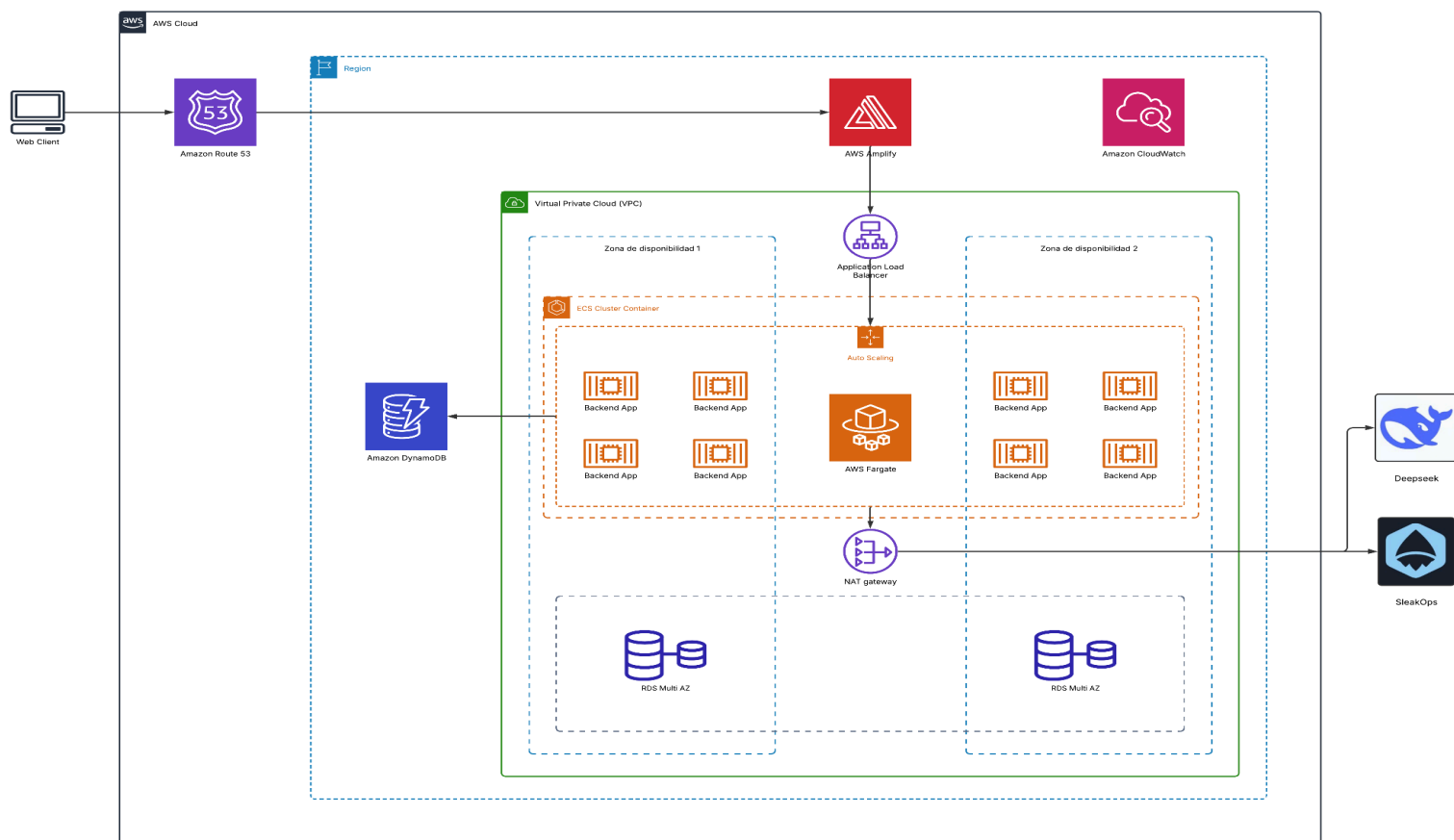


Diagrama de Red

Esta arquitectura busca cumplir con todos los requisitos técnicos solicitados para esta prueba, enfocándose en alta disponibilidad, escalabilidad, desacoplamiento y uso eficiente de los servicios distribuidos en AWS.



Descripción de la arquitectura

Cargas variables

La arquitectura incluye un **Application Load Balancer (ALB)** que distribuye el tráfico entrante entre múltiples instancias backend desplegadas con **AWS Fargate**. Además, un grupo de **Auto Scaling** permite escalar horizontalmente el número de contenedores en función de la demanda, garantizando así una respuesta dinámica ante picos o caídas de tráfico.

Alta disponibilidad

Los servicios backend están distribuidos entre **dos zonas de disponibilidad (AZs)**, lo que asegura la **redundancia ante fallos** en una zona. La base de datos relacional (Amazon RDS) está desplegada en modo **Multi-AZ**, lo que garantiza replicación automática ante cualquier problema.

Frontend en JavaScript

El frontend está construido con un framework JavaScript (como Astro) y es desplegado en **AWS Amplify**, lo que permite una integración continua (CI/CD), hosting globalmente distribuido y gestión automática del contenido estático.

⚙️ Backend con bases de datos relacional y no relacional

El backend corre dentro de contenedores **serverless en Fargate**. Este servicio accede a:

- **Amazon RDS** como base de datos relacional (por ejemplo, MySQL), ideal para datos estructurados.
- **Amazon DynamoDB** como base NoSQL, ideal para datos flexibles o con alta carga de lectura/escritura.

🔗 Consumo de 2 microservicios externos

El backend se comunica con dos microservicios externos, representados como **Deepseek** y **SleakOps** en el diagrama. Estos pueden representar servicios externos de autenticación, analítica o pagos. Para acceder a ellos de forma segura, los contenedores usan un **NAT Gateway** ubicado en la subred pública, permitiendo salida a Internet sin ser expuestos.

☁️ Servicios clave utilizados

Route 53: DNS global para enrutar tráfico.

AWS Amplify: Hosting y despliegue continuo del frontend.

ALB: Balanceador de carga para distribuir tráfico al backend.

Fargate + Auto Scaling: Contenedores backend sin gestión de infraestructura.

RDS Multi-AZ: Base de datos relacional con replicación.

DynamoDB: Base NoSQL distribuida y escalable.

CloudWatch: Monitoreo y logging centralizado.

NAT Gateway: Conectividad segura hacia servicios externos.