



# SISTEMA DE RESERVA Y GESTION DE EVENTOS

**SQL**

**Freddy Saldivia**

**Tutor: Gonzalo Satina Naggi**

**Comisión: 53170**

## INDICE

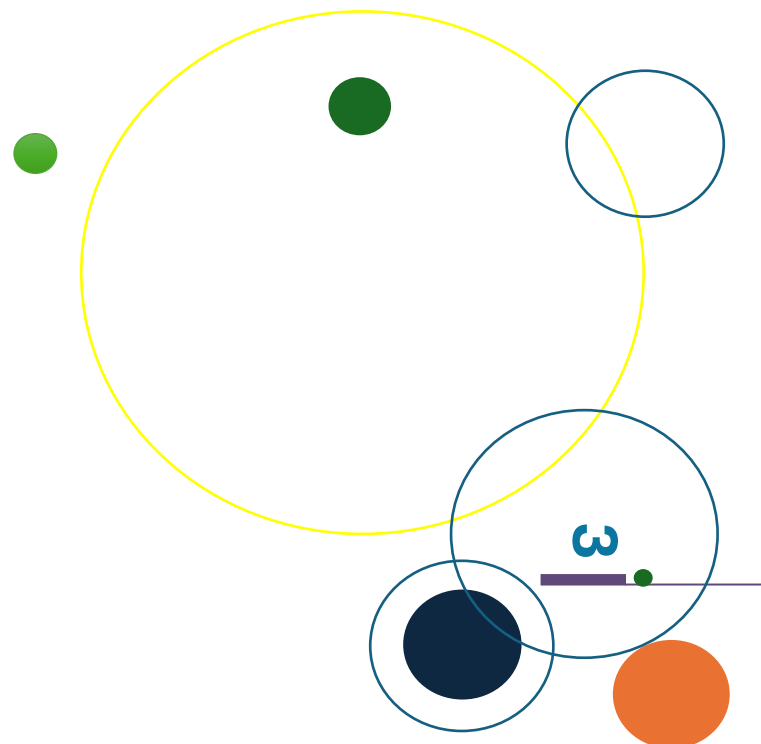
INDICE	PAG. 2
INTRODUCCION	PAG. 3
OBJETIVOS	PAG. 4
DIAGRAMA DE ENTIDAD Y RELACION	PAG. 5
LISTADO DE TABLAS	PAG. 6 al 8
VISTAS	PAG. 8 y 9
FUNCIONES	PAG. 8 y 9
PROCEDIMIENTOS ALMACENADOS	PAG. 9 y 10
TRIGGERS	PAG. 10 y 11
HERRAMIENTAS UTILIZADAS	PAG. 11
REPORTE	PAG. 12 y 14
a) Estado de los asientos (disponible vs ocupados).	
b) Reservas por lugar o sitio.	
c) Relacion entre evento y estado de reserva del evento.	
DCL	PAG. 15 y 16
TCL	PAG. 16 y 17
BACKUP	PAG. 17
ESCALABILIDAD	PAG. 17 y 18
CONCLUSION	PAG. 18



## INTRODUCCION

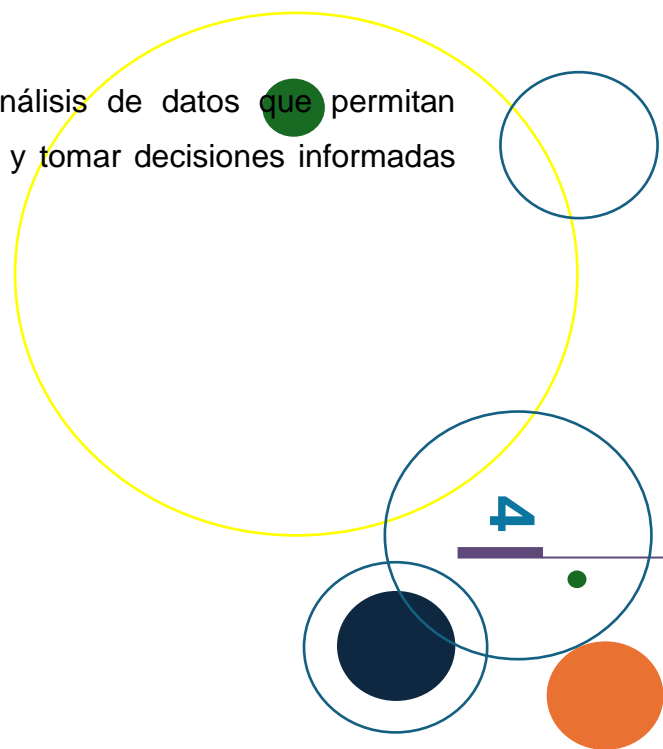
La gestión eficaz de eventos es crucial para el éxito de cualquier reunión, conferencia, o exposición. Sin embargo, los sistemas de gestión de eventos se enfrentan a una serie de desafíos en la actualidad, que van desde la integración de datos hasta la seguridad y la experiencia del usuario.

