

TRABAJO PRÁCTICO GESTIÓN DE DATOS

1er Cuatrimestre – 2018

FRBA-Hotel

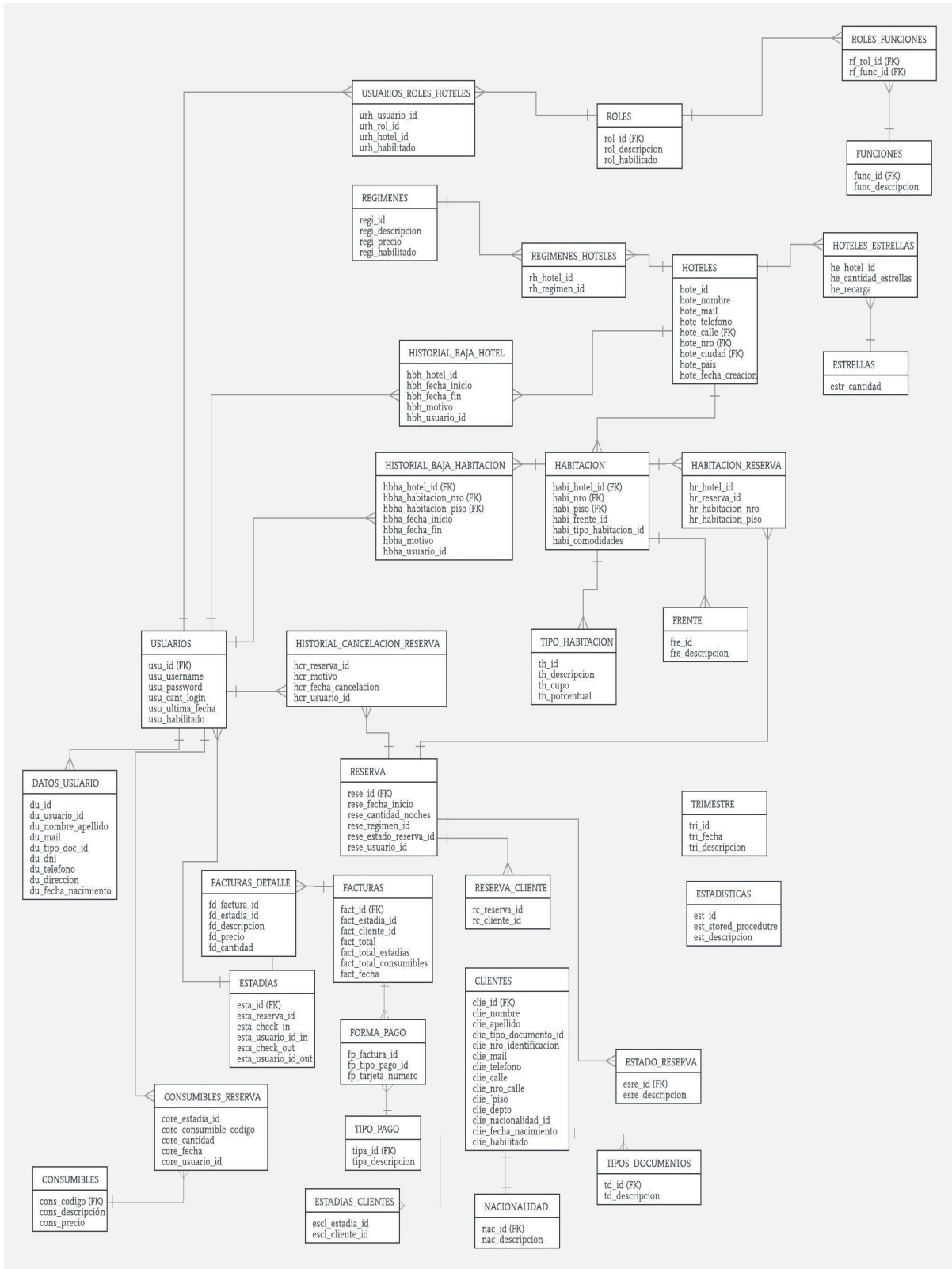
Grupo PUNTO_ZIP:

- Julián Goldfarb 155996-5
- Gonzalo Rochlin 155986-2
- Marcos Ortiz 156545-0
- Nicolás Burza 156033-5

