

# CHILL HOTEL

“LIBERAR LA MENTE”



**DOCUMENTACIÓN  
DEL SISTEMA**

# Sistema de gestión integral de Chill Hotel

---

## 1. Introducción y objetivación

### Introducción

El presente proyecto detalla la implementación de una base de datos relacional diseñada para soportar y centralizar las operaciones principales de un hotel (inventado) llamado **Chill Hotel**. El sistema está estructurado para gestionar las reservas de huéspedes, la disponibilidad y estado de las habitaciones, la administración del personal y la logística del servicio a la habitación.

### Objetivo

El objetivo fundamental de esta base de datos es lograr una gestión de excelencia en el manejo y veracidad de los datos del hotel buscando darle un salto de calidad, un antes y después, un punto de inflexión de mejoría en la calidad del hotel, mejorando cuestiones como:

- **Gestión Operacional:** Proveer al personal de recepción y administrativo una herramienta fiable para consultar la disponibilidad, realizar reservas, y gestionar los procesos de check-in y check-out de manera ágil.
- **Logística:** Optimizar la gestión del personal de limpieza y mantenimiento, permitiendo consultar en tiempo real el estado de cada habitación ('Disponible', 'Ocupada', 'Mantenimiento'). Además, el sistema de productos permitirá un control de inventario para el servicio a la habitación, donde el cliente podrá pedir lo que desee.

## 2. Situación Problemática

La problemática que **Chill hotel** pretende resolver es la gestión de información descentralizada, como hojas de cálculo, registros en papel o sistemas aislados, generando arrastra errores como, por ejemplo: Buena cantidad de errores humanos, ineficiencia operativa, falta de visibilidad estratégica, experiencia del cliente insatisfecha, etc.

La implementación de este sistema de base de datos relacional quiere automatizar procesos, garantizando la consistencia de los datos y proporcionando una plataforma sólida para el crecimiento y la mejora continua del servicio.

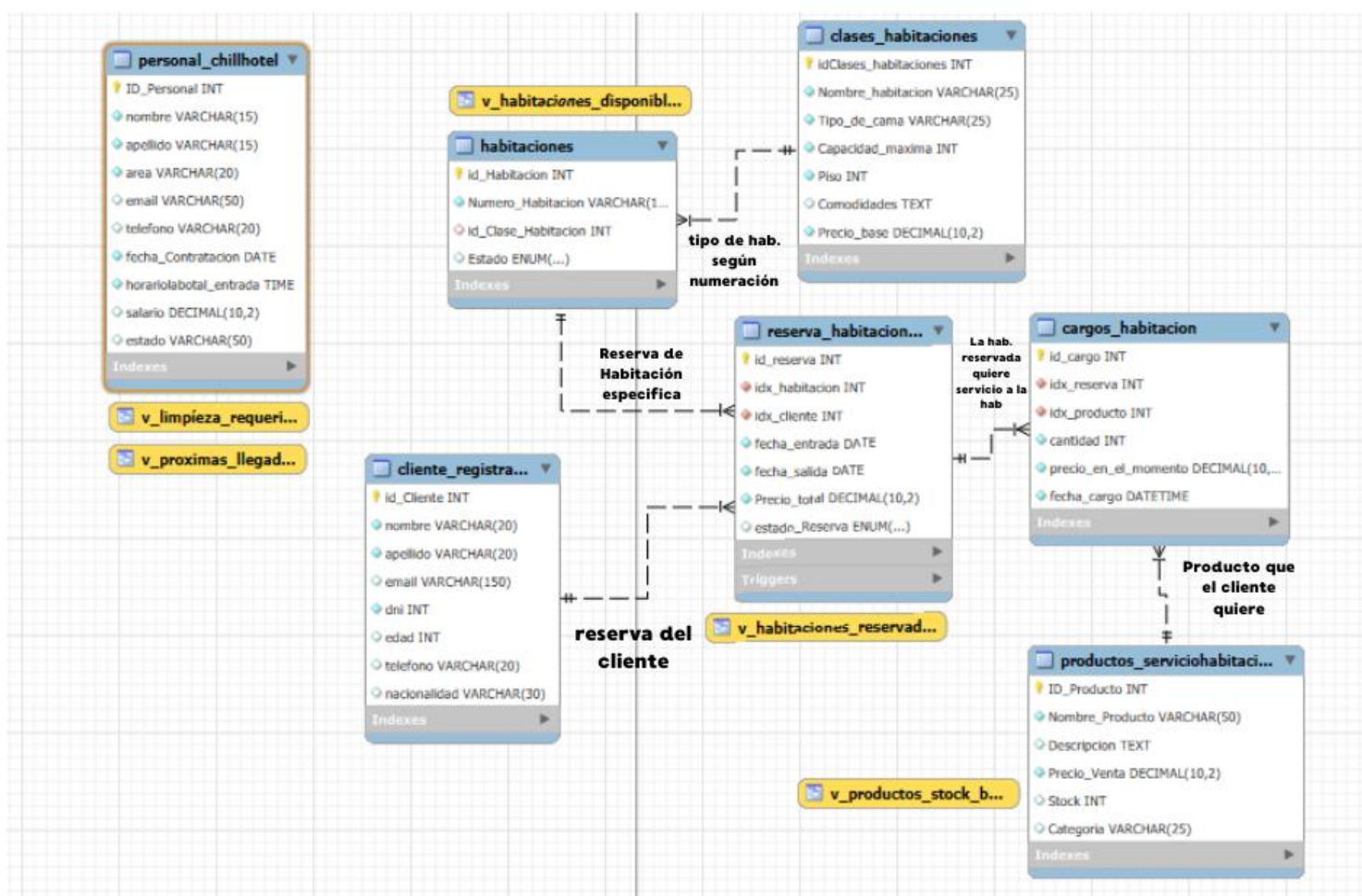
Por otro lado, entendemos muy bien, que una persona acude a un hotel, porque está de viaje (sea por el motivo que sea) y: Desea descansar para poder visitar la ciudad/hacer negocios, desea despreocuparse de molestar a sus parientes de la zona, desea un trato sin igual con buena alimentación, con limpieza integra... Pero no solo eso, sino también estar abierto para aquellas personas locales, que quieran vivir una experiencia realmente gratificante a modo de "Spa", donde se den "una escapadita" del ajetreo diario, como alternativa de aquellas actividades que parecieran "relajarle" pero que solo esquivan la realidad.

