

Infraestructura para la Computación de Altas

Prestaciones

Proyecto Final: AquaSenseCloud

Francisco Javier Mercader Martínez

Javier Moreno

Pablo Meseguer

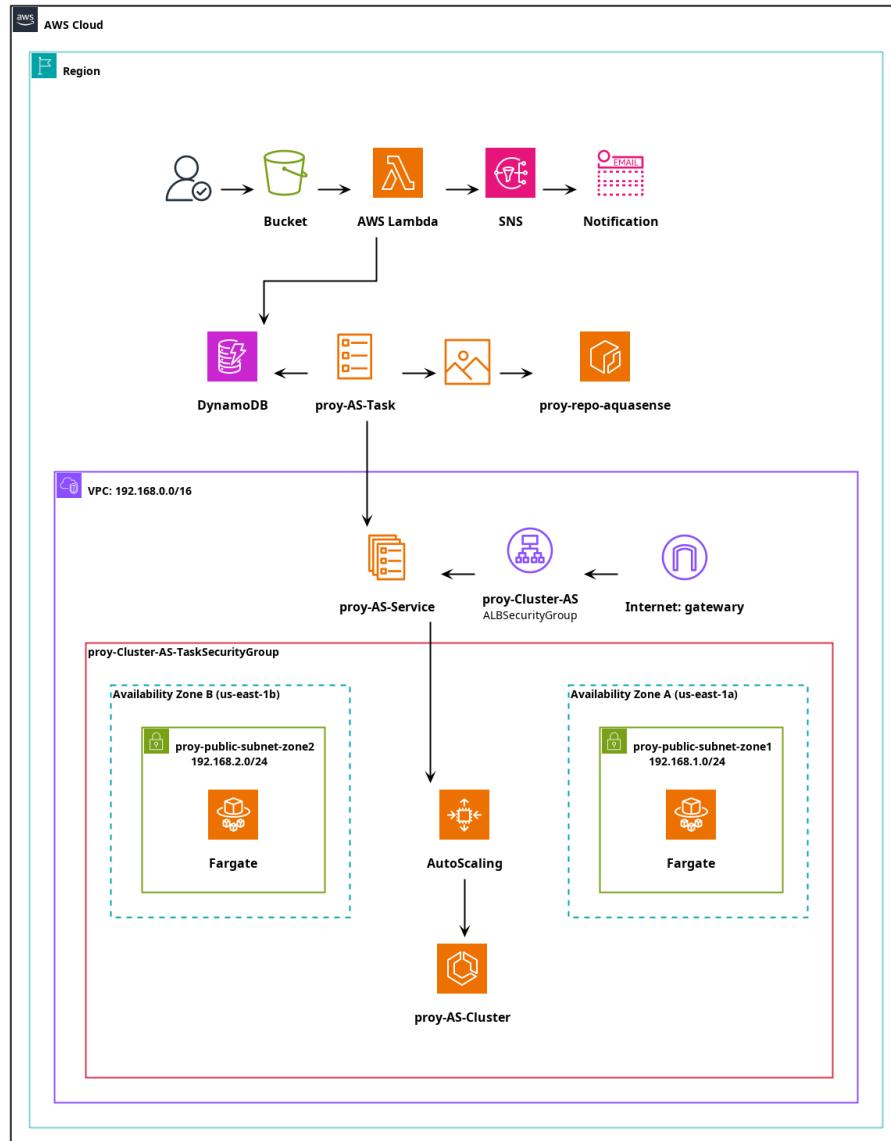
Curso: 2024/2025

Índice

1	Cronograma	2
2	Diagrama Arquitectura de la solución	3
3	Listado de Recursos y Servicios con su funcionalidad	3

1 Cronograma

2 Diagrama Arquitectura de la solución



3 Listado de Recursos y Servicios con su funcionalidad

1. Ingesta y almacenamiento crudo

- **Amazon S3 (AWS::S3::Bucket):** Almacena los archivos CSV de datos brutos con las medias y desviaciones típicas de temperaturas, se utilizan como entrada para el pipeline de datos. La ingestión de nuevos datos provoca el desencadenamiento de la función Lambda desarrollada, por lo tanto, nuestro bucket es asignado como desencadenador.

2. Pipeline de datos

- **AWS Lambda (AWS::Lambda::Function):** Ejecuta funciones que procesan los archivos almacenados en S3, transformando dichos datos y cargándolos en una tabla

DynamoDB. Esos datos almacenados en la tabla ya están preparados para ser consultados.

