Project Design and Architecture

Tecnologías de Servicios para Ciencia de Datos

Carlos Mendoza Eggers & Joel Clemente López Cabrera
CreamCode

(WE ARE STILL WORKING ON THIS)

Contents

1 Introduction

In the last months, we have been immersed in the development of a robust and scalable architecture designed to handle complex data workflows. This project revolves around creating seamless integration between various services, each playing a pivotal role in data collection, storage, processing, and analysis. The overarching goal is to provide an efficient, graph-based API that enables users to query and manipulate complex relationships stored in a graph database.

The architecture is composed of several interconnected components. These include the Graph API, which serves as the primary interface for users; the Neo4j database, which manages graph storage and processing; the Datamart, responsible for aggregating and processing data; and the Datalake, which ensures long-term storage of both raw and processed data. Additionally, the Collector service was developed to ingest raw data from external sources and push it into the Datalake. Together, these components form the backbone of the system, designed to support highly interactive and data-driven workflows.

One of the most significant challenges of the project was deploying the entire system infrastructure on AWS. Building a cloud-native solution required extensive planning and implementation to ensure efficient communication between services, reliability, and scalability. From provisioning Virtual Private Clouds (VPCs) to configuring services, the focus was on creating a secure and high-performing environment for the application. Moreover, Terraform was used extensively to automate infrastructure provisioning, allowing reproducibility and minimizing manual configuration errors.

By leveraging a modular and containerized architecture and integrating technologies such as Spring Boot, Neo4j and Hazelcast, the system ensures that each service operates independently while seamlessly integrating into the larger ecosystem. These efforts culminated in a foundational system capable of meeting current requirements.

This document outlines the progress achieved, focusing on the technical milestones, the deployment strategies employed on AWS, and the next steps necessary to fully realize our vision for the ideal system. Through this effort, we have demonstrated a commitment to clean architecture, maintainability, and scalability, ensuring a solid foundation for long-term growth.