Índice

DER.....	3
Estructuras de datos.....	4
<i>Tablas.....</i>	<i>4</i>
<i>StoredProcedures.....</i>	<i>12</i>
<i>Funciones.....</i>	<i>25</i>
Aplicación C#.....	26

DER



Estructuras de datos

Tablas

- **Usuario:** Tabla con los usuarios con responsabilidad en el hotel, ya sea administrador o recepcionista. El número de intentos máximo del login antes de bloquear el usuario es de 3.
 - **Id_Usuario:** varchar(20) (PK)
 - **Password:** varchar(64)
 - **Cantidad_Login:** int
 - **Ultima_Fecha:** datetime
 - **Habilitado:** bit
- **Rol:** Tabla que contiene los tipos de roles que existen.
 - **Id_Rol:** int IDENTITY(1,1) (PK)
 - **Descripcion:** varchar(20)
 - **Habilitado:** bit
- **Funcionalidad:** Tabla que contiene las funcionalidades del sistema a las que se pueden acceder
 - **Id_Funcionalidad:** int (PK)
 - **Descripcion:** varchar(40)
- **Rol_Funcionalidad:** Tabla que relaciona las funcionalidades del sistema, según el rol que se tenga.
 - **Id_Rol:** int (PK, FK Rol)
 - **Id_Funcionalidad:** int (PK, FK Funcionalidad)
- **Tipo_Identificacion:** Tabla de parametría de tipos de documentos.
 - **Id_Tipo_Identificacion:** int IDENTITY(1,1) (PK)
 - **Descripcion:** varchar(255)

- **Datos_Usuario:** Tabla con los datos personales de los usuarios (administrador, recepcionista).
 - **Id_Usuario:** varchar(20) (FK Usuario)
 - **Nombre_Apellido:** varchar(50)
 - **Mail:** varchar(255)
 - **Tipo_DNI:** int
 - **Nro_DNI:** int
 - **Telefono:** varchar(20)
 - **Direccion:** varchar(50)
 - **Fecha_Nacimiento:** datetime

- **Hotel:** Tabla de los hoteles que se tienen datos.
 - **Id_Hotel:** int IDENTITY(1,1)
 - **Nombre:** varchar(255)
 - **Mail:** varchar(20)
 - **Telefono:** varchar(20)
 - **Calle_Direccion:** varchar(255) (PK)
 - **Calle_Nro:** numeric(18,0) (PK)
 - **Ciudad:** varchar(255) (PK)
 - **Pais:** varchar(255)
 - **Fecha_Creacion:** datetime

- **Estrellas:** Tabla que especifica las estrellas.
 - **Estrellas:** int

- **Hotel_Estrellas:** Tabla con la cantidad de estrellas y su recargo por hotel.
 - **Id_Hotel:** int (FK Hotel)
 - **Cantidad_Estrellas:** int (FK Estrellas)
 - **recarga:** numeric(18,0)

- **Usuario_Rol_Hotel:** Tabla que almacena los roles asignados para cada usuario y hotel y si este se encuentra habilitado.
 - **Id_Usuario:** int (FK Usuario)
 - **Id_Rol:** int (FK Rol)
 - **Id_Hotel:** int (FK Hotel)
 - **Habilitado:** bit

- **Regimen:** Tabla con todos los regímenes disponibles.
 - **Id_Regimen:** int IDENTITY(1,1) (PK)
 - **Descripcion:** varchar(255)
 - **Precio:** numeric(18,2)
 - **Habilitado:** bit

- **Regimen_Hotel:** Tabla que almacena los regímenes disponibles para cada hotel.
 - **Id_Hotel:** int (FK Hotel)
 - **Id_Regimen:** int (FK Regimen)

- **Tipo_Habitacion:** Tabla con los distintos tipos de habitaciones que existen.
 - **Id_Tipo_Habitacion:** numeric(18,0)
 - **Descripcion:** varchar(255)
 - **Cupo:** int
 - **Porcentual:** numeric(18,2)

- **Historial_Baja_Hotel:** Tabla con los registros de baja momentánea del hotel, con sus respectivos momentos.
 - **Id_Hotel:** int (FK Hotel)
 - **Fecha_Inicio:** datetime
 - **Fecha_Fin:** datetime
 - **Motivo:** varchar(255)