En esta introducción, exploraremos los principales problemas que enfrentan los sistemas de gestión de eventos y cómo los objetivos de la base de datos pueden ayudar a mitigarlos. Al establecer metas específicas, como integración de datos, personalización, seguridad y accesibilidad, podemos avanzar hacia soluciones más efectivas que satisfagan las necesidades de organizadores y asistentes por igual.



## OBJETIVOS

- **Integración de datos:**
  - Objetivo: Diseñar una base de datos que permita la integración de datos de múltiples fuentes.
- **Personalización:**
  - Objetivo: Desarrollar algoritmos de personalización que utilicen datos de la base para adaptar la experiencia del usuario según sus preferencias.
- **Seguridad y privacidad de datos:**
  - Objetivo: Implementar medidas de seguridad robustas para proteger los datos del evento y de los participantes.
- **Experiencia del usuario:**
  - Objetivo: Mejorar la experiencia del usuario mediante una interfaz intuitiva y funciones móviles.
- **Accesibilidad:**
  - Objetivo: Garantizar que la plataforma sea accesible para personas con discapacidades mediante el cumplimiento de estándares de accesibilidad web.
- **Gestión de cambios y actualizaciones:**
  - Objetivo: Facilitar la implementación de cambios y actualizaciones en la base de datos para adaptarse a las necesidades cambiantes del evento.
- **Análisis de datos y Retorno de inversión:**
  - Objetivo: Proporcionar herramientas de análisis de datos que permitan evaluar el Retorno de inversión del evento y tomar decisiones informadas para futuros eventos.



## DIAGRAMA ENTIDAD Y RELACION



## LISTADOS DE TABLAS

A continuación, se presentarán las tablas que componen la base de datos:

**“Cliente”:** Esta tabla contiene información del cliente.

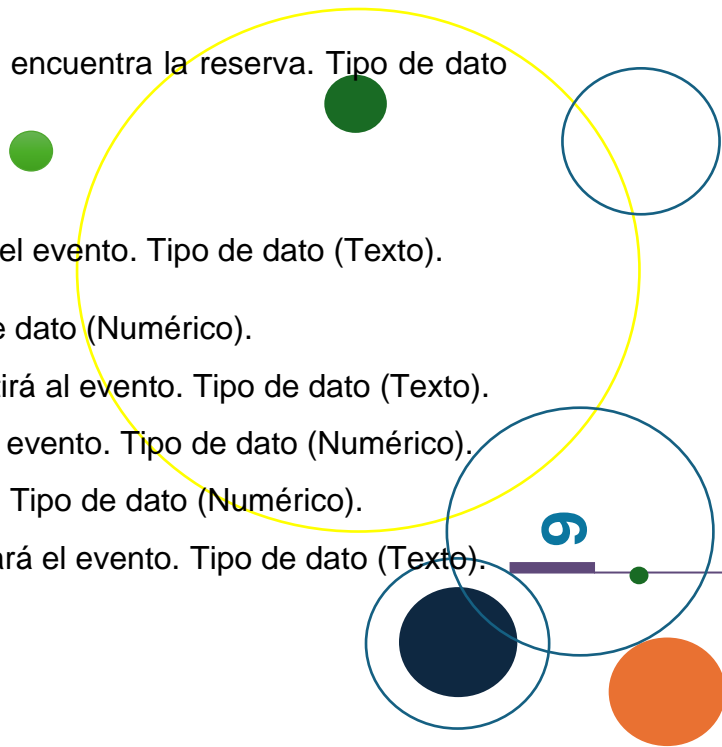
- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Nombre: Nombre del cliente. Tipo de dato (Texto).
- Apellido: Apellido del cliente. Tipo de dato (Texto).
- Email: Correo electrónico del cliente. Tipo de dato (Alfanumérico).
- Teléfono: Numero del teléfono del cliente. Tipo de dato (Numérico).

**“Reserva”:** Esta tabla contiene información referente a las reservas.

- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Evento: Acontecimiento que esta por suceder. Tipo de dato (Texto).
- Cliente: Información de la persona que reserva. Tipo de dato (Texto).
- Fecha de reserva: DD/MM/AA en que se realizó la reserva. Tipo de dato (Numérico).
- Número de personas: Numero de personas que asistirá al evento. Tipo de dato (Numérico).
- Estado de reserva: Estado en el que se encuentra la reserva. Tipo de dato (Texto).

**“Evento”:** Esta tabla contiene nombre del evento. Tipo de dato (Texto).

- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Nombre: Nombre de la persona que asistirá al evento. Tipo de dato (Texto).
- Fecha: DD/MM/AA en que se realizara el evento. Tipo de dato (Numérico).
- Hora: Hora en que se realizara el evento. Tipo de dato (Numérico).
- Lugar: Nombre del lugar donde se realizará el evento. Tipo de dato (Texto).



- Temática: Tipo de evento a llevarse a cabo. Tipo de dato (Texto).

**“Asiento”**: En esta tabla se contiene el numero de asientos disponibles para el evento. Tipo de dato (Numérico).

- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Num asiento: Numero de los asientos. Tipo de dato (Numérico).
- Evento id: Numero de evento único. Tipo de dato (Numérico).
- Estado: Estado de ocupación del asiento. Tipo de dato (Texto).

