

¿Por qué elegimos PostgreSQL?

Porque es gratuito, y es el tema principal del curso, de código abierto, y tiene una comunidad muy activa que resuelve dudas.

Estándares de diseño

Seguimos las reglas de normalización para evitar datos repetidos, como separamos *Usuario* y *Departamento* en tablas distintas en lugar de mezclarlas, usamos nombres claros para tablas y definimos claves foráneas para conectar todo.

Entidades más importantes

Las tablas *Usuario*, *FlujoAprobacion*, y *Solicitud* son clave. Los usuarios son el centro del sistema, los flujos definen cómo se aprueban las cosas, y las solicitudes son el "corazón" del negocio.

Normalización

Aplicamos hasta la tercera forma normal, en lugar de guardar el nombre del departamento en cada usuario, usamos *departamento_id* para vincularlo a su tabla.

Restricciones e integridad

Usamos *CHECK* para validar valores, *NOT NULL* en campos obligatorios, y *FOREIGN KEY* para asegurar que los IDs enlacen correctamente.

Cambios en la estructura

Creamos una versión inicial, la cual tuvo que cambiar después de la primera presentación de la base de datos, donde se tuvieron que agregar tablas nuevas para el manejo de solicitudes sencillas, grupos de aprobadores entre otras tablas que nos ayudaron a mejorar la base de datos de nuestro sistema BPM.

Datos de prueba

Creamos usuarios ficticios de distintos departamentos, flujos de aprobación con pasos realistas, y solicitudes en estados variados.

Mi contribución y organización

Yo con otro compañero nos enfocamos principalmente en implementar los cambios sugeridos durante la primera presentación de la base de datos.

Equidad en el trabajo

Considero que para esta entrega el nivel de trabajo no fue igual para todos, pudimos mejorar, pero al final considero que el aporte de cada persona fue clave para lograr obtener el trabajo final.

¿Qué cambiaría?

probaría más casos de error, como eliminar un flujo que está en uso, para ver cómo responde la base de datos.