

Escuela de Computación de Alajuela

Ingeniería en Computación

*IC-4301 Bases de Datos I -  
Grupo 90*

Sistema de Gestión Hotelera

Profesor: Cristian Campos Agüero

Aaron Jose Segura Alfaro

2023800052

aarsegura@estudiantec.cr

Angel Nah'un Araya Trejos

2024083241

a.araya.4@estudiantec.cr

Alajuela, Costa Rica

Enero 2025

<b>Introducción</b>	<b>4</b>
<b>Requerimientos</b>	<b>5</b>
Requerimientos Funcionales (RF)	5
Módulo A – Hospedaje (establecimientos)	5
Módulo B – Habitaciones	5
Módulo C – Clientes	5
Módulo D – Reservaciones	6
Módulo E – Facturación	6
Módulo F – Recreación (empresas y actividades)	6
Requerimientos No Funcionales (RNF) — Categorizados	7
Seguridad y privacidad	7
Disponibilidad y confiabilidad	7
Usabilidad	7
Rendimiento	7
Compatibilidad / plataforma	7
<b>Diagrama Entidad-Relación</b>	<b>8</b>
Diagrama de Recreación	8
Tipo de habitación	9
Apartado del Hotel	9
Apartado del cliente	10
<b>Diseño de la base de datos, descripción de las tablas, atributos y restricciones</b>	<b>11</b>
<b>Diccionario de la base de datos</b>	<b>19</b>
Sistema de Gestión Hotelera	19
1. Tabla: Hotel	19
2. Tabla: DireccionHotel	19
3. Tabla: TelefonoHotel	20
4. Tabla: ServicioHotel	20
5. Tabla: HotelServicio	20
6. Tabla: RedSocial	20
7. Tabla: HotelRedSocial	21
Habitaciones	21
8. Tabla: TipoHabitacion	21
9. Tabla: Comodidad	21
10. Tabla: TipoHabitacionComodidad	22
11. Tabla: FotoTipoHabitacion	22
12. Tabla: Habitacion	22
Clientes	23
13. Tabla: Cliente	23
14. Tabla: TelefonoCliente	23
15. Tabla: DireccionClienteCR	23
Reservación y Facturación	24
16. Tabla: Reservacion	24
17. Tabla: Factura	25
Recreación	25

18. Tabla: EmpresaRecreacion	25
19. Tabla: ActividadRecreacion	26
20. Tabla: TipoActividad	26
21. Tabla: ActividadTipoActividad	26
<b>Diagrama relacional de la base de datos</b>	<b>27</b>
Lado Derecho con Hotel, cliente y reservación	27
Lado izquierdo con hotel - Recreacion - habitación	28
<b>Prototipos de la aplicación con enlace público</b>	<b>29</b>
Inicio de sesión	29
Apartado de Crear usuario	29
Apartado para registrarse como anfitrión	30
Apartado para indicar el tipo de lugar	30
Apartado para indicar formas de contactar	31
Forma de agregar habitaciones	31
Pagina inicial - vista usuario	32
Apartado de habitaciones o lugares	32
Reservaciones	33
Facturas	34
Vista de Anfitrión con búsqueda por filtros	35
Filtro	35
<b>Análisis de resultados</b>	<b>36</b>
<b>Bibliografía</b>	<b>37</b>

# Introducción

Este documento constituye la documentación técnica completa para el proyecto "Sistema de Gestión Turística Digital para la Municipalidad de Limón". El objetivo central es presentar el análisis, diseño y propuesta de solución para una plataforma digital que permita centralizar y promover la oferta de servicios turísticos de las pequeñas y medianas empresas de la región.

La presente entrega se estructura de la siguiente manera:

Una investigación y análisis en donde parte de un análisis de los requerimientos del problema, desglosando las necesidades funcionales del sistema para hospedaje, gestión de clientes, reservaciones, facturación y actividades de recreación.

Se presenta una solución arquitectónica basada en una base de datos relacional, detallando el diseño lógico y físico de la información.

Esta va a tener un diseño conceptual en donde se incluye el diagrama Entidad-Relación (DER) que modela las entidades, sus atributos y las relaciones fundamentales del negocio.

Se muestra la implementación concreta del modelo de datos mediante

- Un Diagrama Relacional detallado.
- La descripción completa de tablas, incluyendo atributos, tipos de datos, restricciones y claves.
- El script SQL para la creación de todos los objetos en SQL Server.

Se presentan los prototipos de la aplicación que visualiza el flujo y las pantallas principales del sistema (CRUD: Crear, Leer, Actualizar, Eliminar), junto con un enlace público para su revisión interactiva.

Se culmina con un análisis de resultados que valida el diseño propuesto frente a los requerimientos iniciales y se listan las referencias consultadas.

# Requerimientos

## Requerimientos Funcionales (RF)

### Módulo A – Hospedaje (establecimientos)

RF01. El sistema debe permitir registrar un establecimiento de hospedaje con: nombre, cédula jurídica única, tipo, dirección completa, GPS, 2 teléfonos con código país, correo, URL de sitio web (opcional), redes sociales y lista de servicios del establecimiento.

RF02. El sistema debe validar que la cédula jurídica del establecimiento sea única (no se permite duplicar).

RF03. El sistema debe permitir modificar los datos de un establecimiento registrado.

RF04. El sistema debe permitir eliminar un establecimiento registrado (con manejo de dependencias: habitaciones, reservas, etc.).

RF05. El sistema debe permitir listar los establecimientos registrados y ver su detalle.

### Módulo B – Habitaciones

RF06. El sistema debe permitir registrar tipos de habitación con: nombre, descripción, tipo de cama, lista de comodidades, precio y fotos.

RF07. El sistema debe permitir registrar habitaciones con: número, tipo de habitación y estado (activo/inactivo) asociadas a un establecimiento.

RF08. El sistema debe permitir modificar, eliminar y listar tipos de habitación.

RF09. El sistema debe permitir modificar, eliminar y listar habitaciones.

### Módulo C – Clientes

RF10. El sistema debe permitir registrar clientes con: nombre, apellidos, fecha de nacimiento, tipo de identificación, identificación única, país de residencia, dirección (si es costarricense), 2 teléfonos y correo.

RF11. El sistema debe validar que la identificación del cliente sea única.

RF12. El sistema debe permitir modificar, eliminar y listar clientes.

## Módulo D – Reservaciones

RF13. El sistema debe permitir crear una reservación indicando: identificación del cliente, habitación, fecha y hora de ingreso, cantidad de personas, si posee vehículo (sí/no) y fecha de salida.

RF14. El sistema debe validar que la habitación esté activa al momento de reservar.

RF15. El sistema debe permitir modificar, cancelar/eliminar y listar reservaciones.

## Módulo E – Facturación

RF16. El sistema debe permitir crear una factura a partir del número de reserva registrando: cliente, cargos de habitación, número de noches e importe total.

RF17. El sistema debe registrar en la factura fecha y hora (hora:minuto) del momento de creación.

RF18. El sistema debe permitir registrar el método de pago: efectivo o tarjeta de crédito.

RF19. El sistema debe permitir listar facturas y consultar detalle por número de factura o número de reserva.

## Módulo F – Recreación (empresas y actividades)

RF20. El sistema debe permitir registrar empresas de recreación con: nombre, cédula jurídica, correo, teléfono, persona de contacto, dirección, tipo(s) de actividad, tipos de servicios, descripción y precio.

RF21. El sistema debe permitir modificar, eliminar y listar empresas/actividades de recreación.

RF22. El sistema debe permitir que el cliente busque empresas de recreación usando filtros (ejemplo estilo Airbnb).

## Requerimientos No Funcionales (RNF) — Categorizados

### Seguridad y privacidad

RNF-S01. El sistema debe proteger la información personal de clientes y contactos (accesos controlados).

RNF-S02. El sistema debe aplicar integridad de datos: cédulas jurídicas e identificaciones de clientes no duplicadas.

### Disponibilidad y confiabilidad

RNF-D01. El sistema debe garantizar la persistencia de datos en SQL Server (no se pierde información al cerrar).

RNF-D02. El sistema debe mantener consistencia referencial (no permitir registros huérfanos por FKs).

### Usabilidad

RNF-U01. La interfaz debe ser clara y consistente para operaciones CRUD: agregar, modificar, eliminar y listar.

RNF-U02. La búsqueda de recreación debe permitir filtros múltiples y mostrar resultados ordenados de forma entendible.

