

Reflexión Estuardo Castro

Proyecto 3

1. ¿Qué criterios usaron para decidir qué entidades y relaciones debían formar parte del modelo?

Para definir nuestras entidades y relaciones, analizamos los componentes esenciales de la gestión de eventos culturales. Identificamos los actores clave que vendrían a ser los usuarios, artistas o patrocinadores. También notamos los objetos fundamentales que son las sedes y recursos. Por último, se tomaron en cuenta los procesos principales, que son las inscripciones, actividades y patrocinios.

Priorizamos entidades que permitieran un seguimiento completo del ciclo de vida de un evento, desde su planificación hasta la evaluación posterior. Decidimos omitir ciertos aspectos, como la gestión interna de las sedes o los detalles financieros específicos, para mantener el modelo centrado en su funcionalidad principal.

2. ¿Qué tan adecuadas fueron las claves primarias y foráneas que definieron en su diseño?

Utilizamos principalmente claves primarias autoincrementales con el primal serial key para la mayoría de las tablas, lo que facilitó la generación de identificadores únicos y simplificó las relaciones entre tablas.

Las claves foráneas se definieron siguiendo principios de integridad referencial, permitiendo mantener la consistencia de los datos y modelar correctamente las dependencias entre entidades. Un ejemplo para esto, podría ser la relación entre eventos y sedes, o también entre inscripciones y usuarios.

En tablas de relación como actividad_artista, implementamos claves compuestas que reflejan la naturaleza N de estas relaciones, optimizando tanto el almacenamiento como las consultas.

3. ¿En qué medida aplicaron la normalización? ¿Qué beneficios y limitaciones experimentaron?

A nuestras tablas les aplicamos una normalización que llegaría hasta la 3FN, lo que nos llegaría a permitir el poder:

- Eliminar redundancias, especialmente en información de contacto y descripciones
- Facilitar actualizaciones sin anomalías

- Mantener la integridad y consistencia de los datos

Aun así, en algunos casos tuvimos que balancear entre normalización y rendimiento. Esto se puede ver que en la tabla de eventos, mantuvimos algunos campos derivables como podría ser el estado, que podrían calcularse, pero los incluimos más que todo para facilitar consultas frecuentes sobre eventos activos o finalizados.

4. ¿Qué restricciones y reglas del negocio implementaron directamente en la base de datos y por qué?

Le implementamos bastantes restricciones para que las tablas sean más ordenadas y tengan una mejor integridad. Esto lo podemos demostrar en que nuestra base poseía todos estos factores:

- Tipos ENUM para campos con valores predefinidos para darle un mejor aspecto y limpieza a los datos (o al menos así lo veo)
- Restricciones CHECK para validar datos numéricos
- Triggers para mantener la coherencia de datos relacionados
- Restricciones de unicidad en campos clave como nombres de eventos o correos electrónicos para evitar repeticiones que provocarían problemas a futuro

Decidimos implementar estas reglas en la base de datos y no solo en la aplicación para garantizar que cualquier sistema que acceda a los datos respete estas restricciones fundamentales.

5. ¿Qué ventajas o desventajas identificas del modelo que construyeron al momento de hacer consultas complejas?

Ventajas:

- La estructura normalizada facilita consultas con múltiples joins manteniendo la integridad
- Los tipos ENUM y campos de estado permiten filtros eficientes
- Las relaciones bien definidas permiten navegar fácilmente entre entidades relacionadas

Desventajas:

- Consultas que requieren datos de muchas tablas (como reportes de asistencia completos) necesitan múltiples joins

- Las consultas que combinan datos de eventos, actividades, inscripciones y asistencias pueden volverse complejas
- La separación entre eventos y actividades, aunque conceptualmente correcta, añade complejidad a algunas consultas

6. ¿Qué cambiarían en el diseño de la base de datos si tuvieran que escalar este sistema a un entorno de producción?

Para un entorno de producción, consideraríamos:

1. Añadir índices adicionales para optimizar consultas frecuentes, especialmente en fechas y estados
2. Implementar particionamiento por fechas en tablas grandes como inscripciones y asistencias
3. Evaluar la posibilidad de desnormalizar estratégicamente algunas tablas para mejorar el rendimiento de consultas frecuentes
4. Implementar vistas materializadas para reportes comunes
5. Añadir campos para auditoría (created_by, updated_by, etc.)
6. Mejorar el manejo de datos históricos con estrategias de archivado
7. Incorporar manejo de transacciones más robusto para operaciones críticas