**“Zona asiento”**: Esta tabla contiene información referente a la zona de donde se encuentra el asiento.

- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Nombre: Nombre de la persona quien se sentará en el asiento. Tipo de dato (Texto).
- Evento id: Numero de evento único. Tipo de dato (Numérico).

**“Recursos”**: En esta tabla se contiene los recursos utilizados para realizar el evento.

- ID: Identificador único de la tabla. Tipo de dato (Numérico).
- Cantidad disponible: Numero de recursos utilizados para el evento. Tipo de dato (Numérico).

## VISTAS

- **VIEW VW\_VISTA\_DISPONIBILIDAD\_ASIENTOS.**

Muestra la disponibilidad de asientos en tiempo real.

- **VIEW VW\_VISTA\_RECURSOS.**



Muestra la cantidad de recursos utilizados para realizar el evento.

- **VIEW VW\_RESERVA\_DETALLE.**

Muestra con los detalles de las reservas realizadas.

- **VIEW VW\_RESERVAS\_POR\_TEMATICA**

Muestra con detalle las reservas por temática o tipo de evento realizado en tiempo real.

## **FUNCIONES**

- **FN\_ACTUALIZAR\_ESTADO\_RESERVA.**

Muestra en tiempo real el estado de la reserva realizada por el cliente.

- **FN\_CALCULAR\_TOTAL\_PERSONAS**

Muestra la cantidad total de personas que asistirán al evento.

- **FN\_CREAR\_RESERVA.**

Muestra la creación de la reserva al evento que el cliente haya escogido.

- **FN\_GENERAR\_FACTURA.**

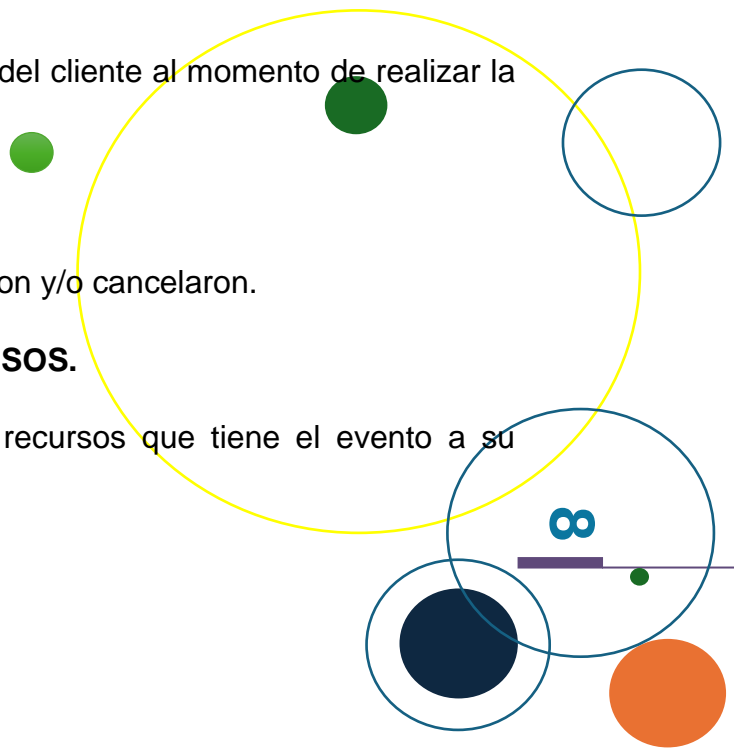
Muestra la realización de la boleta y/o factura del cliente al momento de realizar la reserva.

- **FN\_CANCELAR\_RESERVA.**

Muestra la cantidad de reservas que se anularon y/o cancelaron.

- **FN\_GENERAR\_INVENTARIO\_RECURSOS.**

Muestra en tiempo real el inventario de los recursos que tiene el evento a su disposición.





- **FN\_REGISTRAR\_ENTRADA\_SALIDA\_RECURSOS.**

Muestra en tiempo real la entrada y salida de recursos para realizar el evento.

- **FN\_VERIFICAR\_DISPONIBILIDAD\_RECURSOS.**

Muestra en tiempo real la disponibilidad de los recursos para realizar el evento.

## PROCEDIMIENTOS

- **SP\_CREAR\_RESERVA.**

Permite crear una reserva.

- **SP\_ACTUALIZAR\_ESTADO\_RESERVA.**

Permite actualizar en tiempo real el estado de la reserva.

- **SP\_CANCELAR\_RESERVA\_PROCEDURE**

Permite cancelar la reserva.

- **SP\_CALCULAR\_TOTAL\_PERSONAS\_EVENTO.**

Permite calcular el total de personas que asisten a un evento específico.