### Rendimiento

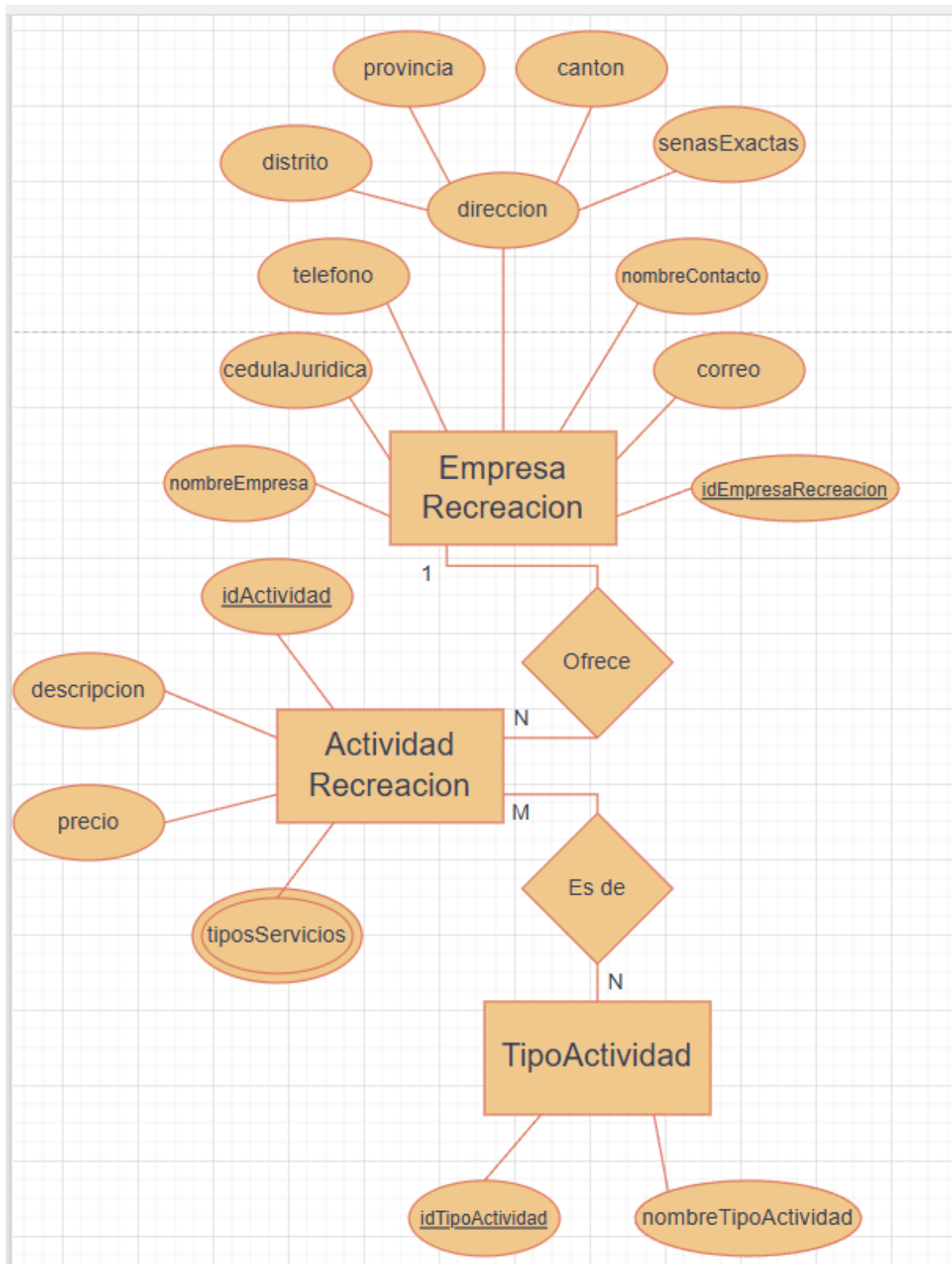
RNF-R01. El sistema debe responder a consultas de listado y búsqueda en un tiempo de 3 segundos en condiciones normales.

### Compatibilidad / plataforma

RNF-C01. La base de datos debe implementarse en SQL Server usando un script de creación completo (tablas, relaciones y restricciones).