- ***Id_Usuario:*** int (FK Usuario)
- **Frente:** Parametría de tipo de frente
 - ***Id_Frente:*** int IDENTITY(1,1)
 - ***Descripcion:*** varchar(255)
- **Habitacion:** Tabla con cada habitación dependiendo del hotel.
 - ***Id_Hotel:*** int (PK, FK Hotel)
 - ***Nro:*** int (PK)
 - ***Piso:*** int (PK)
 - ***Frente:*** int
 - ***Tipo_Habitacion:*** numeric(18,0) (FK Tipo_Habitacion)
 - ***Comodidades:*** varchar(255)
- **Historial_Baja_Habitacion:** Tabla con los registros de baja momentánea de las habitaciones, con sus respectivos momentos.
 - ***Id_Hotel:*** int (PK, FK Habitacion)
 - ***Habitación_Nro:*** int (PK, FK Habitacion)
 - ***Habitación_Piso:*** int (PK, FK Habitacion)
 - ***Fecha_Inicio:*** datetime
 - ***Fecha_Fin:*** datetime
 - ***Motivo:*** varchar(255)
 - ***Id_Usuario:*** int (FK Usuario)
- **Nacionalidad:** Tabla de parametría de las nacionalidades.
 - ***Id_Nacionalidad:*** int IDENTITY(1,1) (PK)
 - ***Descripcion:*** varchar(255)

- **Cientes:** Tabla con los datos personales de los clientes registrados en el sistema.
 - **Id_Cliente:** int IDENTITY(1,1) (PK)
 - **Nombre:** varchar(255)
 - **Apellido:** varchar(255)
 - **Tipo_Identificacion:** int (FK Tipo_Identificacion)
 - **Nro_identificacion:** numeric(18,0)
 - **Mail:** varchar(255)
 - **Telefono:** varchar(255)
 - **Calle_Direccion:** varchar(255)
 - **Calle_Nro:** numeric(18,0)
 - **Calle_Piso:** numeric(18,0)
 - **Calle_Depto:** varchar(50)
 - **Nacionalidad_id:** int (FK Nacionalidad)
 - **Fecha_Nacimiento:** datetime
 - **Habilitado:** bit

- **Estado_Reserva:** Tabla con la parametría de los estados de las reservas.
 - **Id_Estado:** int IDENTITY(1,1) (PK)
 - **Descripcion:** varchar(50)

- **Reserva:** Tabla con todas las reservas hasta la fecha.
 - **Id_Reserva:** numeric(18,0) IDENTITY(1,1) (PK)
 - **Fecha_Inicio:** datetime
 - **Estadia:** int
 - **Tipo_Regimen:** int (FK Regimen)
 - **Estado:** int (FK Estado_Reserva)
 - **Id_Usuario:** int (FK Usuario)

- **Habitacion_Reserva:** Tabla con las habitaciones reservadas en cada hotel por reserva.
 - **Id_Hotel:** int (FK Habitacion)
 - **Id_Reserva:** numeric(18,0) (FK Reserva)
 - **Habitacion_Nro:** int (FK Habitacion)
 - **Habitacion_Piso:** int (FK Habitacion)

- **Reserva_Cliente:** Tabla que almacena la relación entre la reserva y el cliente que la realizo.
 - **Id_Reserva:** numeric(18,0) (FK Reserva)
 - **Id_Cliente:** int (FK Clientes)

- **Historial_Cancelacion_Reserva:** Tabla con los registros de las cancelaciones realizadas, con fecha y motivo de la misma.
 - **Id_Reserva:** numeric(18,0) (FK Reserva)
 - **Motivo:** varchar(200)
 - **Fecha_Cancelacion:** datetime
 - **Id_Usuario:** varchar(20)

- **Estadia:** Tabla con las fechas de check-in y check-out de cada reserva y con los respectivos usuarios que la llevaron a cabo.
 - **Id_Estadia:** int IDENTITY(1,1) (PK)
 - **Id_Reserva:** numeric(18,0) (FK Reserva)
 - **Check_In:** datetime
 - **Id_Usuario_Check_In:** varchar(20)
 - **Check_out:** datetime
 - **Id_Usuario_Check_Out:** varchar(20)

- **Consumible:** Tabla con todos los productos consumibles disponibles.
 - **Id_Codigo:** numeric(18,0) (PK)
 - **Descripcion:** varchar(255)
 - **Precio:** numeric(18,2)

- **Consumible_Reserva:** Tabla que almacena los gastos de consumibles por reserva.
 - **Id_Estadia:** int (FK Estadía)
 - **Id_Codigo:** numeric(18,0) (FK Consumible)
 - **Cantidad:** int
 - **Fecha:** datetime
 - **Id_Usuario:** int (FK Usuario)