En pocas palabras, sacarse la mochila de sus preocupaciones, relajarse un rato y rendir al máximo al siguiente día. Para eso, nosotros deberemos rendir al máximo para darle la mejor calidad.

### 3. Modelo de Negocio

**Chill Hotel** se posiciona como un hotel de tipo boutique, enfocado en ofrecer una experiencia inigualable, tanto a turistas como a viajeros de negocios, buscando que el cliente logre relajarse lo máximo posible, disfrutando por cada uno de los escenarios del Hotel.

### 4. Diagrama Entidad-Relación (E-R)





## 5. Tablas

El código de las tablas lo encontrarás en SQL [clicando aquí](#)

1

### Nombre: Cliente\_registrado

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
id_Cliente	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identificador único y clave principal.
nombre	VARCHAR(20)	NOT NULL	Nombre del cliente.
apellido	VARCHAR(20)	NOT NULL	Apellido del cliente.
email	VARCHAR(150)	UNIQUE, DEFAULT NULL	Correo electrónico. Debe ser único si se proporciona un valor.
dni	INT	UNIQUE, NOT NULL	Documento Nacional de Identidad. Requerido y único.
edad	INT UNSIGNED	DEFAULT NULL	Edad del cliente (solo números positivos o cero).
telefono	VARCHAR(20)	UNIQUE, DEFAULT NULL	Número de teléfono. Debe ser único si se proporciona un valor.
nacionalidad	VARCHAR(30)	DEFAULT 'Argentina'	Nacionalidad del cliente. Por defecto es 'Argentina'.

