Gestión de datos

Índice

Contenido

1- Gestión y protección de datos

1.1- Contexto del proyecto

El proyecto consiste en un chatbot de IA que permite a distintos usuarios consultar información almacenada en una base de datos PostgreSQL desplegada en Render.

La IA trabaja directamente sobre las tablas del sistema para responder preguntas relacionadas con datos operativos, de recursos humanos y marketing.

Aunque el entorno actual es pequeño y de uso gratuito, la estrategia de gestión y protección de datos se define para un escenario productivo teórico, donde la pérdida, alteración o exposición de información tendría impacto operativo, legal y reputacional.

1.2- Estructura de datos y evaluación de la sensibilidad

A partir del esquema de la base de datos, se ha identificado la sensibilidad de cada conjunto de datos y su impacto en términos de seguridad.

1.2.1- Tabla users (Autenticación)

Campos:

- user_id
- role
- email
- password (hasheada)

Nivel de sensibilidad: Alta

Justificación:

- Contiene credenciales de acceso al sistema.
- El campo password, aunque esté hasheado, es crítico.
- La exposición de esta tabla permitiría accesos no autorizados.

Implicaciones de seguridad:

- Backups cifrados obligatoriamente.
- Acceso muy restringido a las copias.
- Restauraciones controladas y auditadas.

1.2.2- Tabla employees (Recursos Humanos)

Campos:

- employee_id
- first_name
- last_name
- email
- position
- department
- salary

Nivel de sensibilidad: Muy alta

Justificación:

- Contiene datos personales y económicos de empleados.
- El campo salary incrementa la criticidad.
- Datos protegidos por normativa de protección de datos (RGPD).

Implicaciones de seguridad:

- Backups cifrados en reposo.
- Retención estrictamente definida.
- No utilización de estos datos en entornos de prueba sin anonimización.

1.2.3- Tabla customers (Marketing / Comercial)

Campos:

- customer_id
- first_name_customer
- last_name_customer
- email
- region

Nivel de sensibilidad: Media

Justificación:

- Contiene datos personales básicos de clientes.
- No incluye información financiera directa.
- Riesgo moderado en caso de exposición.

Implicaciones de seguridad:

- Backups protegidos, aunque con menor criticidad que RRHH.
- Accesos controlados a las copias.

1.2.4- Tabla products (Marketing)

Campos:

- product_id
- product_name
- category
- unit_price

Nivel de sensibilidad: Baja

Justificación:

- Información comercial.
- No contiene datos personales.
- Bajo impacto en caso de pérdida o exposición.

Implicaciones de seguridad:

- Incluida en backups generales.
- No requiere controles adicionales específicos.

1.2.5- Tabla sales (Datos operativos y comerciales)

Campos:

- sale_id
- employee_id
- customer_id
- product_id
- sales_channel

- quantity
- discount_percentage
- payment_method
- subtotal
- discount_amount
- total
- sale_timestamp

Nivel de sensibilidad: Media-Alta

Justificación:

- Relaciona empleados, clientes y productos.
- Incluye información económica y de comportamiento comercial.
- Puede permitir inferencias sobre ingresos y rendimiento.

Implicaciones de seguridad:

- Backups protegidos y cifrados.
- Restauraciones cuidadosas para evitar inconsistencias.

1.3- Estrategia de backup

La estrategia de backups se diseña para proteger especialmente las tablas con datos personales y críticos, garantizando disponibilidad y recuperación ante incidentes.

1.3.1- Periodicidad

En un escenario productivo sobre Render con PostgreSQL gestionado:

- Backups automáticos diarios (cada 24 horas):
    - Realizados por el servicio gestionado de PostgreSQL.
    - Permiten recuperación a un punto en el tiempo ante errores recientes.
- Exportaciones automáticas de backups:
    - Se implementa un mecanismo automático para exportar la base de datos mediante tareas programadas (pg_dump).
    - Las exportaciones se almacenan en un sistema externo seguro, independiente de Render.
- Backups manuales bajo demanda:
    - Antes de cambios críticos:
        - Modificaciones del esquema
        - Cambios en la lógica del chatbot
        - Actualizaciones de roles y permisos
        - Cambios en relaciones entre tablas

1.3.2- Retención

- Backups automáticos internos:
    - Retención: 7 días
    - Enfocados a recuperación rápida.
- Exportaciones automáticas externas:
    - Retención: 30 días
    - Enfocadas a incidentes graves, análisis forense o detección tardía de errores.

Esta combinación reduce la dependencia de un único proveedor y mejora la resiliencia del sistema.