- **Facturacion:** Tabla con el registro completa de la factura, sin el detalle.
 - **Id_Factura:** numeric(18,0) IDENTITY(1,1) (PK)
 - **Id_Estadia:** int (FK Estadía)
 - **Id_Cliente:** int (FK Clientes)
 - **Total_Factura:** numeric(18,2)
 - **Total_Estadia:** numeric(18,2)
 - **Total_Consumibles:** numeric(18,2)
 - **Fecha_Facturacion:** datetime

- **Facturacion_Detalle:** Tabla con los ítems de cada factura.
 - **Id_Factura:** numeric(18,0) (FK Facturación)
 - **Id_Estadia:** int (FK Estadía)
 - **Descripcion:** varchar(50)
 - **Precio:** numeric(18,2)
 - **Cantidad:** int

- **Tipo_Pago:** Parametrización de los distintos tipos de pagos que se pueden elegir a la hora de pagar la facturación.
 - **Id_Tipo_Pago:** int IDENTITY(1,1) (PK)
 - **Descripcion:** varchar(255)

- **Forma_Pago:** Tabla donde se almacenan los tipos de pagos respecto de cada factura.
 - ***Id_Factura:*** numeric(18,0) (FK Facturación)
 - ***Id_Tipo_Pago:*** int (FK Tipo_Pago)
 - ***Tarjeta_numero:*** int

- **Estadia_Cliente:** Tabla que almacena por estadía los integrantes que se registraron.
 - ***Id_Estadia:*** int (FK Estadía)
 - ***Id_Cliente:*** int (FK Clientes)

- **Estadistica**
 - ***Id_Estadistica:*** int IDENTITY(1,1)
 - ***Store_Procedure:*** varchar(100)
 - ***Descripcion:*** varchar(100)

- **Trimestre**
 - ***Id_Trimestre:*** int IDENTITY(1,1)
 - ***Fechas:*** varchar(20)
 - ***Descripcion:*** varchar(30)

StoredProcedures

- **Sp_login_check_valid_user:** verifica si el usuario ingresado se encuentra habilitado.

Parámetros: ID Usuario

- **Sp_password_check_ok:** verifica si la password ingresada corresponde al usuario.

Parámetros: ID Usuario, Password ingresada

- **Sp_password_change:** modifica la password del usuario.

Parámetros: ID Usuario, Nueva Password

- **Sp_login_check_password:** verifica si la password ingresada corresponde al usuario. En caso contrario suma un intento más a los fallidos, y si correspondiera, inhabilita al usuario.

Parámetros: ID Usuario, Password ingresada

- **Sp_rol_exist_one_by_user:** Verifica la asignación de un rol para un usuario determinado. En caso contrario, dicho rol se setea a NULL.

Parámetros: ID Usuario

- **Sp_hotel_exist_one_by_user:** Verifica la asignación de un hotel para un usuario determinado. En caso contrario, dicho hotel se setea a NULL.

Parámetros: ID Usuario, ID Rol

- **Sp_menu_list_functionality_by_user:** devuelve la lista de funcionalidades correspondientes a un rol.

Parámetros: ID Rol

- **Sp_user_search:** filtra la tabla de usuarios por nombre, rol y hotel.

Parámetros: Nombre Usuario, ID Rol, ID Hotel

- **Sp_user_enable_disable:** cambia el estado del usuario a habilitado/inhabilitado.

Parámetros: Nombre Usuario, ID Hotel, Nuevo Estado

- **Sp_user_clean_login:** reinicia los intentos fallidos al loguearse del usuario.

Parámetros: Nombre Usuario

- **Sp_user_search_rol_hotel_by_user:** busca los roles habilitados para un usuario y hotel determinados.

Parámetros: Nombre Usuario, ID Hotel

- **Sp_user_save_update:** guarda los cambios producidos en los datos del usuario.

Parámetros: Nombre Usuario, Apellido Usuario, Tipo de Identificación Usuario, Numero de identificación Usuario, Mail Usuario, Telefono Usuario, Direccion Usuario, Fecha Nacimiento Usuario, Estado Usuario, ID Hotel, ID Rol, Password Usuario

- **Sp_rol_search:** devuelve los roles que corresponden al nombre que se le pase.

Parámetros: Nombre Rol

- **Sp_rol_enable_disable:** setea el estado del rol con el nuevo estado especificado.