- **Amazon SNS (AWS::SNS::Topic):** Se utilizó para enviar notificaciones cuando la desviación estándar semanal de las temperaturas supera el umbral de 0.5. Esta modificación se publica en el tema SNS, informando a los suscriptores (en este caso los analistas) sobre la situación crítica en el monitoreo de temperaturas.

3. Almacenamiento de datos procesados/transformados

- **DynamoDB (AWS::DynamoDB::Table):** Almacena todos los datos transformados y procesados, listos para ser consultados, proporcionando así un acceso rápido y escalable a los datos.

4. Recursos para el desarrollo y pruebas iniciales

- **Instancia EC2 (AWS::EC2::Instance):** Utilizada para desarrollar, probar y ajustar el contenedor con la aplicación AquaSense. En esta instancia se configuró el entorno de Docker y se realizaron pruebas locales antes de enviar la imagen al repositorio de ECR.

5. Infraestructura de red y conectividad

- **VPC (AWS::EC2::VPC):** Proporciona un entorno de red aislado donde se despliegan los recursos de la infraestructura.
- **Subnets (AWS::EC2::Subnet):** Crea subredes públicas que permiten el acceso a Internet para los recursos dentro de la VPC.
- **Internet Gateway (AWS::EC2::InternetGateway):** Facilita la conexión de la VPC a Internet, permitiendo que los analistas accedan a los servicios configurados y desplegados.
- **Route Table (AWS::EC2::RouteTable):** Define las rutas de tráfico de red dentro de la VPC, asegurando que el tráfico fluya adecuadamente entre las subredes y el Internet Gateway.

6. Almacenamiento de imágenes

- **ECR (Amazon Elastic Container Registry):** Repositorio para almacenar y versionar la imagen del contenedor de AquaSense después de desarrollarla en EC2. Este recurso nos facilita la integración con ECS para el despliegue en producción de nuestra aplicación.

7. Recursos para el despliegue de la aplicación

- **ECSCluster (AWS::ECS::Cluster):** Agrupa, administra y organiza todas las tareas y servicios relacionados con la ejecución del contenedor de la aplicación. Actúa como punto central de administración para la ejecución de tareas, asegurando que estas puedan desplegarse de manera distribuida y eficiente.
- **ALB (AWS::ElasticLoadBalancingV2::LoadBalancer):** Balanceador de carga que gestiona/balancea el tráfico de los analistas hacia las tareas ECS. Redirige las solicitudes desde el puerto 80 (tráfico HTTP) al puerto 5000 del contenedor, además, mejora la disponibilidad y la tolerancia a fallos de nuestra aplicación.
- **ALBListener (AWS::ElasticLoadBalancingV2::Listener):** Configura reglas en el balanceador de carga para redirigir las solicitudes hacia el grupo de destino, escucha el tráfico de los analistas a través del puerto 80.
- **ECSTargetGroup (AWS::ElasticLoadBalancingV2::TargetGroup):** Asocia las tareas ECS con el balanceador de carga y configura verificaciones de estado mediante el endpoint `/health` que ha ido configurado en nuestra aplicación. Este permite conocer el estado en que se encuentran las tareas para que, en caso de que alguna falle, se marca la tarea como no saludable y eso informa al ECS Service de que hay que lanzar nuevas tareas para mantener su número deseado.

8. Definición y gestión de tareas

- **ECSTaskDefinition (AWS::ECS::TaskDefinition):** Este recurso describe las configuraciones necesarias para ejecutar contenedores en ECS. Especifica detalles como la imagen del contenedor a usar, la cantidad de CPU y memoria asignada, y la compatibilidad con el modo Fargate, que permite ejecutar contenedores sin gestionar servidores subyacentes. Además, define un mapeo de puertos, asignando el puerto anterior del contenedor (5000) al tráfico entrante. Es el punto central de la especificación para las tareas ECS.
Describe los contenedores que ejecutan la aplicación AquaSense, incluyendo la configuración de recursos (CPU, memoria) y el puerto utilizado (5000).
- **ECSService (AWS::ECS::Service):** Administra la ejecución de las tareas ECS, asegura la ejecución continua de un número deseado de tareas definidas en la configuración de tarea. En este caso, se garantiza que siempre haya dos tareas activas para manejar las solicitudes entrantes. Se utiliza el modo de lanzamiento Fargate para simplificar la gestión de la infraestructura. El servicio monitorea las tareas y las reinicia automáticamente si fallan, garantizando alta disponibilidad y resiliencia para nuestra aplicación.