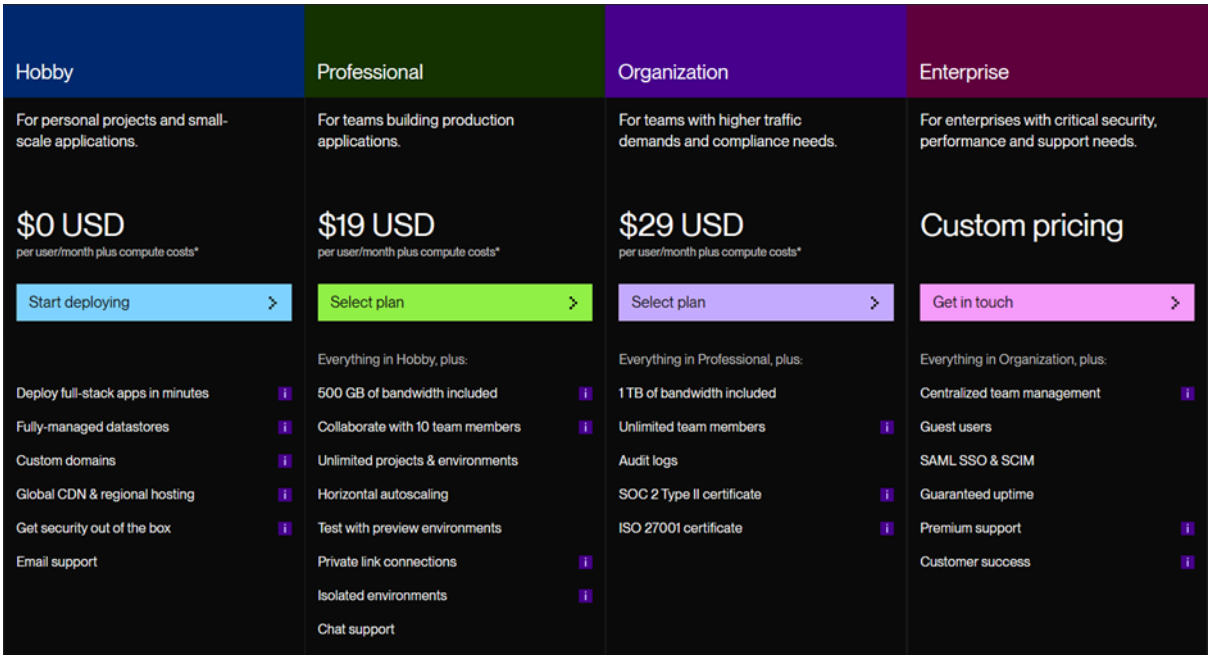
1.4- Costes de la estrategia de backup (estimación)

1.4.1- Supuestos

- Base de datos PostgreSQL desplegada en Render, de tamaño reducido a medio (5–10 GB).
- Backups automáticos diarios gestionados por Render.
- Exportaciones automáticas externas con retención de 30 días.
- API del chatbot desplegada en AWS.
- La API no almacena datos persistentes críticos, pero genera logs y configuraciones que deben considerarse.

1.4.2- Estimación de costes

- **Tabla de precios de Render:**



- **Backups automáticos en Render:**

  - Incluidos en el coste del plan de base de datos.
  - Sin coste adicional específico.

- **Exportaciones automáticas externas:**

  - Coste aproximado de almacenamiento: 0,02 € – 0,05 € por GB/mes
  - Para 10 GB con retención de 30 días: Coste inferior a 1 €/mes

- **Backups y costes asociados a la API en AWS**

La API del chatbot, desplegada en AWS, no mantiene datos persistentes críticos. No obstante, se consideran los siguientes elementos dentro de la estrategia de backup:

- Código y configuración: Gestionados mediante sistemas de control de versiones, sin necesidad de almacenamiento adicional para backups.

- Logs de aplicación y auditoría: Almacenados en servicios de monitorización de AWS para trazabilidad y detección de incidentes.

- Coste estimado de almacenamiento de logs: 1 € – 3 €/ mes, en escenarios de bajo volumen.

No se requieren backups de datos estructurados adicionales en AWS, ya que la información crítica reside en la base de datos en Render.

Conclusión:

El coste total de la estrategia de backup es bajo y controlado.

Los costes principales se concentran en el almacenamiento externo de copias de seguridad de la base de datos, mientras que el impacto económico de los backups relacionados con la API en AWS es mínimo.

Esta estrategia ofrece una protección adecuada de los datos sensibles del sistema sin introducir costes elevados ni complejidad innecesaria.

1.5- Controles de seguridad aplicados a los backups

- Cifrado de las copias de seguridad.
- Acceso restringido a backups según rol.
- Separación de entornos.
- Procedimientos documentados de restauración.
- Posibilidad de pruebas periódicas de recuperación.

1.6- Riesgos y mitigación

Riesgos:

- Pérdida total de la base de datos.
- Acceso no autorizado a información sensible.
- Corrupción de datos comerciales o de RRHH.
- Dependencia excesiva del proveedor.

Mitigación:

- Backups automáticos diarios.
- Exportaciones externas independientes.
- Retención diferenciada según criticidad de las tablas.
- Estrategia de restauración definida.

2- Conclusión

La estrategia propuesta protege de forma prioritaria las tablas que contienen datos personales, credenciales e información económica, garantizando la disponibilidad y seguridad de la información utilizada por el chatbot de IA.

El diseño permite escalar el sistema a un entorno productivo manteniendo un equilibrio adecuado entre seguridad, coste y complejidad, alineado con buenas prácticas de ciberseguridad.