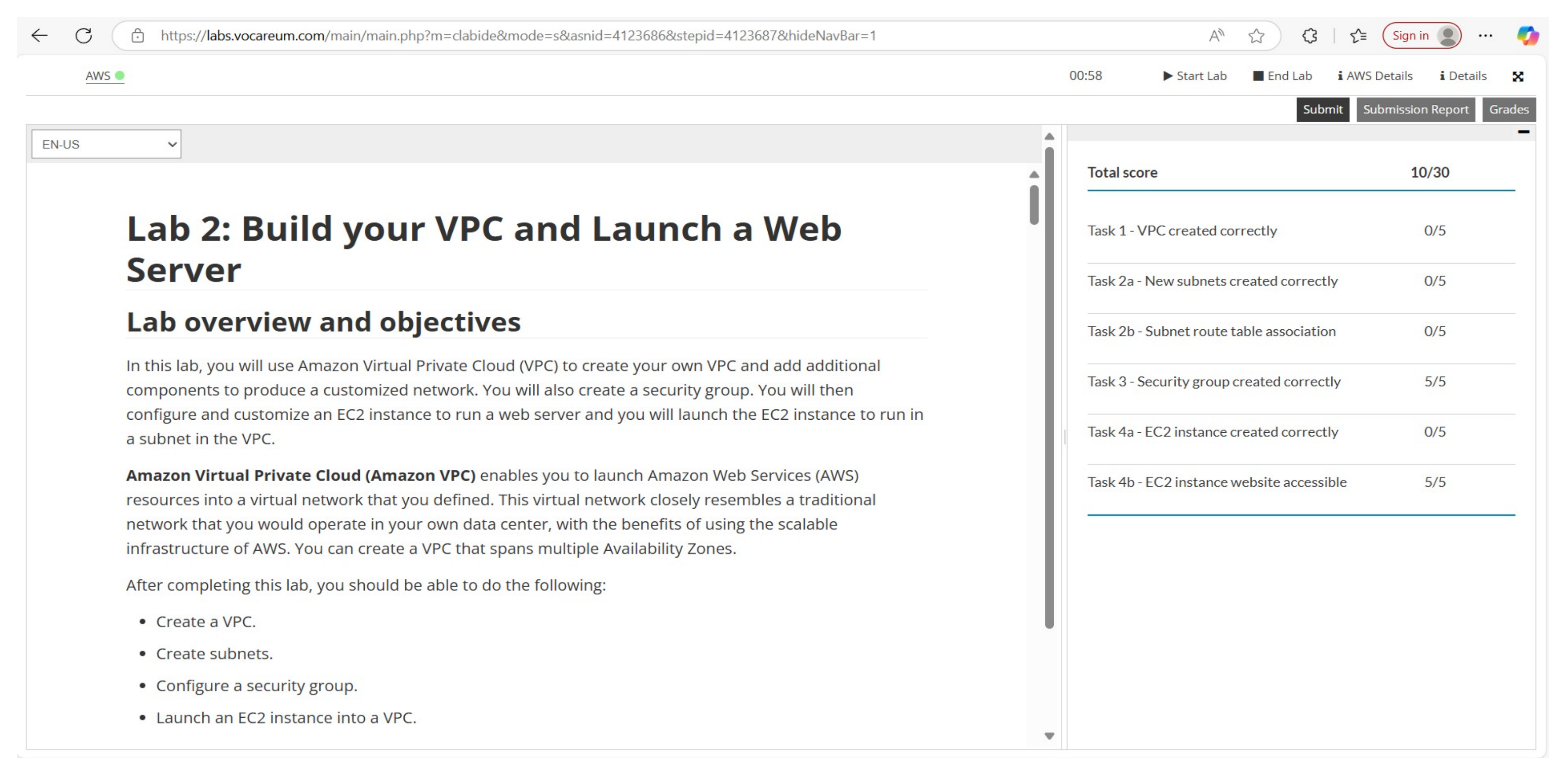
Case Study #9: IAM practrice: Lab 1



# 

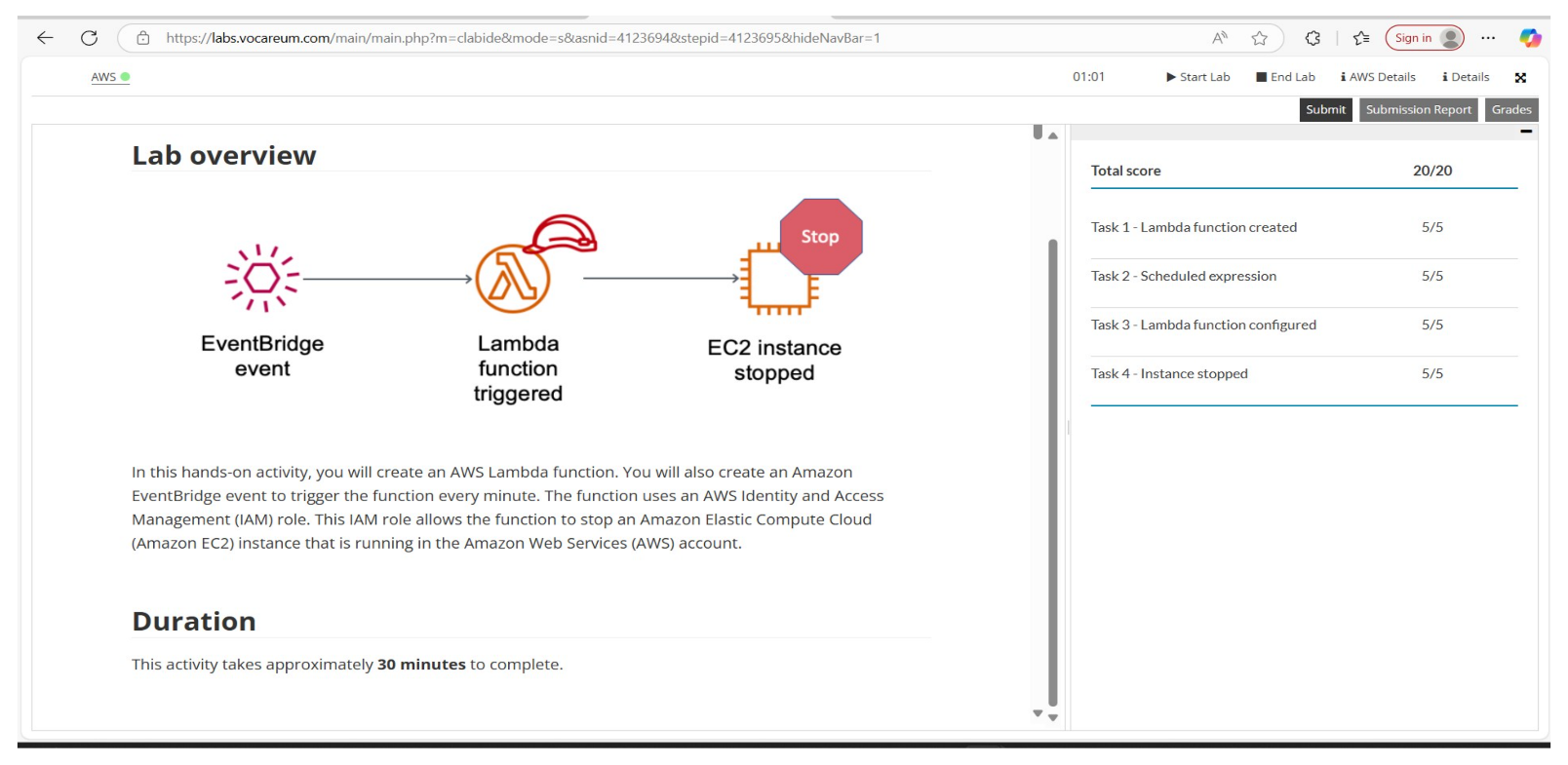
# **AWS VPC**

Case Study #10: Build your VPC: Lab 2



# **AWS Lambda**

Case Study #11: Create an AWS Lambda function



**AWS EBS**

Case Study #12: Working with Amazon EBS: Lab 4