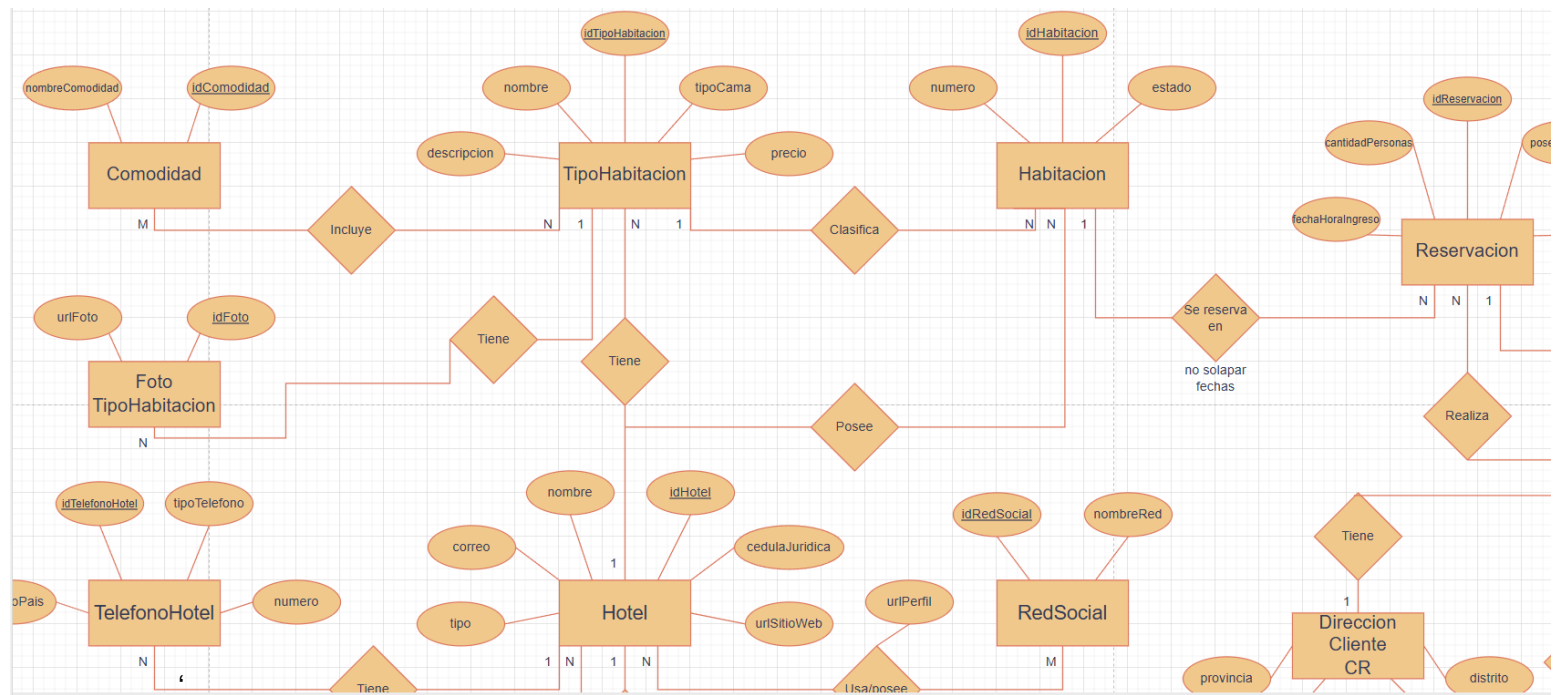
# Diagrama Entidad-Relación

## Diagrama de Recreación

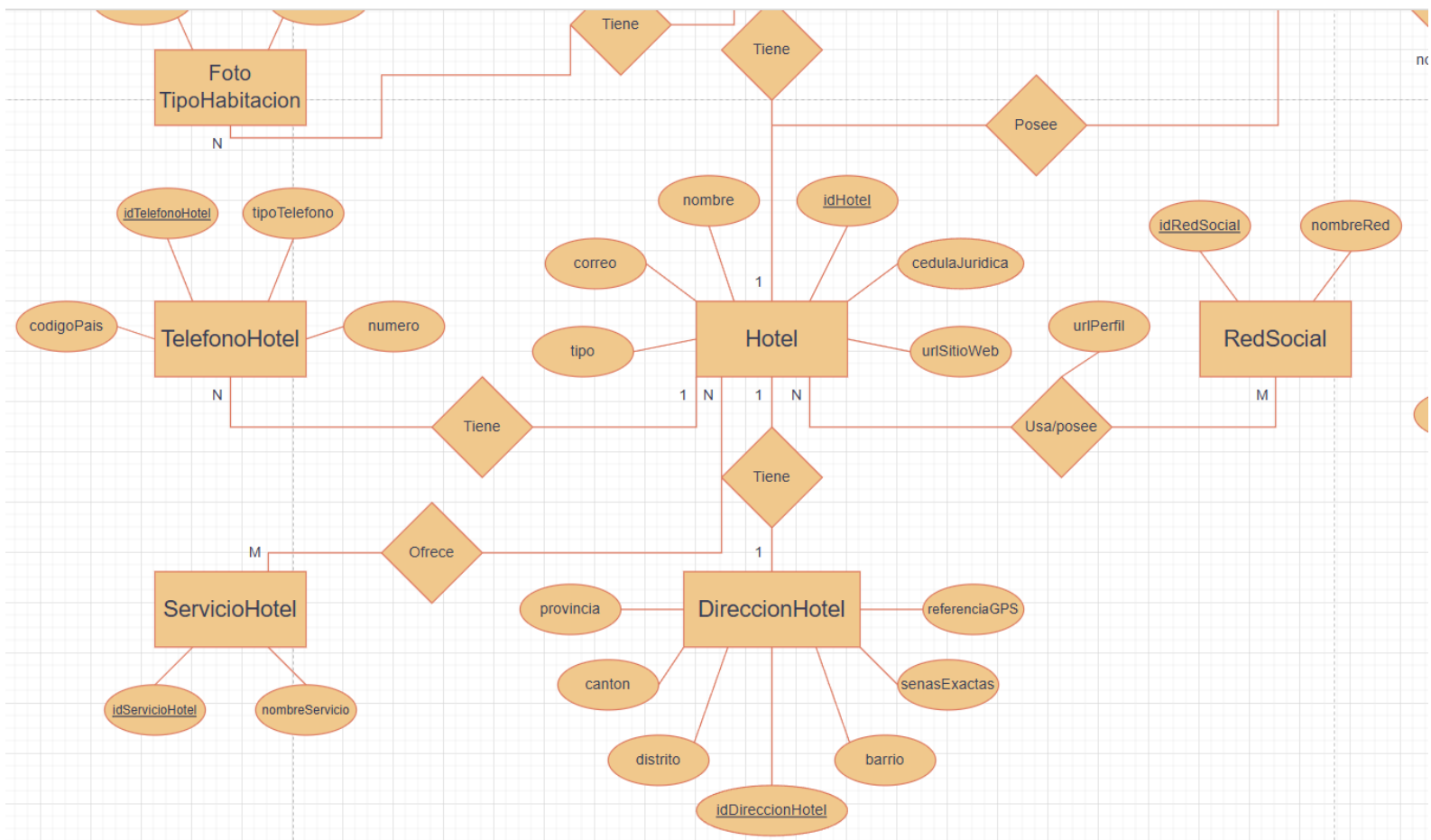




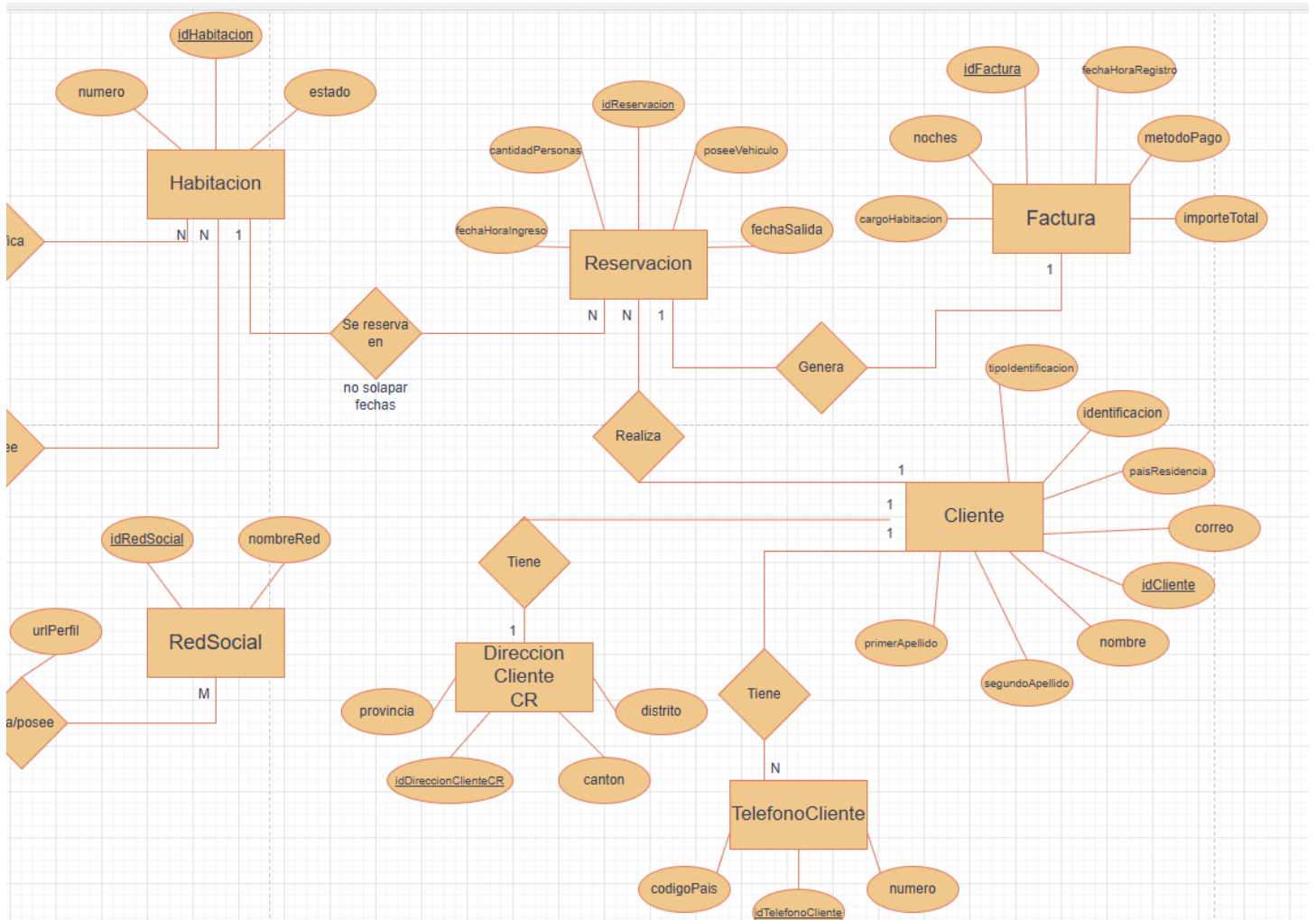
## Tipo de habitación



## Apartado del Hotel



## Apartado del cliente



# Diseño de la base de datos, descripción de las tablas, atributos y restricciones

```
CREATE DATABASE PROYE1;
```

```
USE PROYE1;  
GO
```

```
CREATE TABLE dbo.Hotel (  
    idHotel          INT IDENTITY(1,1) PRIMARY KEY,  
    nombre           NVARCHAR(120) NOT NULL,  
    cedulaJuridica   NVARCHAR(30) NOT NULL,  
    tipo             NVARCHAR(30) NOT NULL,  
    correo           NVARCHAR(120) NOT NULL,  
    urlSitioWeb      NVARCHAR(200) NULL  
);
```

```
ALTER TABLE dbo.Hotel  
ADD CONSTRAINT UQ_Hotel_Cedula UNIQUE (cedulaJuridica);
```

```
ALTER TABLE dbo.Hotel  
ADD CONSTRAINT CK_Hotel_Tipo CHECK (tipo IN  
( 'Hotel', 'Hostal', 'Casa', 'Departamento', 'Cuarto compartido', 'Cabaña' ));  
GO
```

```
CREATE TABLE dbo.DireccionHotel (  
    idDireccionHotel INT IDENTITY(1,1) PRIMARY KEY,  
    idHotel          INT NOT NULL UNIQUE, -- 1 a 1  
    provincia        NVARCHAR(60) NOT NULL,  
    canton           NVARCHAR(60) NOT NULL,  
    distrito         NVARCHAR(60) NOT NULL,  
    barrio           NVARCHAR(60) NULL,  
    senasExactas     NVARCHAR(300) NOT NULL,  
    referenciaGPS     NVARCHAR(120) NULL,  
  
    CONSTRAINT FK_DirHotel_Hotel  
        FOREIGN KEY (idHotel) REFERENCES dbo.Hotel(idHotel)  
        ON DELETE CASCADE ON UPDATE CASCADE  
);  
GO
```

```

CREATE TABLE dbo.TelefonoHotel (
    idTelefonoHotel INT IDENTITY(1,1) PRIMARY KEY,
    idHotel          INT NOT NULL,
    tipoTelefono     NVARCHAR(20) NULL, -- Principal/Secundario (opcional)
    codigoPais       NVARCHAR(8) NOT NULL, -- ej: +506
    numero           NVARCHAR(20) NOT NULL,

    CONSTRAINT FK_TelHotel_Hotel
        FOREIGN KEY (idHotel) REFERENCES dbo.Hotel(idHotel)
        ON DELETE CASCADE ON UPDATE CASCADE
);
GO