2 Original Idea

3 Services Overview

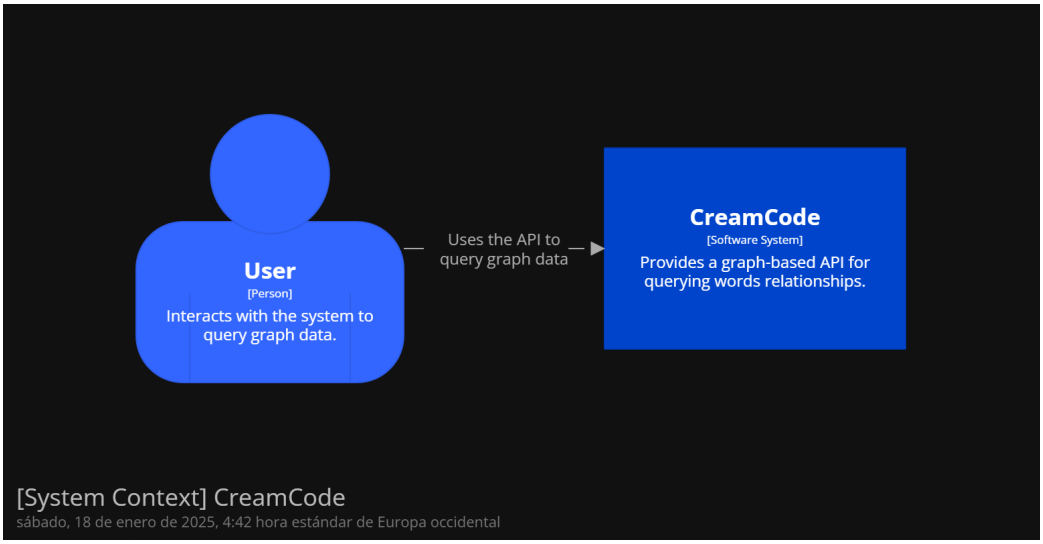

Figure 1: System Context Diagram

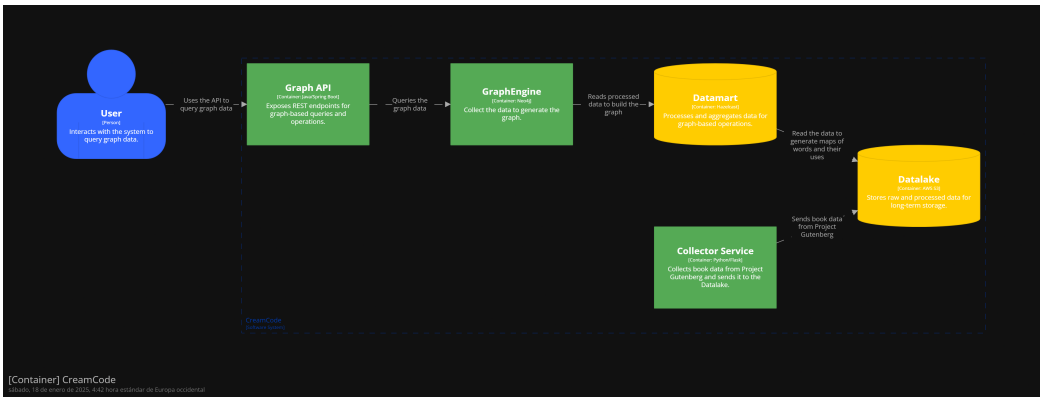