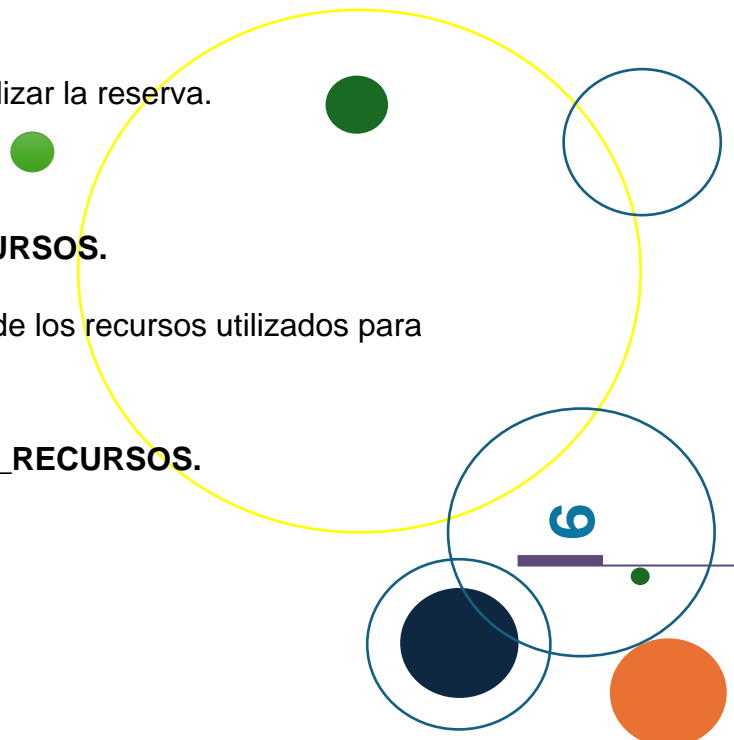
- **SP\_GENERAR\_FACTURA.**

Permite generar factura al cliente luego de realizar la reserva.

- **SP\_GESTIONAR\_INVENTARIO\_RECURSOS.**

Permite gestionar el inventario en tiempo real de los recursos utilizados para realizar el evento.

- **SP\_REGISTRAR\_ENTRADA\_SALIDA\_RECURSOS.**



Permite registrar la entrada y salida de recursos que se utilizan para realizar el evento.

- **SP\_VERIFICAR\_DISPONIBILIDAD\_RECURSOS.**

Permite verificar en tiempo real los dispositivos utilizados y el stock.

## TRIGGER

- **TGR\_GESTIONAR\_ESTADO\_ASIENTO.**

Actualiza el estado del asiento.

- **TGR\_CLIENTE\_INSERT\_TRIGGER.**

Actualiza información del cliente.

- **TGR\_VALIDAR\_FECHA\_HORA\_LUGAR.**

Valida la fecha, hora, lugar de la reserva del cliente.

- **TGR\_REGISTRO\_CAMBIO\_RECURSOS.**

Actualiza el stock de los recursos utilizados para el evento.

- **TGR\_RESERVA\_VALIDATE\_NUM\_PERSONAS.**

Valida el número de reservas por cliente.

- **TGR\_ACTUALIZAR\_PERSONAS\_POR\_TEMATICA.**

Actualiza el numero de personas que asiste al evento por temática.

- **TGR\_VALIDAD\_CAPACIDAD\_ZONA\_ASIENTO.**

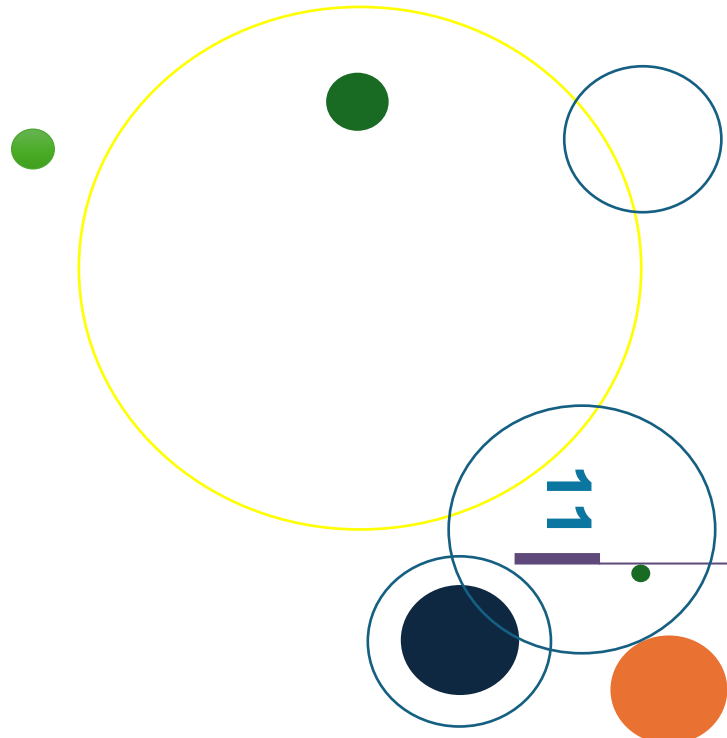
Valida la capacidad de las zonas por asiento.