**Descripción:** En esta tabla podemos ver que es necesario para el hotel, conocer al cliente con el que vamos a trabajar, sabiendo sus datos más principales, para un mejor acercamiento hacia el cliente, pero principalmente para temas legales que pudieran llegar a surgir

**Objetivo:** Dar un espacio y lugar de datos a los clientes que paguen por su estadío mediante un registro.

**Tablas relacionadas:** reserva\_habitación

2

### Nombre: Habitaciones

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
id_Habitacion	INT	PRIMARY KEY, AUTO_INCREMENT	Clave principal y contador automático para identificar la habitación.
Numero_Habitacion	VARCHAR(10)	NOT NULL, UNIQUE	El número o código visible de la habitación (ej: '101', 'A2'). Debe ser único.
id_Clase_Habitacion	INT	FOREIGN KEY	Enlaza a la tabla Clases_Habitaciones, indicando el tipo (ej: simple, doble, suite).
Estado	VARCHAR(50)	DEFAULT 'Disponible'	Indica el estado actual (ej: 'Disponible', 'Ocupada', 'Requiere Limpieza'). Por defecto es 'Disponible'.

**Descripción:** En esta tabla cubrimos estos datos primordiales para un hotel: sus habitaciones, encontrando en ellas los datos como el número de habitación, su estado y la relación según su “Clase de habitación” particular. Sin embargo, y como es mencionado, hay una tabla especial para “Como” o “Que tiene esta habitación, a que rango pertenece”, para una mejor consistencia en los datos.

**Objetivo:** Registrar las habitaciones que tiene el hotel según su numero de habitación, su clase y su estado.

**Tablas relacionadas:** clases\_habitaciones, reserva\_habitaciones

3

**Nombre: Clase\_habitaciones**

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
<b>idClases_habitaciones</b>	INT	PRIMARY KEY, NOT NULL	Clave principal que identifica de forma única la clase o tipo de habitación.
<b>Nombre_habitacion</b>	VARCHAR(25)	UNIQUE, NOT NULL	Nombre descriptivo de la clase (ej: 'Suite Presidencial', 'Doble Estándar'). Debe ser único.
<b>Tipo_de_cama</b>	VARCHAR(25)	NOT NULL	Describe el tipo de cama (ej: 'King', 'Queen', 'Twin').
<b>Capacidad_maxima</b>	INT	NOT NULL	El número máximo de personas que pueden alojarse en esta clase de habitación.
<b>Piso</b>	INT	NOT NULL	Indica el piso donde se encuentran estas habitaciones.
<b>Comodidades</b>	TEXT	DEFAULT NULL	Campo para listar comodidades adicionales (ej: 'Jacuzzi', 'Balcón', 'Minibar').
<b>Precio_base</b>	DECIMAL(10, 2)	NOT NULL	Precio base por noche de esta clase de habitación, con dos decimales.

**Descripción:** En esta tabla podremos ver la “Clase” de la habitación, es decir, si es Premium, para cuantas personas es, el tipo de cama que contienen.

**Objetivo:** Registrar las características de una habitación según su “tipo” o “clase”.

**Tablas relacionadas:** habitaciones

4

**Nombre: Reserva\_habitaciones**

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
<b>id_reserva</b>	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identificador único y clave principal de la reserva.
<b>idx_habitacion</b>	INT	NOT NULL, FOREIGN KEY	Enlaza a la tabla Habitaciones.
<b>idx_cliente</b>	INT	NOT NULL, FOREIGN KEY	Enlaza a la tabla Cliente_registrado.
<b>fecha_entrada</b>	DATE	NOT NULL	La fecha en la que el cliente entra a la habitación.
<b>fecha_salida</b>	DATE	NOT NULL	La fecha en la que el cliente sale de la habitación.
<b>precio_total</b>	DECIMAL(10, 2)	NOT NULL	Costo total de la reserva, con dos decimales.
<b>estado_Reserva</b>	VARCHAR(50)	DEFAULT 'Confirmada'	Estado actual de la reserva (ej: 'Confirmada', 'Cancelada', 'Finalizada').

**Descripción:** En esta tabla encontramos la relación principal entre un cliente y la habitación específica elegida por él: La reserva. Es por eso que también guarda algunos datos extra como cuando entro, cuando salio, etc