Figure 2: Container Diagram

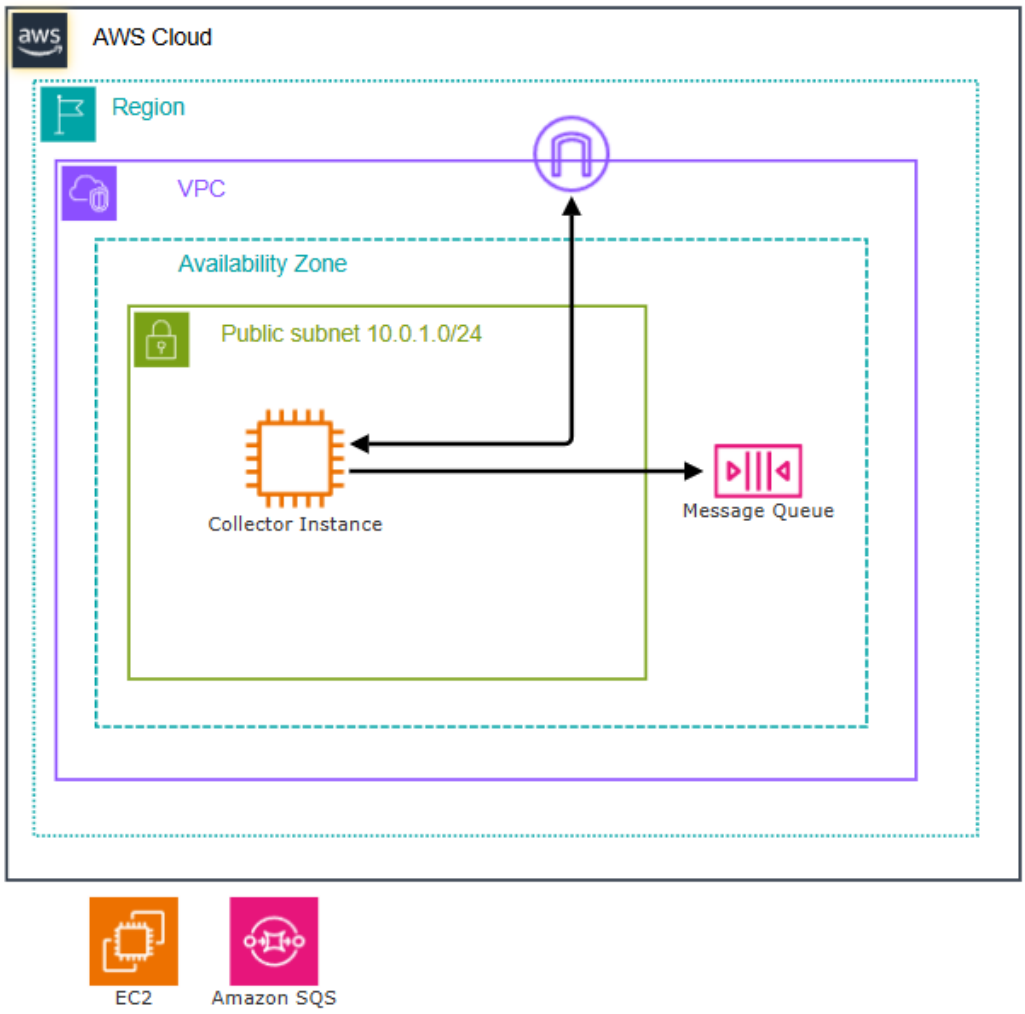

Figure 3: Collector

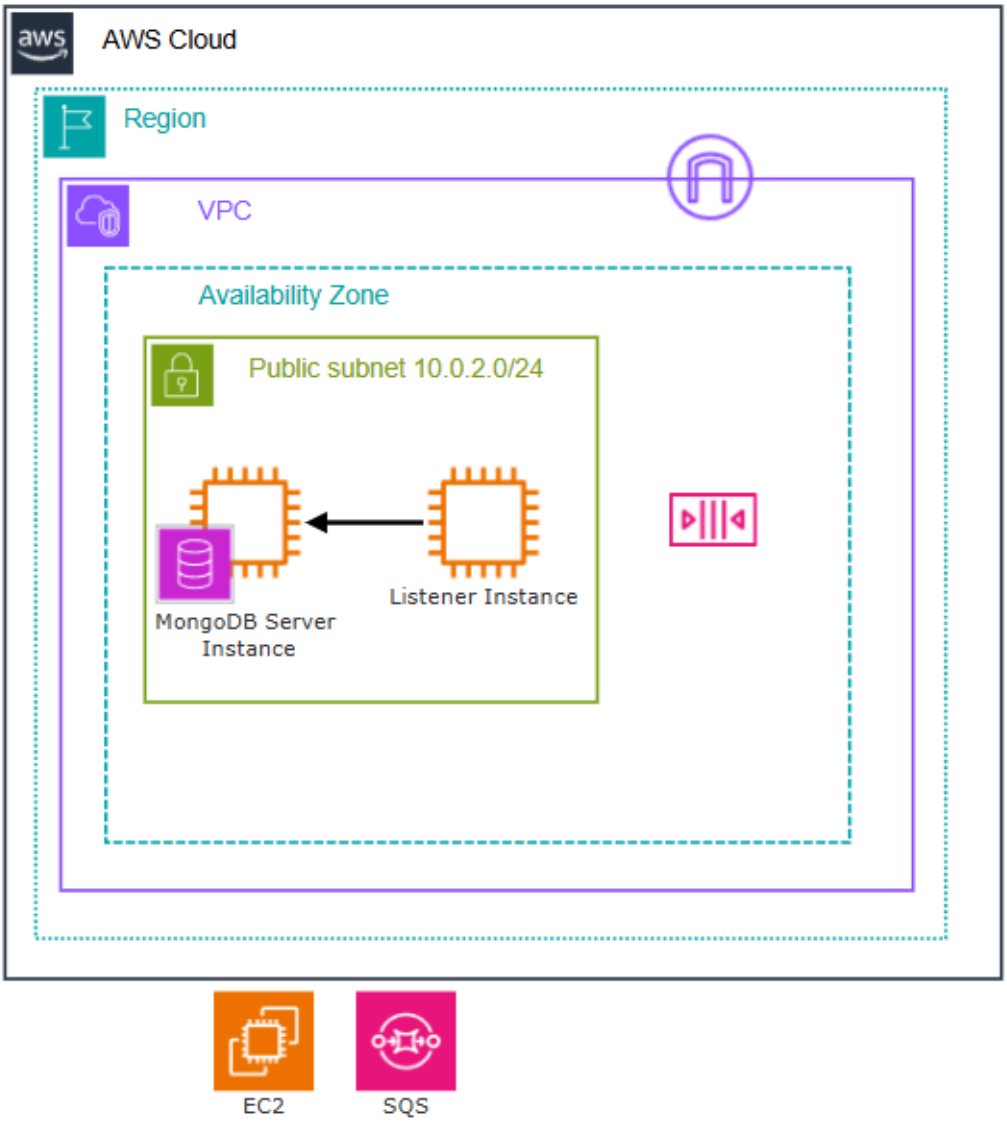