Parámetros: ID Rol, Nuevo estado Rol

- **Sp_rol_functionality_availability:** verifica la disponibilidad de un rol consultando con la funcionalidad del mismo.

Parámetros: ID Rol

- **Sp_rol_functionality_enabled:** verifica si un rol está habilitado consultando con la funcionalidad del mismo.

Parámetros: ID Rol

- **Sp_rol_create:** se usa para la creación de un nuevo rol.

Parámetros: Descripcion Rol

- **Sp_rol_functionality_add:** agrega una nueva funcionalidad al rol.

Parámetros: ID Rol, ID Funcionalidad

- **Sp_rol_functionality_remove:** quita una funcionalidad al rol.

Parámetros: ID Rol, ID Funcionalidad

- **Sp_client_search:** devuelve los clientes filtrados por nombre, apellido, tipo y número de identificación, y/o mail.

Parámetros: Nombre Cliente, Apellido Cliente, Tipo Identificación Cliente, Numero Identificación Cliente, Mail Cliente

- **Sp_client_enable_disable:** setea el estado del cliente con el nuevo estado especificado.

Parámetros: ID Cliente, Nuevo estado

- **Sp_client_data_get_by_id_client:** devuelve todos los datos del cliente que posea el id especificado.

Parámetros: ID Cliente

- **Sp_client_save_update:** guarda los cambios efectuados a los datos del cliente.

Parámetros: Nombre Cliente, Apellido Cliente, Tipo Identificación Cliente, Numero Identificación Cliente, Mail Cliente, Telefono Cliente, Calle Direccion Cliente, Numero Direccion Cliente, Piso Cliente, Depto Cliente, Nacionalidad Cliente, Fecha nacimiento Cliente

- **Sp_client_check_exist_document:** verifica que el tipo y número de identificación ya pertenezcan a otro cliente.

Parámetros: ID Cliente, Tipo Identificación Cliente, Numero Identificación Cliente

- **Sp_client_check_exist_mail:** verifica que el mail ya pertenezca a otro cliente.

Parámetros: ID Cliente, Mail Cliente

- **Sp_hotel_search:** filtra los hoteles por nombre, ciudad, país, cantidad de estrellas e/o id.

Parámetros: Nombre Hotel, Ciudad Hotel, Pais Hotel, Estrellas Hotel, ID Hotel Usuario

- **Sp_hotel_regimen_available:** devuelve los regímenes asignados al hotel especificado.

Parámetros: ID Hotel

- **Sp_hotel_regimen_assign:** asigna un régimen al hotel.

Parámetros: ID Hotel

- **Sp_hotel_data_get_by_id:** devuelve todos los datos pertenecientes al hotel con el id especificado.

Parámetros: ID Hotel

- **Sp_hotel_save_update:** guarda los cambios efectuados a los datos del hotel.

Parámetros: ID Usuario, ID Rol Usuario, Nombre Hotel, Mail Hotel, Calle direccion Hotel, Numero direccion Hotel, Telefono Hotel, Ciudad Hotel, Pais Hotel, Estrellas Hotel, Fecha de creación Hotel

- **Sp_hotel_regimen_add:** agrega un nuevo regimen al hotel.

Parámetros: ID Hotel, ID Regimen

- **Sp_hotel_regimen_remove:** quita un regimen al hotel.

Parámetros: ID Hotel, ID Regimen, Fecha del sistema

- **Sp_hotel_close_period_valid:** en caso que el periodo sea válido, genera el cierre del hotel, especificando el motivo.

Parámetros: ID Usuario, ID Hotel, Fecha de cierre, Fecha de reapertura, Motivo de cierre

- **Sp_habitacion_search:** filtra las habitaciones por id de la habitación, hotel, piso, tipo, frente y/o comodidades.

Parámetros: ID Habitacion, ID Hotel Habitacion, Piso Habitacion, Tipo Habitacion, Frente Habitacion, Comodidades Habitacion, Nombre Usuario

- **Sp_habitacion_close_period_valid:** en caso de que el periodo sea válido, genera el cierre de la habitación, especificando el motivo.

Parámetros: ID Usuario, ID Habitacion, Piso Habitacion, Hotel Habitacion, Fecha de cierre, Fecha de reapertura, Motivo de cierre

- **Sp_habitacion_data_get_by_id:** devuelve todos los datos pertenecientes a la habitación especificada.

