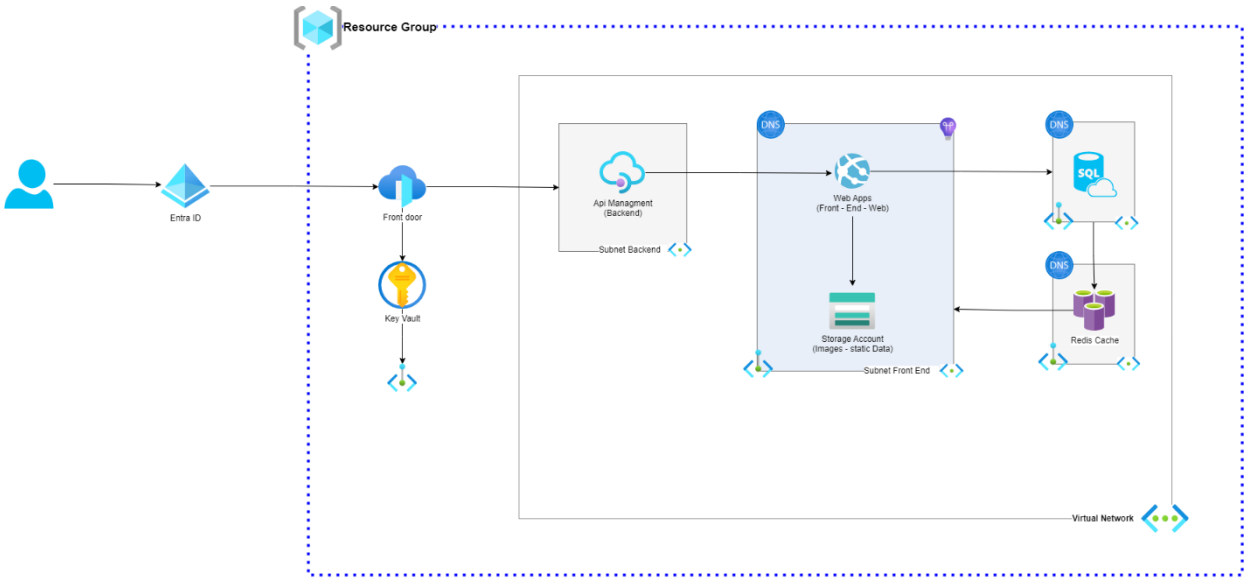
RESPUESTAS:

1. **Diferencias entre nube pública, privada e híbrida**:
    - **Nube pública**: Servicios ofrecidos por terceros a través de internet, accesi bles para cualquier individuo o empresa. Ejemplos: AWS, Google Cloud.
    - **Nube privada**: Infraestructura dedicada a una sola organización, ofreciend o mayor control y seguridad. Ejemplos: nubes internas de empresas grand es.
    - **Nube híbrida**: Combinación de nubes públicas y privadas, permitiendo m over datos y aplicaciones entre ellas según las necesidades.
2. **Prácticas de seguridad en la nube**:
    - **Control de acceso**: Implementar políticas de permisos y autenticación ro busta para asegurar que solo usuarios autorizados puedan acceder a los r ecursos.
    - **Cifrado de datos**: Proteger los datos tanto en tránsito como en reposo utili zando técnicas de cifrado avanzadas.
    - **Monitoreo y auditoría**: Supervisar constantemente el uso y el tráfico en la nube para detectar y responder rápidamente a cualquier actividad sospec hosa.
3. **Infraestructura como Código (IaC)**:
    - **Beneficios**: Facilita la automatización, mejora la consistencia, y permite un desarrollo más rápido y controlado.
    - **Herramientas de IaC**:
        - **Terraform**: Proporciona una manera declarativa de gestionar infrae structuras, compatible con múltiples proveedores de nube.
        - **Ansible**: Automate setup y configuración de sistemas, permitiendo la orquestación de múltiples servicios de manera eficiente.
4. **Métricas esenciales para el monitoreo de soluciones en la nube**:
    - **Latencia**: Tiempo de respuesta de los sistemas.
    - **Utilización de recursos**: Monitoreo de CPU, memoria y almacenamiento.
    - **Disponibilidad**: Tiempo de actividad y registros de fallos.
5. **Docker**:
    - **Descripción**: Docker es una plataforma de contenedorización que permite a los desarrolladores empaquetar aplicaciones y sus dependencias en co ntenedores portátiles.
    - **Componentes principales**:

- **Docker Engine**: Sistema que permite construir, ejecutar y gestionar con tenedores.
- **Docker Hub**: Repositorio en línea para almacenar y compartir imágenes de contenedores.

CASO PRACTICO



EXPLICACIÓN:

Usuario: Todo comienza con el usuario final que interactúa con la aplicación web desde su dispositivo.

Entra ID (Azure AD): El usuario se autentica mediante Azure Active Directory para asegurar que solo usuarios autorizados accedan a la aplicación.

Azure Front Door: Luego, el tráfico del usuario pasa por Azure Front Door, que distribuye las solicitudes a los servicios de backend más cercanos para mejorar el rendimiento y la latencia.

Azure Key Vault: Azure Front Door puede interactuar con Azure Key Vault para gestionar y proteger secretos, claves y certificados.

Azure API Management: El tráfico autenticado llega a Azure API Management, que actúa como una puerta de enlace para las APIs, proporcionando seguridad, monitoreo y políticas de tráfico.

Web Apps (Azure App Service): Desde API Management, el tráfico se dirige a las aplicaciones web alojadas en Azure App Service, en la subred del front-end. Estas aplicaciones manejan la lógica de presentación y se comunican con otros servicios de backend.

Azure Storage Account: Las aplicaciones web pueden acceder a Azure Blob Storage para recuperar y almacenar imágenes y contenido estático.

SQL Database: Las aplicaciones web también interactúan con Azure SQL Database para gestionar datos transaccionales, como usuarios, pedidos, etc.

Redis Cache: Azure Cache for Redis es usado por las aplicaciones web para mejorar el rendimiento mediante el almacenamiento en caché de datos frecuentemente accedidos.

Azure DNS: Servicios como SQL Database y Redis Cache están asociados a DNS para la resolución de nombres, facilitando el enrutamiento y la gestión.

Azure Monitor: Toda la infraestructura es supervisada por Azure Monitor, que proporciona observabilidad, alertas y análisis del rendimiento y estado de los servicios.