Figure 4: Datalake

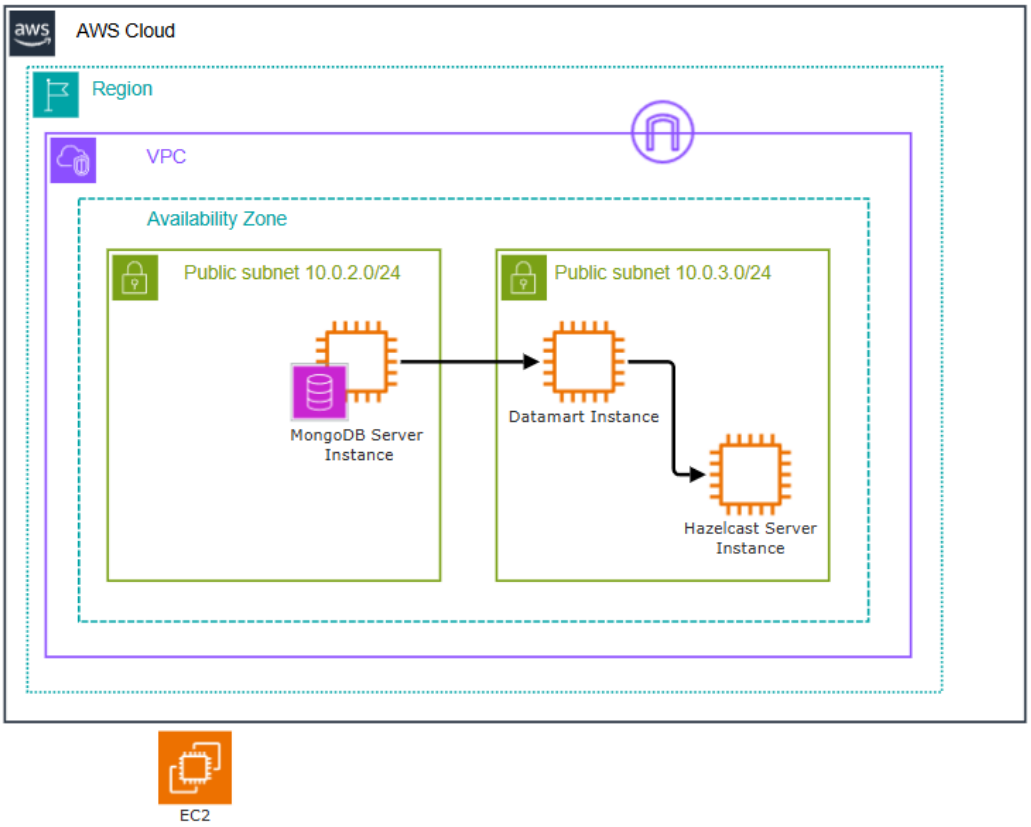

Figure 5: Datamart

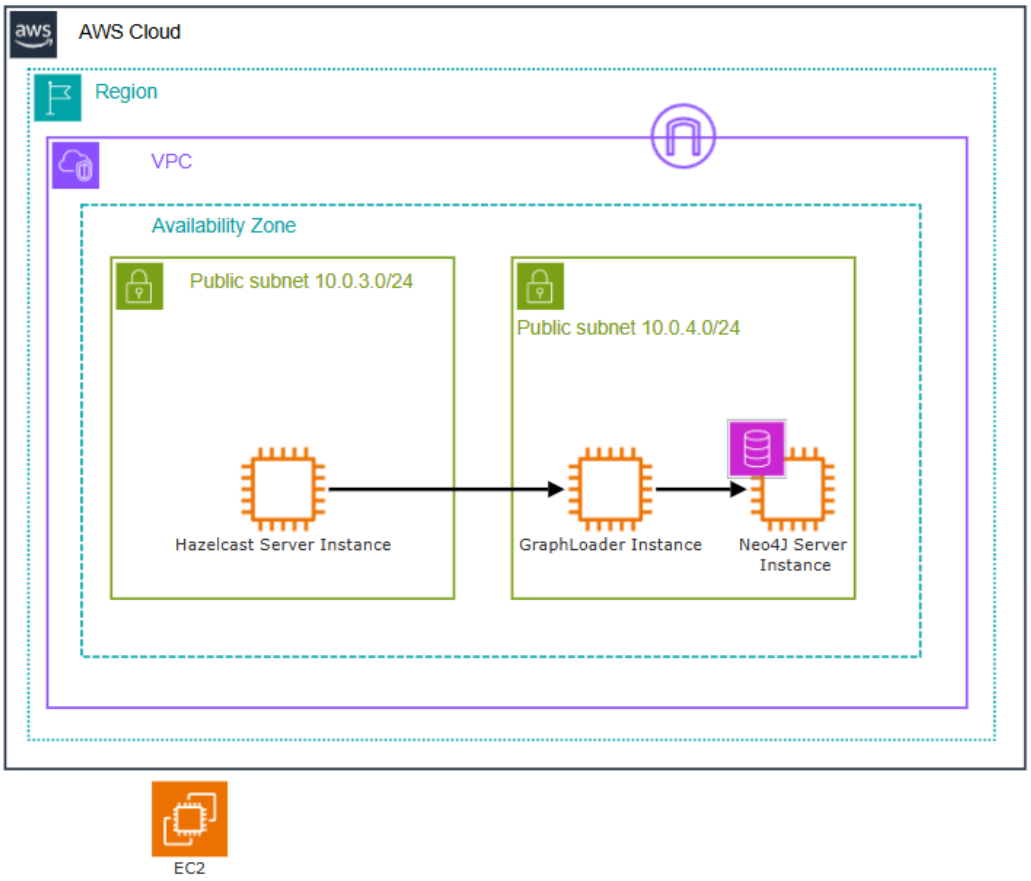

Figure 6: GraphEngine