## HERRAMIENTAS UTILIZADAS

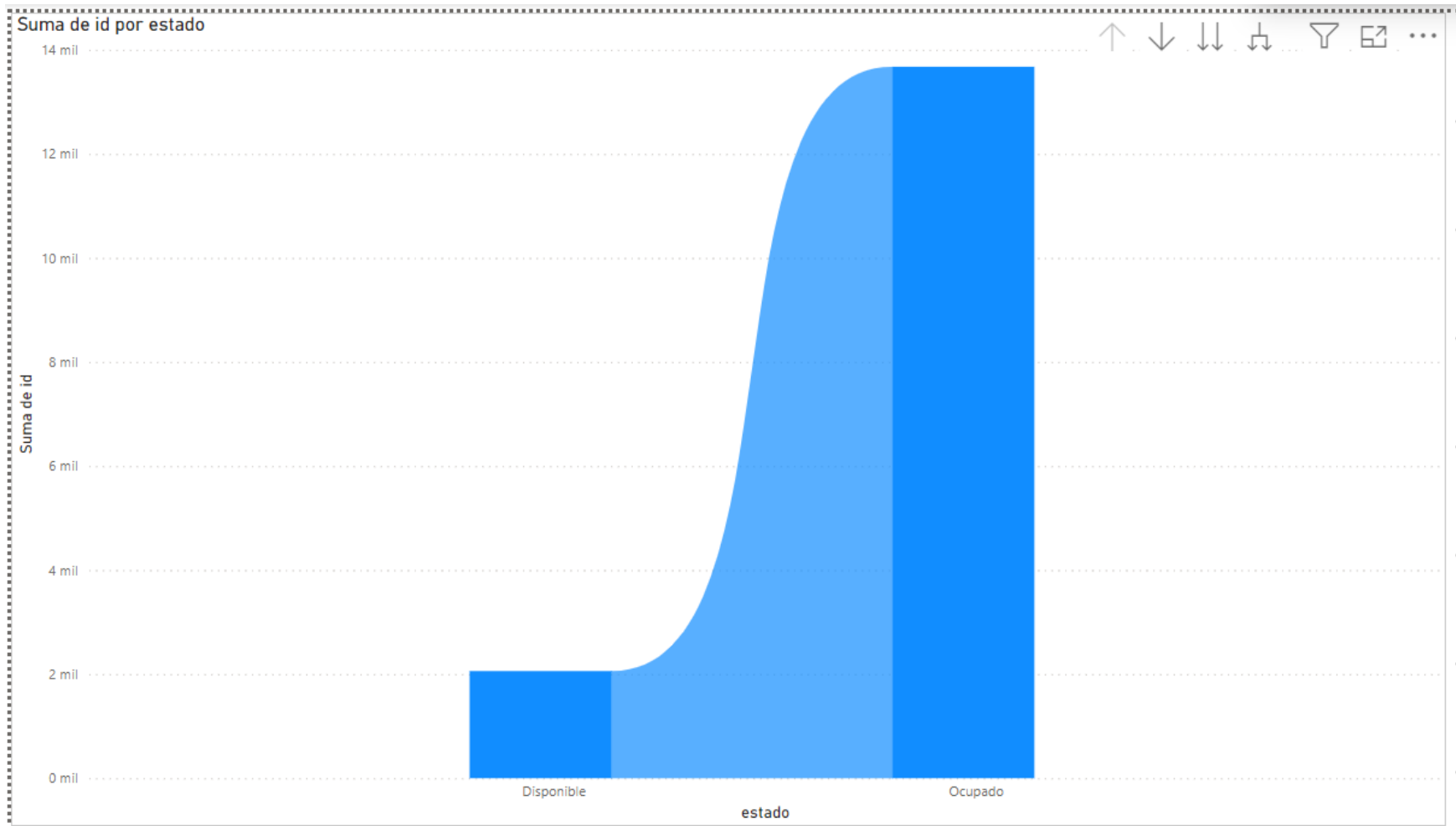
Las herramientas utilizadas para ayudar desde la elaboración hasta el vacío de datos para elaborar el respectivo reporte fueron:

1. **MySQL Workbench.**
2. **PowerBI.**

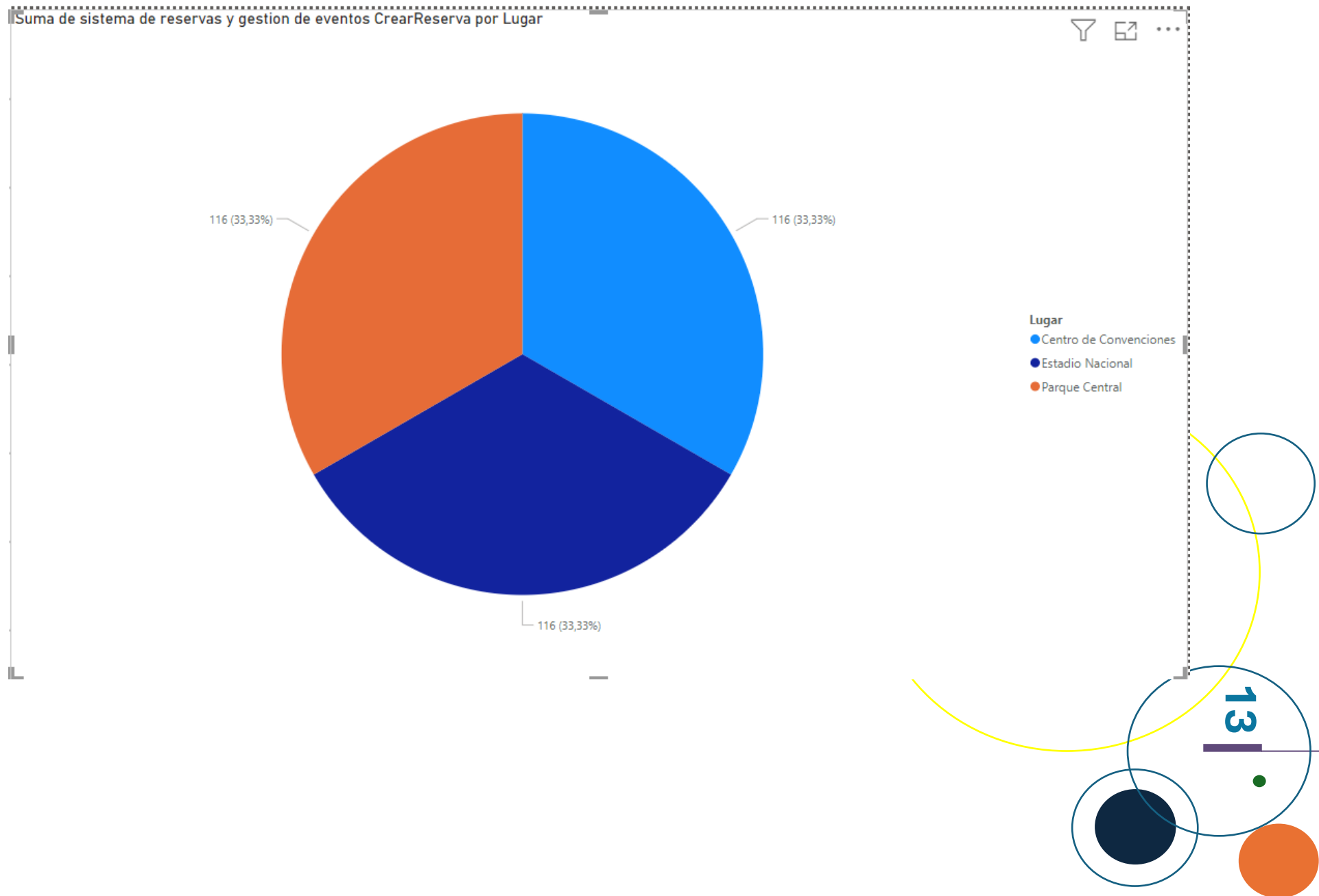


## REPORTE

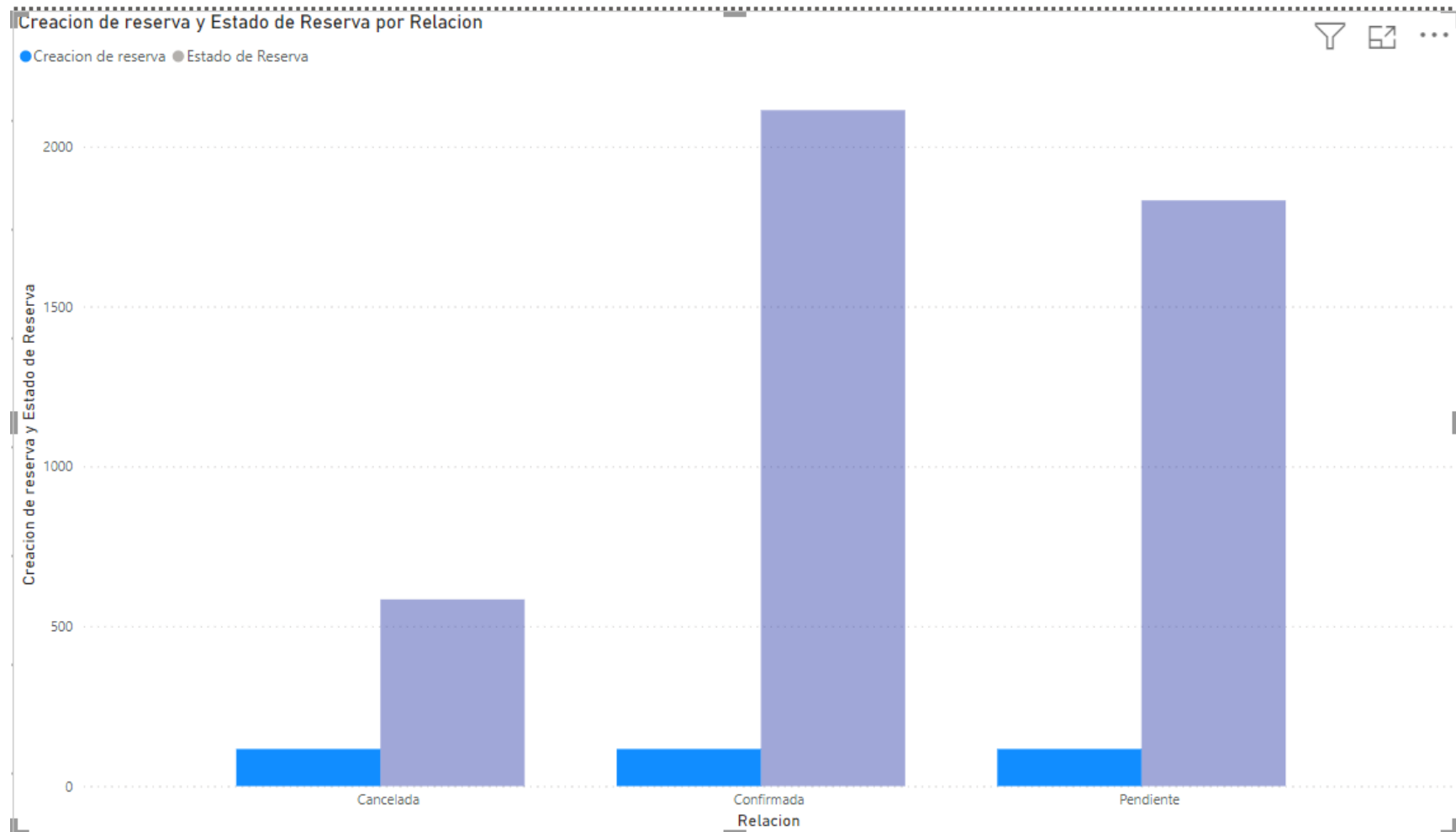
### 1. ESTADO DE LOS ASIENTOS (DISPONIBLE vs OCUPADOS)