Parámetros: ID Habitacion, Piso Habitacion, Hotel Habitacion

- **Sp_habitacion_exist_hotel_room:** verifica la existencia de la habitación.

Parámetros: ID Habitacion, Hotel Habitacion, Piso Habitacion

- **Sp_habitacion_save_update:** inserta o modifica los datos de la habitación.

Parámetros: ID Habitacion, Piso Habitacion, Hotel Habitacion, Tipo Habitacion, Comodidades Habitacion, Frente Habitacion

- **Sp_habitacion_person_per_room_by_booking_id:** retorna la cantidad de personas asignadas a la reserva especificada.

Parámetros: Numero Reserva

- **Sp_regimen_search:** devuelve los datos de un régimen según la descripción especificada.

Parámetros: DescripcionRegimen

- **Sp_cancelacion_reserva_search:** filtra las reservas según el número de reserva, apellido del cliente asociado a la misma o fecha de ingreso.

Parámetros: Numero Reserva, Apellido Cliente, Fecha de ingreso, ID Usuario Hotel

- **Sp_cancelacion_reserva_cancel:** cancela la reserva especificada, guardando el motivo, quien la cancela y la fecha de cancelación.

Parámetros: Numero Reserva, Motivo Cancelacion, ID Usuario, Fecha del sistema

- **Sp_estadia_booking_search:** filtra las estadias por número de reserva, hotel y/o fecha de ingreso.

Parámetros: Numero Reserva, ID Hotel, Fecha de ingreso

- **Sp_estadia_booking_is_exist:** verifica la existencia de una reserva.

Parámetros: Numero Reserva, ID Hotel

- **Sp_estadia_booking_is_cancel:** cancela la reserva especificada.

Parámetros: Numero Reserva, ID Hotel

- **Sp_estadia_booking_is_before:** verifica que la fecha de check in de la reserva sea anterior a la actual.

Parámetros: Numero Reserva, ID Hotel, Fecha del sistema

- **Sp_estadia_booking_is_hotel:** Valida la realización del booking dada una reserva para un determinado hotel.

Parámetros: Numero Reserva, ID Hotel

- **Sp_estadia_generate_stay:** lleva a cabo el proceso de check in, generando una nueva estadia.

Parámetros: Numero Reserva, Nombre Usuario, Fecha del sistema

- **Sp_estadia_cancel_is_after_date_check_in:** cancela la reserva por no show.

Parámetros: Numero Reserva, Fecha del sistema

- **Sp_estadia_is_for_check_in:** verifica la existencia de una estadia con el número de reserva especificado.

Parámetros: Numero Reserva

- **Sp_estadia_exist_full_stay:** Verifica si la estadía se completó con los días que tenía previstos por la reserva realizada.

Parámetros: Numero Reserva

- **Sp_estadia_generate_checkout:** lleva a cabo el proceso de checkout.

Parámetros: Numero Reserva, Nombre Usuario, Fecha del sistema

- **Sp_check_client_search:** filtra a los clientes habilitados por mail, tipo y número de identificación.

Parámetros: Mail Cliente, Tipo Identificacion Cliente, Numero Identificacion Cliente

- **Sp_estadia_save_stay_client:** genera la relación entre un cliente y la estadia generada.

Parámetros: ID Estadia, ID Cliente

- **Sp_consumibles_estadias_search:** busca las estadias que todavía se están desarrollando y coincidan con el hotel y el número de reserva.

Parámetros: ID Hotel, Numero Reserva

- **Sp_consumible_filter:** filtra los consumibles por la descripcion especificada.

Parámetros: Descripcion Consumible

- **Sp_consumible_by_estadia_search:** devuelve todos los consumibles pertenecientes a la estadia.

Parámetros: Numero de reserva

- **Sp_estadia_consumible_add:** agrega un consumible a la estadia.

Parámetros: ID Usuario, Numero Reserva, ID Consumible, Cantidad Consumible, Fecha del sistema

- **Sp_estadia_consumible_remove:** quita un consumible de una estadia.

Parámetros: Numero Reserva, ID Consumible, Fecha del sistema

- **Sp_facturar_estadia_booking_search:** devuelve los datos pertenecientes a la estadia.

Parámetros: Numero Reserva, ID Hotel

- **Sp_facturar_estadia_is_check_in:** verifica que la estadia tenga hecho el check in y pendiente el checkout.

Parámetros: Numero Reserva

- **Sp_facturar_estadia_is_exist:** verifica la existencia de la estadia.