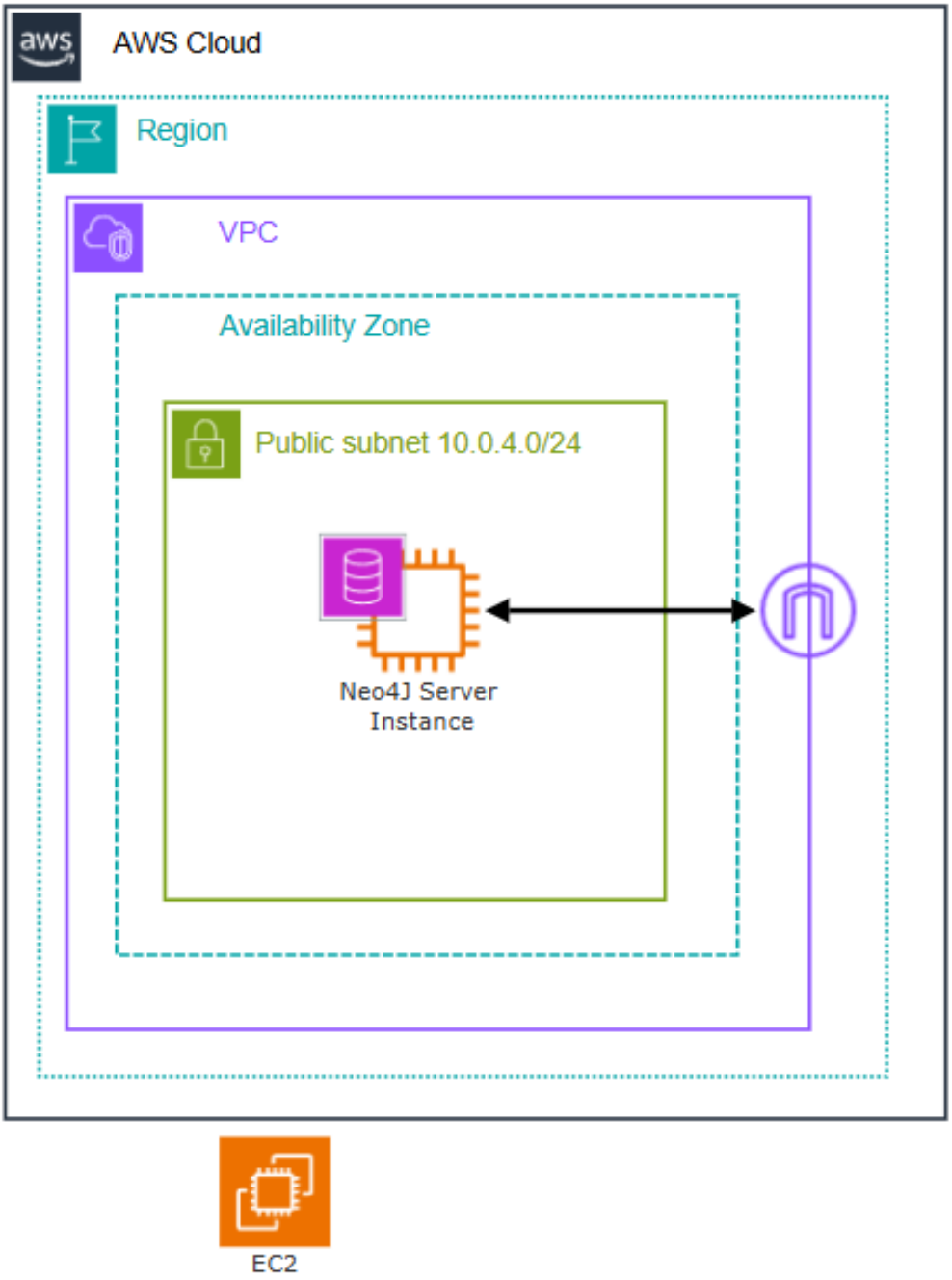
AWS Cloud
Region
VPC
Availability Zone
Public subnet 10.0.4.0/24
Neo4J Server
Instance
EC2