```

```

CREATE TABLE dbo.RedSocial (
    idRedSocial      INT IDENTITY(1,1) PRIMARY KEY,
    nombreRed        NVARCHAR(30) NOT NULL UNIQUE
);
GO

```

```

CREATE TABLE dbo.HotelRedSocial (
    idHotel          INT NOT NULL,
    idRedSocial      INT NOT NULL,
    urlPerfil        NVARCHAR(250) NOT NULL,

    CONSTRAINT PK_HotelRedSocial PRIMARY KEY (idHotel, idRedSocial),
    CONSTRAINT FK_HotelRedSocial_Hotel FOREIGN KEY (idHotel)
        REFERENCES dbo.Hotel(idHotel) ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT FK_HotelRedSocial_Red FOREIGN KEY (idRedSocial)
        REFERENCES dbo.RedSocial(idRedSocial) ON DELETE NO ACTION ON UPDATE
        CASCADE
);
GO

```

```

CREATE TABLE dbo.ServicioHotel (
    idServicioHotel INT IDENTITY(1,1) PRIMARY KEY,
    nombreServicio  NVARCHAR(60) NOT NULL UNIQUE
);
GO

```

```

CREATE TABLE dbo.HotelServicio (
    idHotel          INT NOT NULL,
    idServicioHotel  INT NOT NULL,

    CONSTRAINT PK_HotelServicio PRIMARY KEY (idHotel, idServicioHotel),
    CONSTRAINT FK_HotelServicio_Hotel FOREIGN KEY (idHotel)
        REFERENCES dbo.Hotel(idHotel) ON DELETE CASCADE ON UPDATE CASCADE,
    CONSTRAINT FK_HotelServicio_Serv FOREIGN KEY (idServicioHotel)
        REFERENCES dbo.ServicioHotel(idServicioHotel) ON DELETE NO ACTION ON
UPDATE CASCADE
);
GO

```

```

CREATE TABLE dbo.TipoHabitacion (
    idTipoHabitacion INT IDENTITY(1,1) PRIMARY KEY,
    idHotel          INT NOT NULL,
    nombre           NVARCHAR(80) NOT NULL,
    descripcion      NVARCHAR(300) NULL,
    tipoCama         NVARCHAR(20) NOT NULL,
    precio           DECIMAL(10,2) NOT NULL,

    CONSTRAINT FK_TipoHab_Hotel
        FOREIGN KEY (idHotel) REFERENCES dbo.Hotel(idHotel)
        ON DELETE CASCADE ON UPDATE CASCADE
);

```

```

ALTER TABLE dbo.TipoHabitacion
ADD CONSTRAINT CK_TipoHab_TipoCama CHECK (tipoCama IN
('Individual','Queen','King'));

```

```

ALTER TABLE dbo.TipoHabitacion
ADD CONSTRAINT CK_TipoHab_Precio CHECK (precio >= 0);
GO

```

```

CREATE TABLE dbo.Comodidad (
    idComodidad      INT IDENTITY(1,1) PRIMARY KEY,
    nombreComodidad  NVARCHAR(60) NOT NULL UNIQUE
);
GO

```

```

CREATE TABLE dbo.TipoHabitacionComodidad (
    idTipoHabitacion INT NOT NULL,
    idComodidad INT NOT NULL,

    CONSTRAINT PK_TipoHabCom PRIMARY KEY (idTipoHabitacion, idComodidad),
    CONSTRAINT FK_TipoHabCom_TipoHab FOREIGN KEY (idTipoHabitacion)
        REFERENCES dbo.TipoHabitacion(idTipoHabitacion) ON DELETE CASCADE ON
UPDATE CASCADE,
    CONSTRAINT FK_TipoHabCom_Comod FOREIGN KEY (idComodidad)
        REFERENCES dbo.Comodidad(idComodidad) ON DELETE NO ACTION ON UPDATE
CASCADE
);
GO

```

```

CREATE TABLE dbo.FotoTipoHabitacion (
    idFoto INT IDENTITY(1,1) PRIMARY KEY,
    idTipoHabitacion INT NOT NULL,
    urlFoto NVARCHAR(250) NOT NULL,

    CONSTRAINT FK_FotoTH_TipoHab FOREIGN KEY (idTipoHabitacion)
        REFERENCES dbo.TipoHabitacion(idTipoHabitacion) ON DELETE CASCADE ON
UPDATE CASCADE
);
GO

```

```

CREATE TABLE dbo.Habitacion (
    idHabitacion INT IDENTITY(1,1) PRIMARY KEY,
    idHotel INT NOT NULL,
    idTipoHabitacion INT NOT NULL,
    numero NVARCHAR(15) NOT NULL,
    estado NVARCHAR(10) NOT NULL,

    CONSTRAINT FK_Hab_Hotel FOREIGN KEY (idHotel)
        REFERENCES dbo.Hotel(idHotel) ON DELETE CASCADE ON UPDATE CASCADE,

    CONSTRAINT FK_Hab_TipoHab FOREIGN KEY (idTipoHabitacion)
        REFERENCES dbo.TipoHabitacion(idTipoHabitacion) ON DELETE NO ACTION ON
UPDATE CASCADE
);

```

```

ALTER TABLE dbo.Habitacion
ADD CONSTRAINT CK_Hab_Estado CHECK (estado IN ('Activo','Inactivo'));

```

```

-- No repetir numero de habitacin dentro del mismo hotel
ALTER TABLE dbo.Habitacion
ADD CONSTRAINT UQ_Hab_HotelNumero UNIQUE (idHotel, numero);
GO

```

```

CREATE TABLE dbo.Cliente (
    idCliente      INT IDENTITY(1,1) PRIMARY KEY,
    nombre         NVARCHAR(60) NOT NULL,
    primerApellido NVARCHAR(60) NOT NULL,
    segundoApellido NVARCHAR(60) NOT NULL,
    fechaNacimiento DATE NOT NULL,
    tipoidentificacion NVARCHAR(20) NOT NULL,
    identificacion  NVARCHAR(30) NOT NULL,
    paisResidencia NVARCHAR(60) NOT NULL,
    correo         NVARCHAR(120) NOT NULL
);

```

```

ALTER TABLE dbo.Cliente
ADD CONSTRAINT UQ_Cliente_Identificacion UNIQUE (identificacion);

```

```

ALTER TABLE dbo.Cliente
ADD CONSTRAINT CK_Cliente_Tipoid CHECK (tipoidentificacion IN
('Pasaporte','Dimex','Cédula nacional','Otro'));
GO

```

```

CREATE TABLE dbo.DireccionClienteCR (
    idDireccionClienteCR INT IDENTITY(1,1) PRIMARY KEY,
    idCliente            INT NOT NULL UNIQUE, -- 0..1
    provincia            NVARCHAR(60) NOT NULL,
    canton               NVARCHAR(60) NOT NULL,
    distrito             NVARCHAR(60) NOT NULL,

    CONSTRAINT FK_DirCliente_Cliente FOREIGN KEY (idCliente)
        REFERENCES dbo.Cliente(idCliente) ON DELETE CASCADE ON UPDATE
        CASCADE
);
GO