Parámetros: Numero Reserva

- **Sp_facturar_estadia_was_charged:** verifica que la estadia no haya sido previamente facturada.

Parámetros: Numero Reserva

- **Sp_facturar_estadia_get_charge:** obtiene el total a pagar de la estadia.

Parámetros: Numero Reserva

- **Sp_facturar_estadia_charge:** realiza el cobro de la estadia.

Parámetros: Numero Reserva, ID Cliente, Numero Tarjeta, Tipo Pago

- **Sp_estadistic_top_5_hotel_canceled:** devuelve los 5 hoteles con mayor cantidad de cancelaciones.

Parámetros: Fecha desde, Fecha hasta

- **Sp_estadistic_top_5_hotel_consumable_charge:** devuelve los 5 hoteles que más consumibles vendieron.

Parámetros: Fecha desde, Fecha hasta

- **Sp_estadistic_top_5_hotel_more_days_out:** devuelve los hoteles que más tiempo estuvieron cerrados.

Parámetros: Fecha desde, Fecha hasta

- **Sp_estadistic_top_5_room_hotel_most_occupied:** devuelve los 5 hoteles que más fueron ocupados.

Parámetros: Fecha desde, Fecha hasta

- **Sp_estadistic_top_5_client_more_points:** devuelve los 5 clientes que más puntos obtuvieron.

Parámetros: Fecha desde, Fecha hasta

- **Sp_reserva_listar:** filtra las reservas por hotel, nombre y apellido del cliente y/o numero de reserva.

Parámetros: ID Hotel, Nombre Cliente, Apellido Cliente, Numero Reserva

- **Sp_hotel_regimen_retrieve:** devuelve los regímenes disponibles del hotel especificado.

Parámetros: ID Hotel

- **Sp_tipo_habitacion_available:** devuelve los tipos de habitación que quedan disponibles en el hotel.

Parámetros: ID Hotel

- **Sp_reserva_occupied:** devuelve la cantidad de estadias de un hotel según el tipo de habitación entre las fechas especificadas.

Parámetros: ID Hotel, Tipo Habitación, Fecha desde, Fecha hasta

- **Sp_room_type_total:** devuelve la cantidad de habitaciones de un tipo que posee el hotel.

Parámetros: ID Hotel, Tipo Habitacion

- **Sp_get_reserva:** Devuelve los datos de la reserva realizada.

Parámetros: Numero de reserva, Regimen Reserva, Fecha desde, Cantidad de días, Estado Reserva, Numero Habitacion, Piso Habitacion, Tipo Habitacion

- **Sp_check_hotel_availability:** verifico si el hotel está disponible para una determinada fecha.

Parámetros: ID Hotel, Fecha de check in, Estado Hotel

- **Sp_assign_room:** asigna una habitación a un cliente, generando la reserva y dejándola ocupada para esas fechas.

Parámetros: ID Hotel, ID Usuario, ID Cliente, ID Reserva, Fecha de check in, Estado Habitacion, Tipo Habitacion, Regimen,

- **Sp_get_hotel_by_booking:** devuelve el número de hotel que corresponde al número de reserva.

Parámetros: Numero de reserva

- **Sp_get_cliente_by_booking:** devuelve el número de cliente que corresponde al número de reserva.

Parámetros: Número de reserva

Funciones

- **Check_availability:** verifico si la habitación está disponible para una determinada fecha.

Parámetros: ID Hotel, Piso Habitacion, ID Habitacion, Fecha de check in, Estado Habitacion

Aplicación C#

Para la aplicación de escritorio, se utilizó una orientación a objetos mapeando las entidades de la base de datos con clases de C#. Esto fue necesario por las siguientes razones:

- Debido a que las clases en C# pueden ser utilizadas como objetos, se permite la declaración de propiedades dentro de cualquier clase.
- Se admite la herencia. Es decir, una clase que deriva de otra, contiene de forma automática todos sus atributos.
- Pueden definirse con varios parámetros de tipo.
- Las clases pueden ser declaradas como static, eso quiere decir que solo tendrá miembros estáticos y dichos miembros son accesibles a través del nombre de clase.
- Por último, las clases pueden anidarse dentro de otra clase.

Los problemas que se pueden tener al utilizar las clases, no generaron inconvenientes en la realización del trabajo práctico, por ende, la utilización correcta de las mismas, proporcionó el eficiente funcionamiento del trabajo.