Figure 7: API

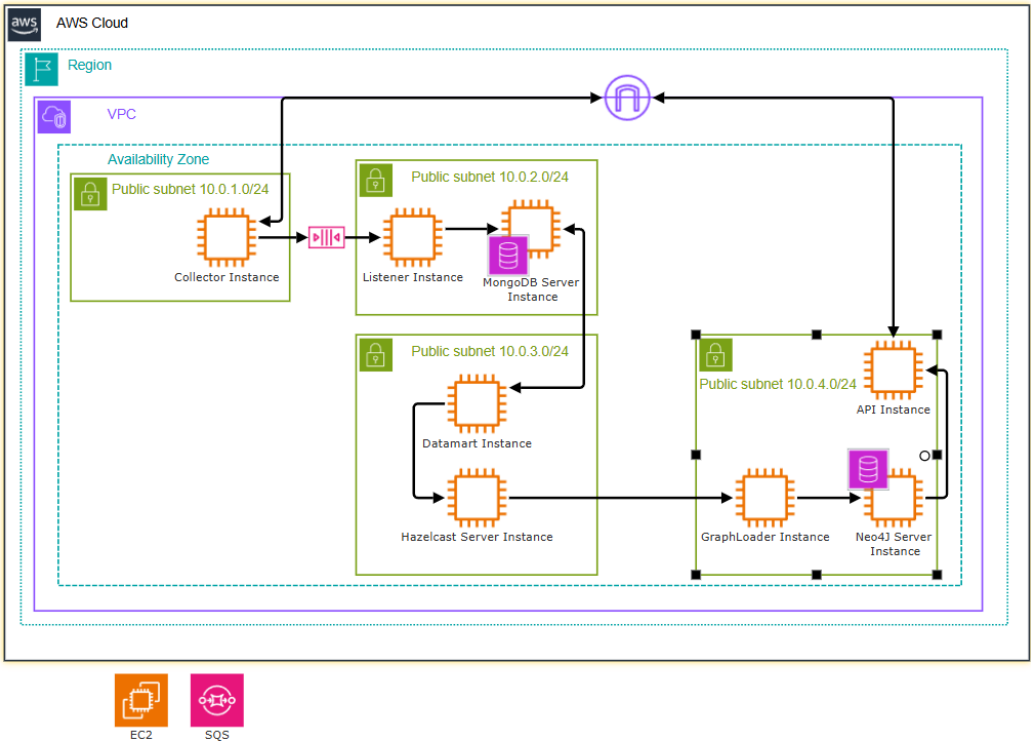

Figure 8: General Summary

4 Services Arquitecture

5 Future Ideas