**Objetivo:** Registrar las reservas identificando al cliente y a la habitación que el cliente quiere reservar

**Tablas relacionadas:** habitaciones, cliente\_registrado

5

**Nombre: Personal\_chillhotel**

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
<b>ID_Personal</b>	INT	PRIMARY KEY, NOT NULL, AUTO_INCREMENT	Identificador único del empleado.
<b>nombre</b>	VARCHAR(15)	NOT NULL	Nombre de pila del empleado.
<b>apellido</b>	VARCHAR(15)	NOT NULL	Apellido del empleado.
<b>area</b>	VARCHAR(20)	NOT NULL	Área o departamento de trabajo (ej: 'Limpieza', 'Recepción', 'Cocina').
<b>email</b>	VARCHAR(50)	UNIQUE	Correo electrónico del empleado. Debe ser único.
<b>telefono</b>	VARCHAR(20)	UNIQUE	Número de teléfono del empleado. Debe ser único.
<b>fecha_Contratacion</b>	DATE	NOT NULL	Fecha en que el empleado fue contratado.
<b>horariolabotal_entrada</b>	TIME	NOT NULL	Hora de inicio del turno de trabajo.
<b>salario</b>	DECIMAL(10, 2)	DEFAULT NULL	Salario del empleado, con dos decimales.
<b>estado</b>	VARCHAR(50)	DEFAULT 'Activo'	Estado laboral del empleado (ej: 'Activo', 'Inactivo', 'Suspendido').

**Descripción:** En esta tabla encontramos los datos del personal que tiene nuestro hotel “Chill hotel”, Contando con las características principales de estos, tales como fecha de contratación, a que “área” pertenece, etc.

**Objetivo:** Registrar el personal del hotel

**Tablas relacionadas:** No tiene (podría hacerse en una macro para gestión de limpieza, por ejemplo)

**Nombre: Productos\_servicioHabitacion**

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
<b>ID_Producto</b>	INT	PRIMARY KEY, AUTO_INCREMENT	Identificador único del producto o servicio.
<b>Nombre_Producto</b>	VARCHAR(50)	NOT NULL	Nombre del artículo disponible para servicio de habitaciones.
<b>Descripcion</b>	TEXT	DEFAULT NULL	Descripción detallada del producto.
<b>Precio_Venta</b>	DECIMAL(10, 2)	NOT NULL	Precio al que se vende el producto, con dos decimales.
<b>Stock</b>	INT	DEFAULT 0	Cantidad de unidades disponibles en inventario.
<b>Categoria</b>	VARCHAR(25)	DEFAULT NULL	Categoría del producto (ej: 'Comidas', 'Bebidas', 'Snacks').

**Descripción:** Unos de los servicios que ofrece el hotel, es la opción del “pedido a la habitación”, el cual un cliente que ya tiene una habitación reservada, puede pedir algunos productos como “Snacks”, “Coca cola”, etc. Esta tabla describe los productos que están disponible para esta funcionalidad de “servicio a la habitación”

**Objetivo:** Registrar los productos disponibles para el “servicio a la habitación”

**Tablas relacionadas:** cargo\_habitacion



7

**Nombre: Cargos\_habitacion**

Columna	Tipo de Dato	Restricciones	Notas / Comentarios
<b>id_cargo</b>	INT	PRIMARY KEY, AUTO_INCREMENT	Clave principal y contador automático para el registro del cargo.
<b>idx_reserva</b>	INT	NOT NULL, FOREIGN KEY	Enlaza con la tabla Reserva_habitaciones (la reserva a la que se aplica el cargo).
<b>idx_producto</b>	INT	NOT NULL, FOREIGN KEY	Enlaza con la tabla Productos_servicioHabitacion (el artículo o servicio cargado).
<b>cantidad</b>	INT	NOT NULL, DEFAULT 1	Número de unidades del producto cargado. Por defecto es 1.
<b>precio_en_el_momento</b>	DECIMAL(10, 2)	NOT NULL	El precio unitario del producto en el momento exacto del cargo (para mantener el historial).
<b>fecha_cargo</b>	DATETIME	NOT NULL	La fecha y hora exactas en que se realizó el cargo.

**Descripción:** Esta tabla hace la integración entre el “pedido a la habitación” con la habitación específica que lo pide, cargando la cantidad de productos, cuanto se le aumenta de precio en su cuenta, etc

**Objetivo:** Registrar los productos que los clientes piden en servicio a la habitación, para la habitación reservada por él

**Tablas relacionadas:** reserva\_habitacion, Productos\_servicioHabitacion

## 6. Vistas

El código de las views en general, lo encontrarás en SQL [clicando aquí](#)

### 1. v\_habitaciones\_reservadas

**Descripción:** Esta vista combina información del cliente, la habitación y la reserva para ver quién ocupa qué, y cuándo.