## 2. REPRESENTA LAS RESERVAS POR SITIO



### 3. RELACION ENTRE CREACIÓN DE EVENTOS Y ESTADO DE RESERVA



## DCL (DATA CONTROL LANGUAGE)

1. **CREATE ROLE:** Se utilizo para crear los roles específicos

```
CREATE ROLE rol_administrador;
```

```
CREATE ROLE rol_cliente;
```

2. **GRANT:** Se utilizo para otorgar permiso específico al usuario o rol sobre objetos de la base de datos.

--Permisos para administradores

```
GRANT ALL PRIVILEGES ON *.* TO rol_administrador;
```

--Permisos para clientes

```
GRANT SELECT ON evento TO rol_cliente;
```

```
GRANT SELECT ON temática TO rol_cliente;
```

3. **CREATE USER:** Ahora, crea usuarios para representar a los administradores y clientes, y asigna los roles correspondientes a cada usuario.

```
CREATE USER 'freddyjimenez'@'hotmail.com' IDENTIFIED BY 'admin123';
```

```
GRANT rol_administrador TO 'freddyjimenez'@'hotmail.com';
```

```
CREATE USER 'Juanperez'@'gmail.com' IDENTIFIED BY 'cliente123';
```

```
GRANT rol_cliente TO 'Juanperez'@'gmail.com';
```

4. **REVOKE:** Si es necesario, se puedes revocar permisos adicionales que no son necesarios para los roles.

```
REVOKE ALL PRIVILEGES ON
```

```
sistema_de_reservas_y_gestion_de_eventos.* FROM rol_cliente;
```

5. **SHOW GRANTS:** Finalmente, verifica que los permisos se hayan otorgado correctamente utilizando la instrucción SHOW GRANTS.

**SHOW GRANTS FOR** 'freddyjimenez'@'hotmail.com';

**SHOW GRANTS FOR** 'Juanperez'@'gmail.com';

### **TCL** (TRANSACTION CONTROL LANGUAGE)

1. Creamos un usuario llamado cliente y le otorgamos permisos de SELECT e INSERT en la tabla reserva.
2. Iniciamos una transacción con **START TRANSACTION**.
3. Verificamos los permisos del usuario cliente con **SHOW GRANTS**.
4. Realizamos la inserción de una reserva en la tabla reserva.
5. Actualizamos la cantidad disponible de recursos en la tabla recursos.
6. Verificamos si hay suficientes recursos disponibles.
7. Si hay suficientes recursos y el cliente tiene permisos, confirmamos la transacción con **COMMIT**.
8. Si no hay suficientes recursos o el cliente no tiene permisos, revertimos la transacción con **ROLLBACK**.