```

```

CREATE TABLE dbo.TelefonoCliente (
    idTelefonoCliente INT IDENTITY(1,1) PRIMARY KEY,
    idCliente          INT NOT NULL,
    codigoPais         NVARCHAR(8) NOT NULL,
    numero             NVARCHAR(20) NOT NULL,

    CONSTRAINT FK_TelCliente_Cliente FOREIGN KEY (idCliente)
        REFERENCES dbo.Cliente(idCliente) ON DELETE CASCADE ON UPDATE
        CASCADE
);
GO

```

```

CREATE TABLE dbo.Reservacion (
    idReservacion INT IDENTITY(1,1) PRIMARY KEY,
    idCliente INT NOT NULL,
    idHabitacion INT NOT NULL,
    fechaHoraIngreso DATETIME2(0) NOT NULL,
    fechaSalida DATE NOT NULL,
    cantidadPersonas INT NOT NULL,
    poseeVehiculo BIT NOT NULL,

    CONSTRAINT FK_Res_Cliente FOREIGN KEY (idCliente)
        REFERENCES dbo.Cliente(idCliente) ON DELETE NO ACTION ON UPDATE
        CASCADE,

    CONSTRAINT FK_Res_Habitacion FOREIGN KEY (idHabitacion)
        REFERENCES dbo.Habitacion(idHabitacion) ON DELETE NO ACTION ON UPDATE
        CASCADE
);

```

```

ALTER TABLE dbo.Reservacion
ADD CONSTRAINT CK_Res_CantPersonas CHECK (cantidadPersonas > 0);
GO

```

```

CREATE TABLE dbo.Factura (
    idFactura INT IDENTITY(1,1) PRIMARY KEY,
    idReservacion INT NOT NULL,
    fechaHoraRegistro DATETIME2(0) NOT NULL,
    noches INT NOT NULL,
    cargoHabitacion DECIMAL(10,2) NOT NULL,
    importeTotal DECIMAL(10,2) NOT NULL,
    metodoPago NVARCHAR(20) NOT NULL,

    CONSTRAINT FK_Fac_Res FOREIGN KEY (idReservacion)
        REFERENCES dbo.Reservacion(idReservacion) ON DELETE NO ACTION ON
        UPDATE CASCADE
);

```

-- 1 a 1: una factura por reservacion

```

ALTER TABLE dbo.Factura
ADD CONSTRAINT UQ_Factura_Reservacion UNIQUE (idReservacion);

```

```

ALTER TABLE dbo.Factura
ADD CONSTRAINT CK_Factura_MetodoPago CHECK (metodoPago IN
('Efectivo','Tarjeta'));

```

```

ALTER TABLE dbo.Factura
ADD CONSTRAINT CK_Factura_Montos CHECK (cargoHabitacion >= 0 AND importeTotal
>= 0);

```



```
ALTER TABLE dbo.Factura
ADD CONSTRAINT CK_Factura_Noches CHECK (noches > 0);
GO
```

```
CREATE TABLE dbo.EmpresaRecreacion (
    idEmpresaRecreacion INT IDENTITY(1,1) PRIMARY KEY,
    nombreEmpresa NVARCHAR(120) NOT NULL,
    cedulaJuridica NVARCHAR(30) NOT NULL,
    correo NVARCHAR(120) NOT NULL,
    telefono NVARCHAR(20) NOT NULL,
    nombreContacto NVARCHAR(120) NOT NULL,
    provincia NVARCHAR(60) NOT NULL,
    canton NVARCHAR(60) NOT NULL,
    distrito NVARCHAR(60) NOT NULL,
    senasExactas NVARCHAR(300) NOT NULL
);
```

```
ALTER TABLE dbo.EmpresaRecreacion
ADD CONSTRAINT UQ_EmpRec_Cedula UNIQUE (cedulaJuridica);
GO
```

```
CREATE TABLE dbo.ActividadRecreacion (
    idActividad INT IDENTITY(1,1) PRIMARY KEY,
    idEmpresaRecreacion INT NOT NULL,
    tiposServicios NVARCHAR(200) NULL,
    descripcion NVARCHAR(400) NOT NULL,
    precio DECIMAL(10,2) NOT NULL,

    CONSTRAINT FK_ActRec_Emp FOREIGN KEY (idEmpresaRecreacion)
        REFERENCES dbo.EmpresaRecreacion(idEmpresaRecreacion)
        ON DELETE CASCADE ON UPDATE CASCADE
);
```

```
ALTER TABLE dbo.ActividadRecreacion
ADD CONSTRAINT CK_ActRec_Precio CHECK (precio >= 0);
GO
```

```
CREATE TABLE dbo.TipoActividad (
    idTipoActividad INT IDENTITY(1,1) PRIMARY KEY,
    nombreTipoActividad NVARCHAR(60) NOT NULL UNIQUE
);
```

```
ALTER TABLE dbo.TipoActividad
ADD CONSTRAINT CK_TipoActividad_Nombre CHECK (nombreTipoActividad IN
('Tour en bote','Tour en lancha','Tour en catamarán','Kayak','Transporte'));
GO
```

```
CREATE TABLE dbo.ActividadTipoActividad (  
    idActividad    INT NOT NULL,  
    idTipoActividad INT NOT NULL,  
  
    CONSTRAINT PK_Act_TipoAct PRIMARY KEY (idActividad, idTipoActividad),  
  
    CONSTRAINT FK_ActTipo_Act FOREIGN KEY (idActividad)  
        REFERENCES dbo.ActividadRecreacion(idActividad) ON DELETE CASCADE ON  
UPDATE CASCADE,  
  
    CONSTRAINT FK_ActTipo_Tipo FOREIGN KEY (idTipoActividad)  
        REFERENCES dbo.TipoActividad(idTipoActividad) ON DELETE NO ACTION ON  
UPDATE CASCADE  
);  
GO
```

```
INSERT INTO dbo.RedSocial(nombreRed)  
VALUES ('Facebook'),('Instagram'),('YouTube'),('TikTok'),('Airbnb'),('Threads'),('X');
```

```
INSERT INTO dbo.TipoActividad(nombreTipoActividad)  
VALUES ('Tour en bote'),('Tour en lancha'),('Tour en catamarán'),('Kayak'),('Transporte');
```

```
-- Servicios del hotel  
INSERT INTO dbo.ServicioHotel(nombreServicio)  
VALUES ('Piscina'),('Wifi'),('Parqueo'),('Restaurante'),('Bar'),('Ranchos');
```

```
-- Comodidades  
INSERT INTO dbo.Comodidad(nombreComodidad)  
VALUES ('Wifi de habitación'),('A/C'),('Ventilador'),('Agua caliente');  
GO
```

# Diccionario de la base de datos

## Sistema de Gestión Hotelera

### 1. Tabla: Hotel

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idHotel	INT	Sí	No	No	Identificador único del hotel
nombre	NVARCHAR(120)	No	No	No	Nombre del establecimiento
cedulaJuridica	NVARCHAR(30)	No	No	No	Cédula jurídica única del hotel
tipo	NVARCHAR(30)	No	No	No	Tipo de hospedaje (Hotel, Hostal, Casa, etc.)
correo	NVARCHAR(120)	No	No	No	Correo electrónico del hotel
urlSitioWeb	NVARCHAR(200)	No	No	Sí	Sitio web del hotel (opcional)

### 2. Tabla: DireccionHotel

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idDireccionHotel	INT	Sí	No	No	Identificador de la dirección del hotel
idHotel	INT	No	Sí	No	Hotel al que pertenece la dirección
provincia	NVARCHAR(60)	No	No	No	Provincia del hotel
canton	NVARCHAR(60)	No	No	No	Cantón del hotel
distrito	NVARCHAR(60)	No	No	No	Distrito del hotel
barrio	NVARCHAR(60)	No	No	Sí	Barrio del hotel
senasExactas	NVARCHAR(300)	No	No	No	Señas exactas
referenciaGPS	NVARCHAR(120)	No	No	Sí	Referencia GPS

### 3. Tabla: TelefonoHotel

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idTelefonoHotel	INT	Sí	No	No	Identificador del teléfono
idHotel	INT	No	Sí	No	Hotel propietario del teléfono
tipoTelefono	NVARCHAR(20)	No	No	Sí	Tipo de teléfono (Principal/Secundario)
codigoPais	NVARCHAR(8)	No	No	No	Código del país
numero	NVARCHAR(20)	No	No	No	Número telefónico

### 4. Tabla: ServicioHotel

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idServicioHotel	INT	Sí	No	No	Identificador del servicio
nombreServicio	NVARCHAR(60)	No	No	No	Nombre del servicio (wifi, piscina, etc.)