**Objetivo:** Ver que habitaciones fueron efectivamente reservadas, y cuando

Columna	Descripción del Dato	Origen del Dato
nombre_completo	Nombre y apellido del cliente.	Cálculo (CONCAT)
Numero_Habitacion	El número visible de la habitación.	Tabla Habitaciones
Estado	Estado actual de la habitación (ej: 'Ocupada').	Tabla Habitaciones
fecha_entrada	Fecha de inicio de la reserva.	Tabla Reserva_habitaciones
fecha_salida	Fecha de fin de la reserva.	Tabla Reserva_habitaciones

---

### 2. VISTA: v\_proximas\_llegadas

**Descripción:** Esta vista se centra en los clientes que ingresarán al hotel **hoy o mañana**.

**Objetivo:** Ver los datos de las habitaciones próximas a llegar

Columna	Descripción del Dato	Origen del Dato
nombre_completo	Nombre y apellido del cliente.	Cálculo (CONCAT)
Numero_Habitacion	El número visible de la habitación.	Tabla Habitaciones
Estado	Estado actual de la habitación.	Tabla Habitaciones
fecha_entrada	Fecha en la que el cliente va a ingresar.	Tabla Reserva_habitaciones

### 3. VISTA: v\_limpieza\_requerida

**Descripción:** Esta vista ayuda al personal de limpieza a priorizar habitaciones que quedaron vacías hoy y necesitan ser preparadas.

**Objetivo:** Ver cuando una habitación está necesitando de una limpieza

Columna	Descripción del Dato	Origen del Dato
nombre_completo	Nombre y apellido del cliente que acaba de irse.	Cálculo (CONCAT)
Numero_Habitacion	El número visible de la habitación.	Tabla Habitaciones
Estado	Estado de la habitación (será 'Requiere Limpieza').	Tabla Habitaciones
fecha_salida	Fecha en la que el cliente salió (hoy).	Tabla Reserva_habitaciones

### 4. VISTA: v\_productos\_stock\_bajo

**Descripción:** Esta vista es un informe de inventario para la reposición de productos de servicio de habitación.

**Objetivo:** Ver cuando un producto está bajo en cantidad de stock

Columna	Descripción del Dato	Origen del Dato
nombre_producto	Nombre del producto o servicio.	Tabla Productos_servicioHabitacion
stock	Cantidad actual en inventario (siempre $\leq 10$ ).	Tabla Productos_servicioHabitacion
Categoria	Categoría del producto (ej: 'Bebidas').	Tabla Productos_servicioHabitacion

---

### 5. VISTA: v\_habitaciones\_disponibles

**Descripción:** Esta vista ofrece un catálogo de las habitaciones disponibles para los recepcionistas.

**Objetivo:** Ver aquellas habitaciones que están disponibles actualmente

Columna	Descripción del Dato	Origen del Dato
Nombre_habitacion	Nombre de la clase de habitación (ej: 'Suite Doble').	Tabla Clases_Habitaciones
Piso	Número de piso.	Tabla Clases_Habitaciones
Tipo_de_cama	Tipo de cama que tiene la habitación.	Tabla Clases_Habitaciones
Capacidad_maxima	Número máximo de huéspedes.	Tabla Clases_Habitaciones
Precio_base	Precio base por noche de esta clase.	Tabla Clases_Habitaciones
Numero_Habitacion	El número visible de la habitación.	Tabla Habitaciones

## 7.Funciones

El código de las funciones en general, lo encontrarás en SQL [clicando aquí](#)

### 1. Función: f\_calcular\_noches\_de\_hospedaje

Parámetro	Tipo	Descripción
p_fecha_entrada	DATE	Fecha de inicio de la estancia.
p_fecha_salida	DATE	Fecha de finalización de la estancia.
Retorna	INT	Número total de noches de hospedaje.

**Descripción de la Función:** Esta función es una herramienta simple y determinista (siempre devuelve el mismo resultado para los mismos parámetros) diseñada para calcular la duración de una estancia. Utiliza la función nativa de MySQL, DATEDIFF(), para obtener la diferencia en días entre la p\_fecha\_salida y la p\_fecha\_entrada.