--Crear usuario cliente y otorgarle permisos

**CREATE USER** 'juanperez'@'gmail.com' IDENTIFIED BY 'cliente123';

**GRANT SELECT, INSERT ON** reserva **TO** 'juanperez'@'gmail.com';

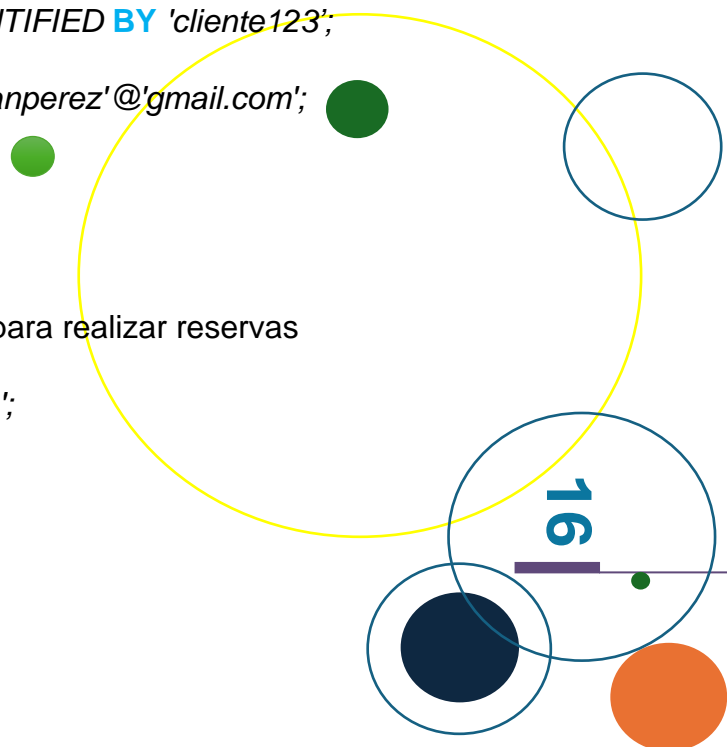
-- Iniciar la transacción

**START TRANSACTION;**

-- Verificar si el usuario cliente tiene permisos para realizar reservas

**SHOW GRANTS FOR** 'juanperez'@'gmail.com';

-- Realizar la reserva





**INSERT INTO** reserva (evento\_id, cliente\_id, fecha\_de\_reserva, num\_personas, estado\_reserva)

**VALUES** (1, 1, NOW(), 2, 'Pendiente');

-- Actualizar la cantidad disponible de recursos

**UPDATE** recursos **SET** cantidad\_disponible = cantidad\_disponible - 2 **WHERE** id = 1;

-- Verificar si hay suficientes recursos disponibles

**SELECT** cantidad\_disponible **FROM** recursos **WHERE** id = 1;

-- Si la cantidad disponible es suficiente, confirmar la transacción

**COMMIT**;

-- Si la cantidad disponible no es suficiente o el usuario no tiene permisos, revertir la transacción

**ROLLBACK**;

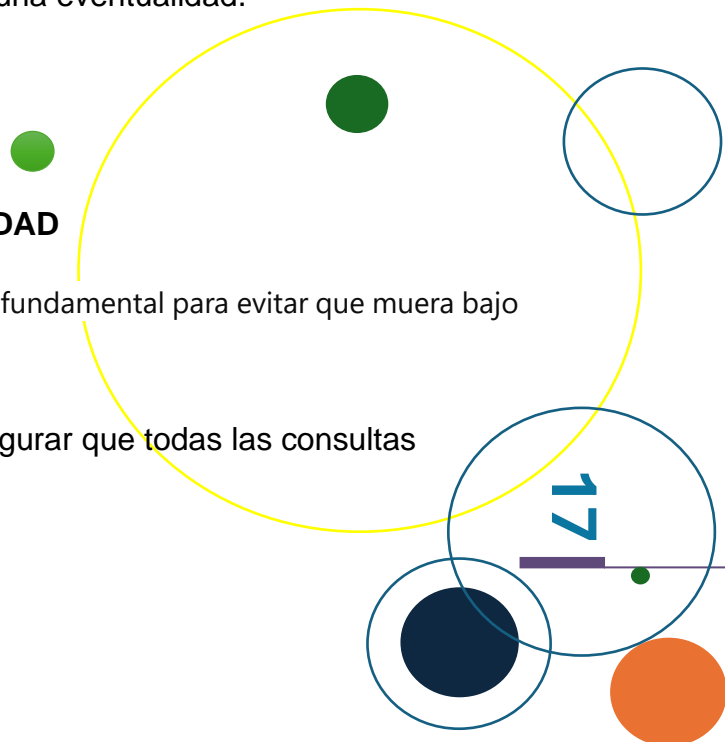
## BACKUP

Contiene el respaldo de todo el proyecto. Nos permitirá restablecer toda la base de datos sin perder la información en caso de alguna eventualidad.

## ESCALABILIDAD

Garantizar la escalabilidad de una base de datos es fundamental para evitar que muera bajo cargas de trabajo cada vez mayores:

1. **Optimización de Consultas:** Hay que asegurar que todas las consultas utilizadas estén optimizadas.



2. **Particionamiento de Tablas:** Se debe particionar las tablas. Esto puede ayudar a mejorar el rendimiento de consultas y operaciones de mantenimiento en grandes conjuntos de datos.
3. **Monitoreo de Rendimiento:** Se debe implementar un sistema de monitoreo de rendimiento que te permita supervisar el rendimiento de la base de. Esto te ayudará a optimizar el rendimiento de manera proactiva.
4. **Escalado Horizontal:** Para alivianar la carga de trabajo se recomienda escalado horizontal de la base de datos.
5. **Cacheo de Consultas:** Se debe utilizar una capa de caché para almacenar en caché los resultados de consultas frecuentes.
6. **Respaldo y Recuperación:** Se debe implementar un plan sólido de respaldo y recuperación para proteger tus datos contra pérdidas y corrupción.
7. **Actualización Regular:** Se debe mantener actualizada la base de datos con las últimas versiones de software y parches de seguridad.

## CONCLUSION

En resumen, la base de datos del "sistema de reservas y gestión de eventos" proporciona una base sólida y segura para la aplicación, permitiendo un almacenamiento eficiente y organizado de la información y garantizando un acceso seguro y confiable a los datos en todo momento. Con un mantenimiento adecuado y una atención continua a la seguridad y el rendimiento, la base de datos seguirá siendo un activo valioso para el sistema en el futuro.