### 5. Tabla: HotelServicio

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idHotel	INT	Sí	Sí	No	Hotel que ofrece el servicio
idServicioHotel	INT	Sí	Sí	No	Servicio ofrecido por el hotel

### 6. Tabla: RedSocial

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idRedSocial	INT	Sí	No	No	Identificador de la red social
nombreRed	NVARCHAR(30)	No	No	No	Nombre de la red social

## 7. Tabla: HotelRedSocial

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idHotel	INT	Sí	Sí	No	Hotel asociado
idRedSocial	INT	Sí	Sí	No	Red social asociada
urlPerfil	NVARCHAR(250)	No	No	No	URL del perfil del hotel

## Habitaciones

### 8. Tabla: TipoHabitacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idTipoHabitacion	INT	Sí	No	No	Identificador del tipo de habitación
idHotel	INT	No	Sí	No	Hotel al que pertenece
nombre	NVARCHAR(80)	No	No	No	Nombre del tipo de habitación
descripcion	NVARCHAR(300)	No	No	Sí	Descripción del tipo
tipoCama	NVARCHAR(20)	No	No	No	Tipo de cama
precio	DECIMAL(10,2)	No	No	No	Precio por noche

### 9. Tabla: Comodidad

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idComodidad	INT	Sí	No	No	Identificador de la comodidad
nombreComodidad	NVARCHAR(60)	No	No	No	Nombre de la comodidad

### 10. Tabla: TipoHabitacionComodidad

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idTipoHabitacion	INT	Sí	Sí	No	Tipo de habitación
idComodidad	INT	Sí	Sí	No	Comodidad asociada

### 11. Tabla: FotoTipoHabitacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idFoto	INT	Sí	No	No	Identificador de la foto
idTipoHabitacion	INT	No	Sí	No	Tipo de habitación asociado
urlFoto	NVARCHAR(250)	No	No	No	URL o ruta de la foto

### 12. Tabla: Habitacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idHabitacion	INT	Sí	No	No	Identificador de la habitación
idHotel	INT	No	Sí	No	Hotel propietario
idTipoHabitacion	INT	No	Sí	No	Tipo de habitación
numero	NVARCHAR(15)	No	No	No	Número de habitación
estado	NVARCHAR(10)	No	No	No	Estado (Activo/Inactivo)

## Cientes

13. Tabla: Cliente

<b>Campo</b>	<b>Tipo de dato</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Descripción</b>
idCliente	INT	Sí	No	No	Identificador del cliente
nombre	NVARCHAR(60)	No	No	No	Nombre del cliente
primerApellido	NVARCHAR(60)	No	No	No	Primer apellido
segundoApellido	NVARCHAR(60)	No	No	No	Segundo apellido
fechaNacimiento	DATE	No	No	No	Fecha de nacimiento
tipoidentificacion	NVARCHAR(20)	No	No	No	Tipo de identificación
identificacion	NVARCHAR(30)	No	No	No	Identificación única
paisResidencia	NVARCHAR(60)	No	No	No	País de residencia
correo	NVARCHAR(120)	No	No	No	Correo electrónico

14. Tabla: TelefonoCliente

<b>Campo</b>	<b>Tipo de dato</b>	<b>PK</b>	<b>FK</b>	<b>Nulo</b>	<b>Descripción</b>
idTelefonoCliente	INT	Sí	No	No	Identificador del teléfono
idCliente	INT	No	Sí	No	Cliente propietario
codigoPais	NVARCHAR(8)	No	No	No	Código del país
numero	NVARCHAR(20)	No	No	No	Número telefónico

### 15. Tabla: DireccionClienteCR

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idDireccionClienteCR	INT	Sí	No	No	Identificador de la dirección
idCliente	INT	No	Sí	No	Cliente asociado
provincia	NVARCHAR(60)	No	No	No	Provincia
canton	NVARCHAR(60)	No	No	No	Cantón
distrito	NVARCHAR(60)	No	No	No	Distrito

## Reservación y Facturación

### 16. Tabla: Reservacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idReservacion	INT	Sí	No	No	Identificador de la reservación
idCliente	INT	No	Sí	No	Cliente que reserva
idHabitacion	INT	No	Sí	No	Habitación reservada
fechaHoraIngreso	DATETIME	No	No	No	Fecha y hora de ingreso
fechaSalida	DATE	No	No	No	Fecha de salida
cantidadPersonas	INT	No	No	No	Cantidad de personas
poseeVehiculo	BIT	No	No	No	Indica si posee vehículo



## 17. Tabla: Factura

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idFactura	INT	Sí	No	No	Identificador de factura
idReservacion	INT	No	Sí	No	Reservación asociada
fechaHoraRegistro	DATETIME	No	No	No	Fecha y hora de emisión
noches	INT	No	No	No	Número de noches
cargoHabitacion	DECIMAL(10,2)	No	No	No	Cargo por habitación
importeTotal	DECIMAL(10,2)	No	No	No	Total a pagar
metodoPago	NVARCHAR(20)	No	No	No	Método de pago

## Recreación

### 18. Tabla: EmpresaRecreacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idEmpresaRecreacion	INT	Sí	No	No	Identificador de empresa
nombreEmpresa	NVARCHAR(120)	No	No	No	Nombre de la empresa
cedulaJuridica	NVARCHAR(30)	No	No	No	Cédula jurídica
correo	NVARCHAR(120)	No	No	No	Correo electrónico
telefono	NVARCHAR(20)	No	No	No	Teléfono
nombreContacto	NVARCHAR(120)	No	No	No	Persona de contacto
provincia	NVARCHAR(60)	No	No	No	Provincia
canton	NVARCHAR(60)	No	No	No	Cantón
distrito	NVARCHAR(60)	No	No	No	Distrito
senasExactas	NVARCHAR(300)	No	No	No	Señas exactas

19. Tabla: ActividadRecreacion

Campo	Tipo de dato	PK	FK	Nulo	Descripción
idActividad	INT	Sí	No	No	Identificador de actividad
idEmpresaRecreacion	INT	No	Sí	No	Empresa asociada
tiposServicios	NVARCHAR(200)	No	No	Sí	Servicios incluidos
descripcion	NVARCHAR(400)	No	No	No	Descripción de la actividad
precio	DECIMAL(10,2)	No	No	No	Precio de la actividad

20. Tabla: TipoActividad

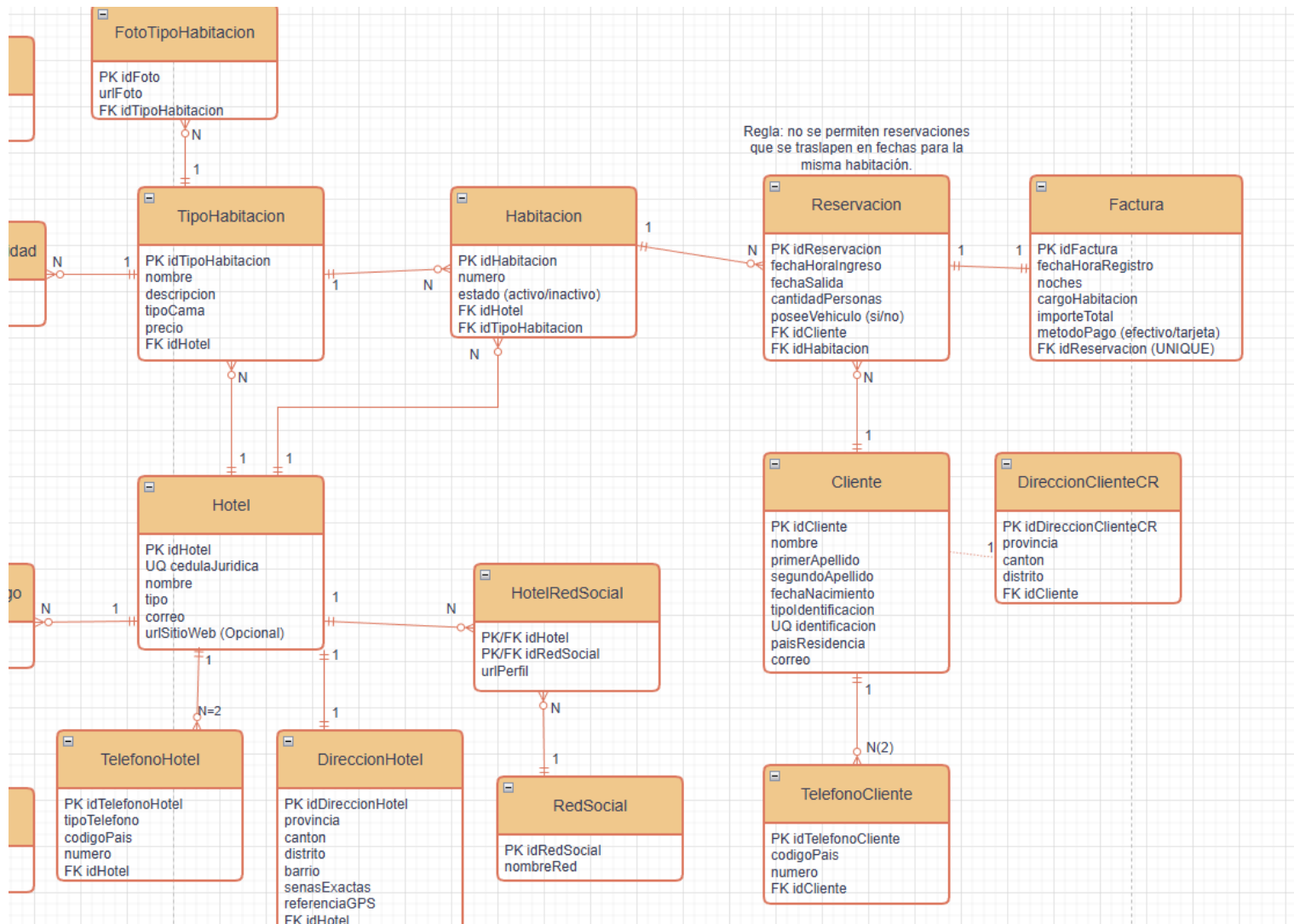
Campo	Tipo de dato	PK	FK	Nulo	Descripción
idTipoActividad	INT	Sí	No	No	Identificador del tipo
nombreTipoActividad	NVARCHAR(60)	No	No	No	Nombre del tipo de actividad

