

Aplicación Web “Red Social Blog” en AWS

1. **Route53:** Permite el uso de múltiples regiones logrando alta disponibilidad de la aplicación. Redirige el tráfico de solicitudes de los clientes a la región con menor latencia y, en caso de que una región se vuelva inaccesible, permite tener disponibilidad en la región alternativa configurada.
2. **CloudFront:** Servicio de CDN (Content Delivery Network) para proporcionar una entrega rápida de contenido estático, en este caso archivos HTML, CSS y JS del frontend de la aplicación. CloudFront ayudará a mejorar la velocidad de carga y la escalabilidad.
3. **API Gateway:** Actúa como la puerta de enlace de la API para las solicitudes de los clientes. Al recibir una petición desde el frontend, encaminará esta petición hasta el Lambda, la cuál se activará y ejecutará el procesamiento correspondiente. API Gateway admite la autenticación y autorización de usuarios mediante Cognito, así como la integración con los microservicios externos (por ejemplo: PayPal y Twilio).
4. **S3 Bucket:** Almacena los archivos estáticos del frontend de la aplicación, como archivos HTML, CSS, JavaScript y archivos multimedia. Estos archivos se distribuyen a través de CloudFront para una entrega rápida y eficiente.
5. **Cognito:** Proporciona la funcionalidad de autenticación y autorización para los usuarios de la web app. Permite a los usuarios registrarse, iniciar sesión y gestionar sus credenciales de forma segura.
6. **AWS Lambda:** Se utiliza para ejecutar funciones que realizan diversas tareas en el backend. Interactúa con las bases de datos y los microservicios externos, como el servicio de pago Paypal y el servicio de notificaciones Twilio. Pueden ser escaladas automáticamente según la demanda y son altamente flexibles.
7. **AWS IAM:** se utiliza para definir políticas de acceso y asignar roles con permisos específicos a los usuarios y funciones de Lambda.
8. **Network Load Balancer:** Distribuye el tráfico de red entre las instancias de Fargate. Proporciona escalabilidad y alta disponibilidad al enrutar el tráfico de manera eficiente y distribuida.
9. **Fargate:** Se utiliza para ejecutar contenedores en AWS sin la necesidad de administrar la infraestructura subyacente. Los contenedores de Fargate se utilizan para alojar los microservicios y estos podrán escalar automáticamente según las necesidades de carga.
10. **Fargate CI/CD Stack:** Es una pila de servicios que se utiliza para implementar y gestionar la infraestructura de la aplicación utilizando prácticas de Integración Continua y Entrega Continua (CI/CD). Permite automatizar la compilación, prueba y despliegue de la aplicación.
11. **Cloud9:** Es un entorno de desarrollo integrado (IDE) basado en la nube que permite a los desarrolladores escribir, depurar y probar código directamente en el entorno de AWS. Proporciona un entorno colaborativo y flexible para el desarrollo de la aplicación.
12. **DynamoDB:** Es una base de datos NoSQL altamente escalable y totalmente administrada que se puede utilizar para almacenar datos no relacionales, como estadísticas, tokens de acceso, etc. Ofrece un rendimiento rápido y una escalabilidad automática según la carga de datos.
13. **RDS:** Es un servicio de base de datos relacional que proporciona disponibilidad, escalabilidad y capacidad de gestión. Se puede utilizar para almacenar por ejemplo publicaciones, información de usuarios, relaciones entre usuarios y publicaciones.

En resumen, esta arquitectura permite el desarrollo de una aplicación web de tipo "Blog" escalable, segura y de alto rendimiento en AWS. El frontend en JS se entrega a través de CloudFront y S3 Bucket, mientras que el backend se implementa utilizando API Gateway, AWS Lambda y Fargate para proporcionar la funcionalidad principal. Cognito asegura la autenticación y autorización de los usuarios, mientras que DynamoDB y RDS gestionan las bases de datos relacional y no relacional. Los servicios externos de PayPal y Twilio se integran a través del API Gateway. Route53 y la duplicación en dos regiones distintas garantizan la alta disponibilidad y la resiliencia del aplicativo web.