**Objetivo de la Función:** Determinar de manera rápida y precisa el número de noches de alojamiento. Se utiliza principalmente en operaciones de cálculo y validación de reservas.

### 2. Función: f\_calcular\_total\_estancia

Parámetro	Tipo	Descripción
p_id_reserva	INT	ID de la reserva cuyo total final se desea calcular.
Retorna	DECIMAL(10, 2)	Costo total de la estancia (habitación + servicios).

**Descripción de la Función:** Esta función realiza un cálculo compuesto para obtener el monto final que el cliente debe pagar al momento de finalizar su estancia. El proceso se divide en dos pasos principales:

1. Precio Base: Recupera el precio\_total precalculado de la habitación de la tabla Reserva\_habitaciones.
2. Cargos por Servicios: Suma el total de todos los consumos registrados en la tabla Cargos\_Habitacion para ese p\_id\_reserva. Se utiliza IFNULL para asegurar que el cálculo no falle si no existen consumos asociados.  
Finalmente, suma el precio base de la habitación (vprecio\_base) y el total de los cargos por servicios (vtotal\_cargos) para obtener el total a pagar.

**Objetivo de la Función:** Centralizar la lógica de facturación final.

**Tablas Afectadas:** Reserva\_habitaciones (Lectura), Cargos\_Habitacion (Lectura).

## 8. Procedimientos almacenados (Stored Procedures)

El código de los SP en general, lo encontrarás en SQL [clicando aquí](#)

### 1. Stored Procedure: sp\_agregar\_servicio\_habitacion

Parámetro	Tipo	Descripción
p_id_reserva	INT	ID de la reserva a la que se aplicará el cargo.
p_id_producto	INT	ID del producto o servicio consumido.
p_cantidad	INT	Cantidad de unidades del producto.

**Descripción del Procedimiento:** Este procedimiento gestiona el proceso de cargar un producto o servicio a la cuenta de una reserva. Primero, verifica que el estado\_Reserva actual de la reserva sea 'Activa'. Luego, recupera el Precio\_Venta actual del producto desde la tabla Productos\_servicioHabitacion.

**Objetivo del Procedimiento:** Registrar consumos y garantizar la integridad del cargo.

**Tablas Afectadas:** Reserva\_habitaciones (Lectura), Productos\_servicioHabitacion (Lectura), Cargos\_Habitacion (Inserción).

### 2. Stored Procedure: sp\_check\_in

Parámetro	Tipo	Descripción
p_id_reserva	INT	ID de la reserva a la que se aplicará el <i>check-in</i> .

**Descripción del Procedimiento:** Este procedimiento ejecuta las acciones necesarias cuando un huésped llega para iniciar su estancia. Primero, consulta la tabla Reserva\_habitaciones para obtener el ID de la habitación reservada (var\_id\_habitacion) y el estado\_Reserva actual.

**Objetivo del Procedimiento:** Formalizar la ocupación de la habitación e iniciar la reserva.

**Tablas Afectadas:** Reserva\_habitaciones (Lectura y Actualización), Habitaciones (Actualización).

## 9. Triggers

El código de los TRIGGERS en general, lo encontrarás en SQL [clicando aquí](#)

### 1. Trigger: tr\_actualizar\_estado\_habitacion\_reserva

**Descripción del Trigger:** Este TRIGGER se ejecuta inmediatamente después (AFTER) de que una nueva reserva es insertada (INSERT) en la tabla Reserva\_habitaciones. El mecanismo utiliza el valor de la nueva fila (NEW.idx\_habitacion) para identificar la habitación afectada.

**Objetivo del Trigger:** Automatizar la actualización del estado de la habitación.

**Tabla Relacionada:** Habitaciones (Actualiza el campo Estado).

### 2. Trigger: tr\_proteger\_estados\_finales\_reserva

**Descripción del Trigger:** Este TRIGGER se ejecuta antes (BEFORE) de que cualquier fila sea actualizada (UPDATE) en la tabla Reserva\_habitaciones. Evalúa el estado original de la reserva (OLD.estado\_Reserva) comparándolo con los estados finales 'Finalizada', 'Cancelada' o 'No Show'.

**Objetivo del Trigger:** Garantizar la integridad y la inmutabilidad de los datos históricos.

**Tabla Relacionada:** Reserva\_habitaciones (Impide la actualización del campo estado\_Reserva).