Nombre del Candidato:

GIOVANNY JAVIER PINARGOTE VILLAMARÍN

Cargo al que aplica:

INGENIERO CLOUD

Dir. de correo:

gj.85@hotmail.com

Telf:

0998189775 / 0989037548

RETO TÉCNICO- INGENIERO CLOUD

1.- ¿Cuál es la diferencia entre nube pública, privada e híbrida?

- **Nube pública:** Servicio que ofrecen varios proveedores externos (ejm.: Oracle, Microsoft, AWS y otros) en la cual sus recursos son compartidos y sus costos pueden ser mensuales o anuales. Ofrecen varios tipos de servicios que se encuentran físicamente fuera del rango que estamos laborando, su operatividad y administración es mucho más fácil, rápida y optimizada, ya que reduce costos y no requiere un mantenimiento directo por parte de los administradores.
- **Nube privada:** Es una infraestructura exclusiva y dedicada para la organización, ubicada físicamente ya sea dentro de la empresa o con un proveedor externo (con accesos restringidos). La nube privada ofrece mayor control, seguridad, y personalización para aquellas empresas/negocios con requisitos regulatorios y estrictos.
- **Nube híbrida:** Es la combinación de las anteriores, esta permite que las empresas o negocios, tengan sus recursos más críticos en la nube privada, mientras que pueden aprovechar la nube pública para trabajos menos sensibles o de alta demanda.

2. Describa tres prácticas de seguridad en la nube.

- **Gestión de accesos e identidades:** Se debe establecer controles estrictos sobre quién puede acceder a los recursos, la utilización de políticas es indispensable y debe ser basada en roles. La implementación de seguridades como la autenticación de doble factor, autenticación

multifactor (MFA), autenticación basada en tokens, autenticación basada en certificados y otros, reduce el riesgo de accesos no autorizados.

- **Cifrado de datos:** Los datos ya sea en tránsito como en reposo, deben estar cifrados asegurando que la información no sea interceptada o comprometida y no pueda ser leída ni utilizada sin las claves adecuadas.
- **Monitoreo y auditoría:** Se debe implementar herramientas de monitoreo de seguridad y registros de auditoría, ya que con esto permite detectar actividades sospechosas o inusuales en tiempo real. Esto ayuda a responder de manera inmediata ante posibles ataques o incidentes, cumpliendo con los requisitos y normativas.

3. ¿Qué es la IaC, y cuáles son sus principales beneficios?, mencione dos herramientas de la IaC y sus principales características.

Infraestructura como código (IaC): Es una práctica en la cual permite definir y gestionar la infraestructura de TI mediante archivos de configuración en lugar de realizar configuraciones manuales. Con este código se puede automatizar, versionar y replicar entornos de forma eficiente y segura.

Beneficios:

- **Automatización y consistencia.** - Se pueden evitar o reducir los errores al configurar entornos que se lo realiza manualmente y permite replicar en ambientes de desarrollo, test y producción de forma idéntica.
- **Escalabilidad y agilidad:** Se puede desplegar entornos completos en minutos, facilitando el trabajo de los equipos DevOps.
- **Versionamiento:** Al ser código, se puede mantener el control de las versiones, facilitando auditorías y retrocesos cuando sea necesario.

Herramientas:

- **Terraform:** Es una herramienta de código abierto que permite definir infraestructura en múltiples proveedores de nube mediante un lenguaje declarativo (HCL). Además facilita el ciclo de vida de la infraestructura (crear, modificar, eliminar)
- **Azure Resource Manager:** Son plantillas en formato JSON utilizadas para definir y desplegar recursos en AZURE, permiten declarar la infraestructura deseada y gestionarla como código. Son compatibles con los controles de versión, despliegue automatizado y validación previa a la ejecución.

4.- ¿Qué métricas consideran esenciales para el monitoreo de soluciones en la nube?

- **Recursos:** Uso del CPU y memoria el cual nos permita evaluar la carga del sistema y anticipar posibles cuellos de botella.
- **Disponibilidad:** Que nos permita medir el tiempo en que los servicios están activos y accesibles.
- **Latencia:** Esto no permite medir el tiempo que tarda una solicitud en recibir una respuesta.
- **Logs y Eventos de seguridad:** Nos ayuda a identificar posibles intentos de accesos no autorizados, cambios sospechosos en la infraestructura y cumplir con la normativa.

5.- ¿Qué es Docker y cuáles son sus componentes principales?

Docker: Es una plataforma de contenedores que permite desarrollar, empaquetar y ejecutar aplicaciones de forma aislada, ligera y portátil. Se utilizan para garantizar que el software se ejecute siempre de la misma manera, sin importar el entorno.

Componentes Principales

- **Docker Engine:** Es el motor que permite crear y ejecutar contenedores.
- **Docker Images:** Son plantillas inmutables que contienen todo lo necesario para ejecutar una aplicación.
- **Docker Containers:** Son instanciasen ejecución de una imagen, pueden ser livianos, portables y se ejecutan en entornos aislados.
- **DockerFile:** Es un archivo de texto con instrucciones que se utiliza para construir una imagen personalizada.
- **Docker Hub:** Es un registro público o privado, donde se almacenan y comparten imágenes.

6.- Caso Práctico

Cree un diseño de arquitectura para una aplicación nativa de nube considerando los siguientes componentes:

Frontend: Una aplicación web que los clientes utilizarán para navegar.