21. Tabla: ActividadTipoActividad

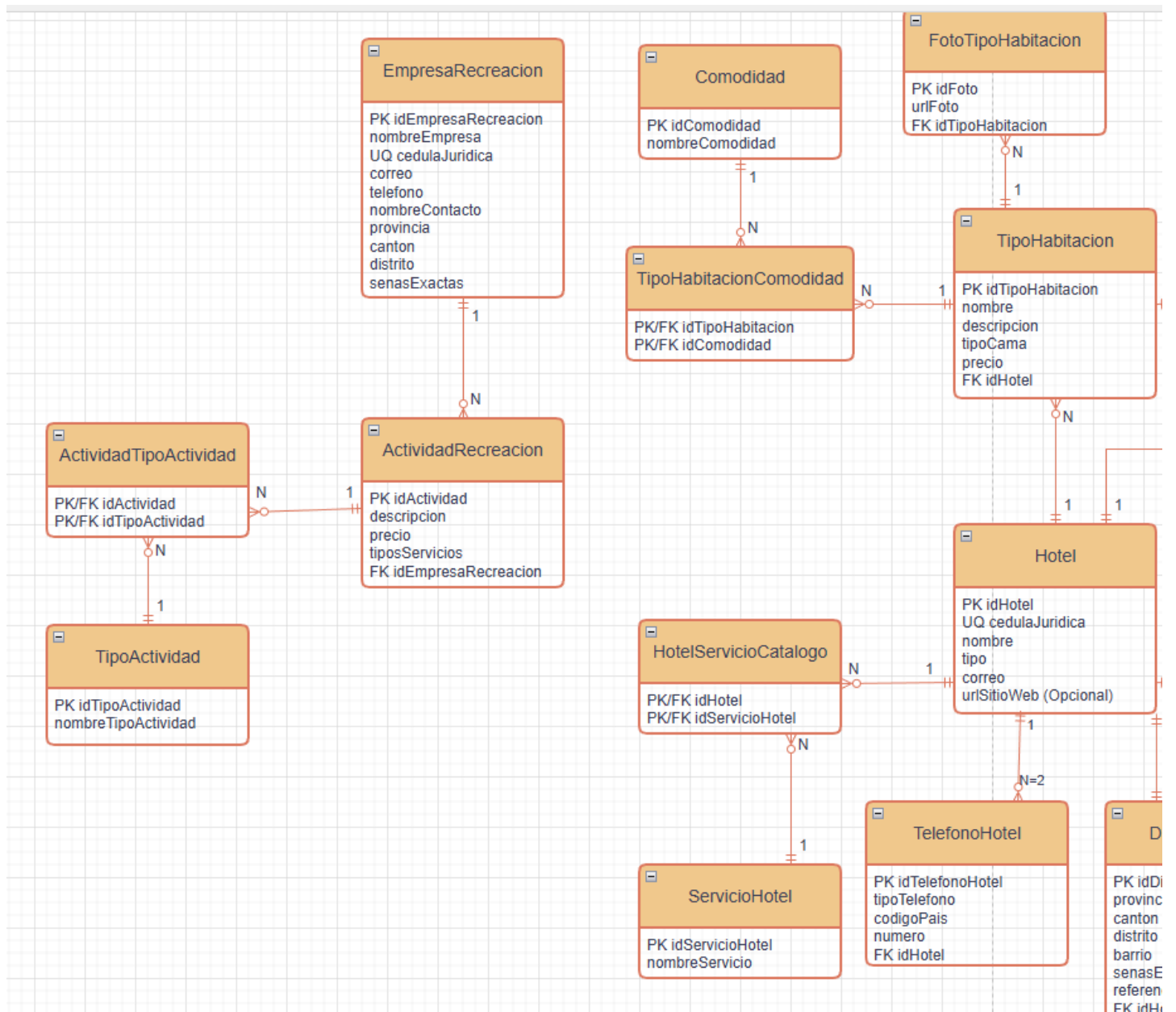
Campo	Tipo de dato	PK	FK	Nulo	Descripción
idActividad	INT	Sí	Sí	No	Actividad asociada
idTipoActividad	INT	Sí	Sí	No	Tipo de actividad asociado

# Diagrama relacional de la base de datos

## Lado Derecho con Hotel, cliente y reservación



## Lado izquierdo con hotel - Recreacion - habitación



# Prototipos de la aplicación con enlace público

## Inicio de sesión



A Web Page  
← → ↻ <https://AlimentosLimon.com>

### Iniciar Sesión

¡Te damos la bienvenida!

Usuario:

Contraseña:







[Crear Usuario/Recuperar Acceso](#)

## Apartado de Crear usuario



### Crear usuario

¡Te damos la bienvenida!

Datos del Usuario

Fecha de nacimiento:

[Sign in](#)

## Apartado para registrarse como anfitrión

### Agregando Anfitrión

¡Te damos la bienvenida!

Comenta como encontrarte

Provincia

Cantón

Distrito

Comenta tu dirección exacta o reseñas



Continuar

[Sign in](#)

## Apartado para indicar el tipo de lugar

### Agregando Anfitrión

¿Cuál de estas opciones describe mejor tu espacio?

Selección múltiple

 Casa	 Cabaña	 Casa rodante	 bed & breakfast
 Hotel	 Hostel	 Barco	 Camping
 Domo	 Casa compartida	 Parque nacional	 Departamentos

Continuar

[Sign in](#)

## Apartado para indicar formas de contactar

### Agregando Anfitrión

¡Ya estas a un paso de terminar!

Comparte tus ultimos datos para administrar tu negocio

Nombre del lugar

Cédula Jurídica

URL del sitio web

Correo electrónico

Numero telefónico

Contraseña

Confirmar contraseña

Continuar

[Sign in](#)

## Forma de agregar habitaciones

### Agregando Habitación

Informacion de la habitacion

Número de personas:

Niños

Adultos

Animales

Vehículos

10

10

10

10

Tipo de Habitación

Tipo de Cama

Comenta sobre la habitación y la lista de servicios que ofrece

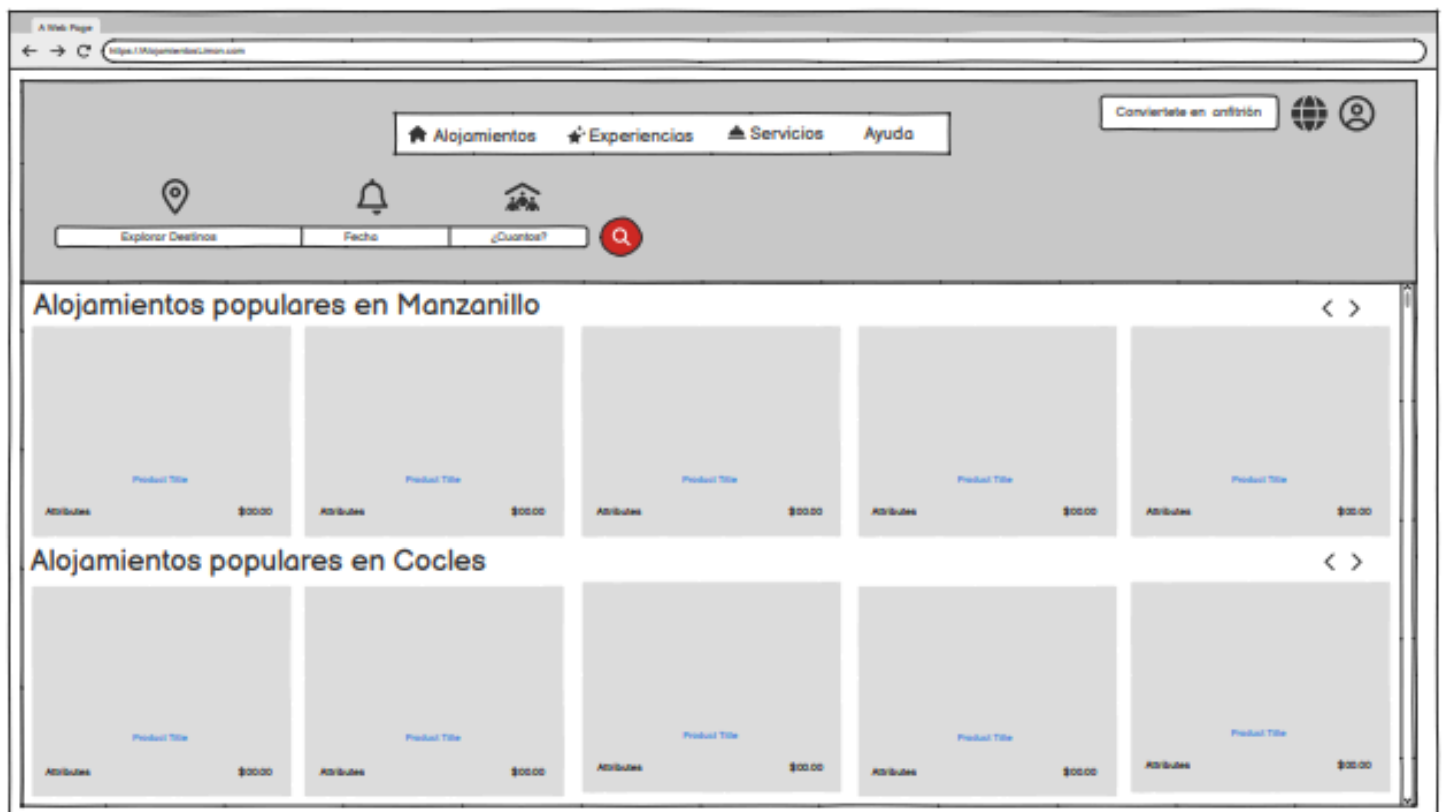
\$ Monto de la habitación

📎

Elija uno o varios archivos.png para cargar

Continuar

## Pagina inicial - vista usuario



## Apartado de habitaciones o lugares

### Habitación

Información de la habitación



 ¿Cómo Llegar?

Texto de reseñas para llegar a la ubicación brindados por el usuario anfitrión



Información del Anfitrión

Nombre del Anfitrión  Número telefónico

Número de personas:

Niños	Adultos	Animales	Vehículos
<input type="text" value="10"/>	<input type="text" value="10"/>	<input type="text" value="10"/>	<input type="text" value="10"/>

Tipo de Habitación  Tipo de Cama

Cantidad de días:

\$ Monto de cancelación

Reservar

Facturar



## Reservaciones

### Haciendo reservación

Estas a un paso de reservar

Indica cuando quieres reservar

Día	Mes	Año
10	Enero	2026

Nombre de la habitación  
Nombre de la persona que reserva  
Niños: 1 Adultos: 2 Animales: 0 Vehículos: 1  
Cantidad de días: 10  
Contacto del anfitrión: +506 24940184  
Descripción completa de la habitación  
Fecha límite de cancelación

Cantidad de días: 10

\$ Monto de cancelación



Formas de pago

Confirmar reservación

Facturar

## Facturas

### Haciendo la factura

Estas a un paso de facturar

ID de factura  
Numero de Habitación  
Nombre de la habitación  
Nombre de la persona que reserva  
Cedula de la persona  
Niños: 1 Adultos: 2 Animales: 0 Vehículos: 1  
Cantidad de días: 10  
Contacto del anfitrión: +506 24940184  
Descripción completa de la habitación  
Empresa o anfitrión asociado  
Cantidad de días: 10  
Lista de servicios ofrecidos por el anfitrión  
Dia de entrada: 19/08/2026  
Hora de entrada: 13:00  
Dia de salida: 29/08/2026  
Hora de salida: 13:10  
Monto a cancelar: \$350,000.



Descargar factura

Continuar

## Vista de Anfitrión con búsqueda por filtros

A Web Page

← → ↻ https://AlajamientosUman.com

Galería

Sobre nosotros

Mis servicios

Ayuda

Agregar Habitación

Habitaciones disponibles

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Habitaciones ocupadas

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Product Title

Attributes

\$00.00

Tipo de Habitación

Niños

10

Adultos

10

Animales

10

Precio

ID o Nombre de la habitación

Nombre o cédula jurídica

## Filtro

Tipo de Habitación

Niños

10

Adultos

10

Animales

10

Precio

ID o Nombre de la habitación

Nombre o cédula jurídica

## Análisis de resultados

El desarrollo del Sistema de Gestión Hotelera facilitó el logro exitoso de las metas propuestas en el proyecto, alcanzando una solución completa que incluye no solo la gestión de los establecimientos hoteleros sino también los servicios adicionales de recreación.

Primero que todo se logró modelar correctamente la información necesaria para reconocer las entidades principales del sistema, tales como hoteles, habitaciones, clientes, reservaciones, facturación y empresas de recreación. El cual se muestra mediante un Diagrama Entidad-Relación, evitando redundancia de datos y asegurando la integridad referencial entre todas las entidades.

El diseño de la base de datos en SQL Server permitió materializar el modelo conceptual propuesto, incorporando claves primarias, claves foráneas, restricciones de unicidad y validaciones básicas que garantizan la consistencia de la información almacenada. Gracias a esta estructura, el sistema puede manejar de forma adecuada escenarios reales como múltiples teléfonos por hotel o cliente, distintos tipos de habitaciones por establecimiento y la generación de una única factura por cada reservación.

Asimismo, el diccionario de datos elaborado facilitó la documentación detallada de cada tabla y atributo, lo cual contribuye a una mejor comprensión del sistema y permite que futuras modificaciones o ampliaciones puedan realizarse de manera ordenada. Esta documentación resulta fundamental para el mantenimiento y escalabilidad del sistema.

Por otro lado, los prototipos de interfaz diseñados permitieron visualizar el funcionamiento del sistema desde la perspectiva del usuario final. A través de pantallas de registro, consulta y gestión, se evidenció que la solución propuesta es intuitiva y responde a las necesidades planteadas en el contexto del proyecto, incluyendo la búsqueda de servicios de recreación mediante filtros, similar a plataformas comerciales como Airbnb.

Finalmente, el desarrollo del proyecto permitió reforzar conocimientos clave en el diseño de bases de datos relacionales, normalización, modelado de relaciones uno a muchos y muchos a muchos, así como la importancia de

una correcta documentación técnica. En conclusión, los resultados obtenidos demuestran que la solución propuesta es funcional, coherente y alineada con los requerimientos establecidos, cumpliendo de manera satisfactoria con los objetivos del proyecto.

## Bibliografía

1. Date, C. J. (2003). \*An Introduction to Database Systems\*. Addison-Wesley.
2. Coronel, C., Morris, S. (2015). \*Database Systems: Design, Implementation, & Management\*. Cengage Learning.
3. ISO/IEC 11179 – Metadata registries.-
4. Oracle Documentation – Data Dictionary Concepts.