Backend: Servicios que se comunican con la base de datos y el frontend.

Base de Datos: Un sistema de gestión de base de datos que almacena información.

Almacenamiento de Objetos: Para gestionar imágenes y contenido estático.

Diseño:

Seleccione un proveedor de servicios de nube (AWS, Azure o GCP) y mantenga su selección

Diseña una arquitectura de nube. Incluye diagramas que representan la arquitectura y justifican sus decisiones de diseño (Utilice https://app.diagrams.net/)

He seleccionado Azure como proveedor ya que cuento con un sandbox para mis pruebas, a continuación describo un poco y adjunto la imagen:

1. Frontend.

Azure App Service (Web App): Para hospedar una aplicación web (React, Angular, Vue, etc.)

2. Backend.

Azure API Management: Para exponer y proteger los servicios backend.

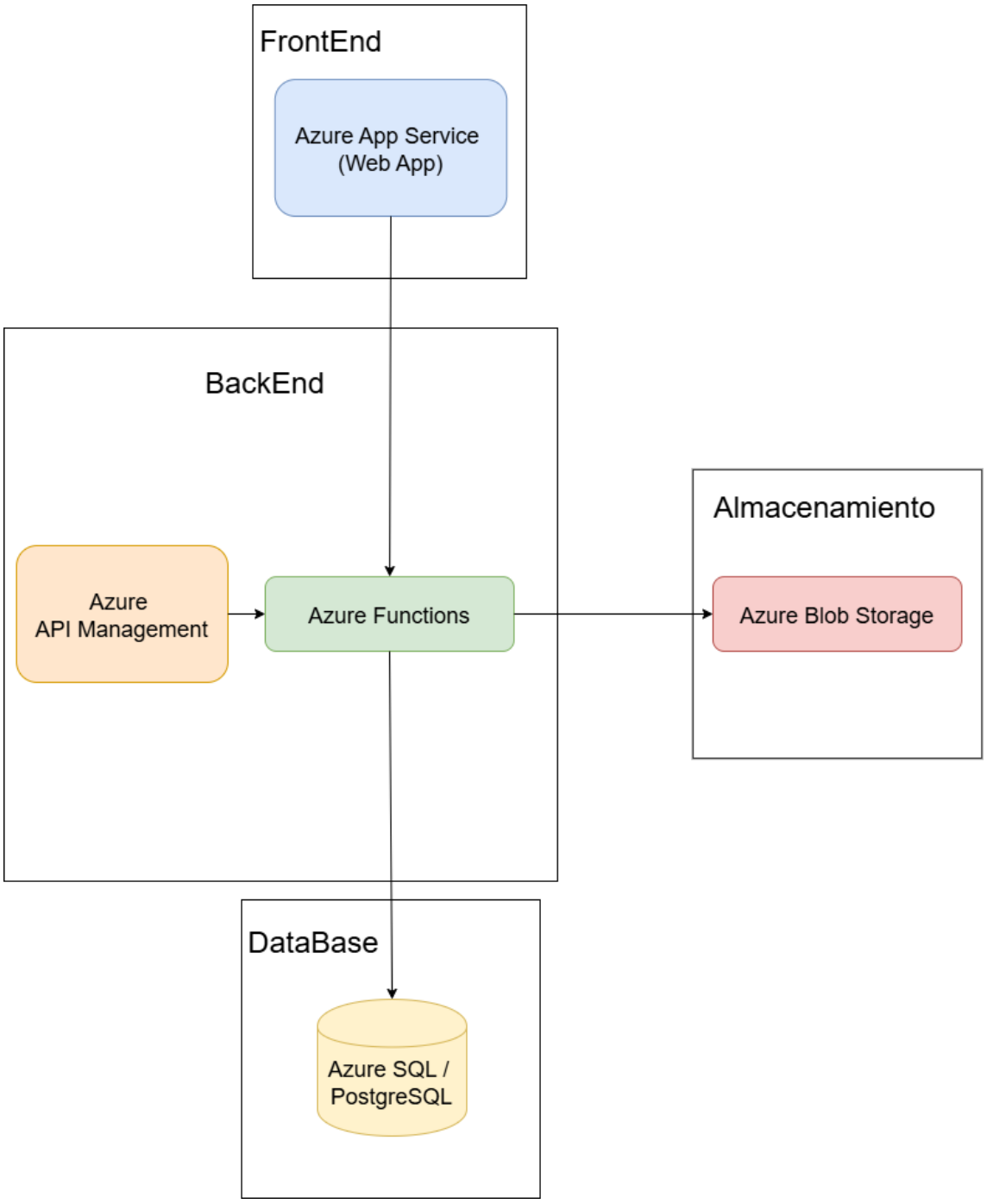
Azure Functions: Servicios serverless para manejar la lógica de negocio (escalabilidad automática y pago por uso).

3. Base de Datos.

Azure Database for PostgreSQL o Azure SQL Database: Solución de base de datos relacional totalmente administrada, escalable y con alta disponibilidad.

4. Almacenamiento de Objetos.

Azure Blob Storage: Para almacenar imágenes, documentos y contenido estático.



Justificación del diseño:

- Escalabilidad automática gracias a Azure Functions y App Service.
- Alta disponibilidad mediante servicios administrados.
- Seguridad reforzada con VNet, Key Vault y control de acceso basado en roles.
- Despliegue continuo con integración a GitHub Actions o Azure DevOps.