



Identificación de Usuarios y Roles

(Reserva-Hotel)

En este proyecto tenemos **3 tipos de usuarios**:



1. Cliente

Descripción:

Usuario externo que entra al sistema para reservar habitaciones en el hotel.

✅ Acciones (permisos):

- Registrarse / iniciar sesión.
- Ver listado de habitaciones disponibles.
- Realizar reservas seleccionando fechas.
- Ver su historial de reservas.
- Cancelar o modificar una reserva (si la fecha de entrada aún no llega).
- Recibir correos de confirmación.



Campos en BDD (usuarios):

Campo	Tipo	Nota
id_usuario	UUID (PK)	Clave primaria
nombre_completo	VARCHAR(100)	Obligatorio
email	VARCHAR(100)	Único, validado, requiere verificación
password_hash	TEXT	Hash bcrypt
telefono	VARCHAR(20)	Opcional
documento_identidad	VARCHAR(20)	Opcional (para identificación del cliente)
fecha_nacimiento	DATE	Opcional
rol	ENUM ('cliente')	Define el rol
estado	ENUM ('activo', 'inactivo')	Para desactivar cuentas sin eliminarlas
created_at	TIMESTAMP	Fecha de creación
updated_at	TIMESTAMP	Fecha de última actualización



2. Recepcionista

Descripción:

Usuario interno que gestiona reservas y habitaciones en nombre del hotel.

✅ Acciones (permisos):

- Iniciar sesión (cuenta creada por el admin).
- Ver todas las reservas activas, pasadas y futuras.
- Crear reservas a nombre de un cliente (walk-in o llamada telefónica).
- Editar o cancelar reservas.
- Gestionar check-in y check-out de clientes.
- Recibir notificaciones de nuevas reservas o check-outs pendientes.



Campos en BDD (usuarios):

Campo	Tipo	Nota
id_usuario	UUID (PK)	Clave primaria
nombre_completo	VARCHAR(100)	Obligatorio
email	VARCHAR(100)	Único, para login
password_hash	TEXT	Hash bcrypt
rol	ENUM ('recepcionista')	Define el rol
estado	ENUM ('activo', 'inactivo')	Para desactivar cuentas sin eliminarlas
created_at	TIMESTAMP	Fecha de creación
updated_at	TIMESTAMP	Fecha de última actualización



3. Administrador

Descripción:

Usuario interno con control total del sistema.

✅ Acciones (permisos):

- Iniciar sesión.
- Crear, editar y eliminar habitaciones.
- Ver todas las reservas y su estado.

- Crear cuentas de recepcionistas y desactivarlas si es necesario.
- Generar reportes básicos (ocupación, reservas activas).
- Configurar políticas de seguridad (ej. contraseñas, autenticación en 2 pasos).



Campos en BDD (usuarios):

Campo	Tipo	Nota
id_usuario	UUID (PK)	Clave primaria
nombre_completo	VARCHAR(100)	Obligatorio
email	VARCHAR(100)	Único, para login
password_hash	TEXT	Hash bcrypt
rol	ENUM ('admin')	Define el rol
estado	ENUM ('activo', 'inactivo')	Para desactivar cuentas sin eliminarlas
created_at	TIMESTAMP	Fecha de creación
updated_at	TIMESTAMP	Fecha de última actualización



Estructura de Roles

Usaremos un campo rol en la tabla usuarios (más simple para MVP).

Los **middlewares del backend** harán:

- ✓ Validación JWT.
- ✓ Revisión del rol para restringir el acceso a rutas.

Rol	Nivel de acceso
admin	Total (CRUD + gestión de usuarios/roles).
recepcionista	Gestión de reservas y check-in/out.
cliente	Solo gestión de sus propias reservas.



Diagrama de Relación: Usuarios y Roles

(Simplificado para el Sprint 0 MVP)

```

+-----+ +-----+ | usuarios | | habitaciones | +-----+
+-----+ | id_usuario (PK) | | id_habitacion (PK) | | nombre_completo | |
numero | | email | | descripcion | | password_hash | | estado | | rol | | precio | | estado | |
created_at | | created_at | | updated_at | | updated_at | +-----+ +-----+
-----+ ^ | | | +-----+ reservas +-----+ |
id_reserva (PK) | | id_usuario (FK) | | id_habitacion (FK) | | fecha_inicio | | fecha_fin | |
estado | | metodo_pago | | estado_pago | | comentarios | | created_at | | updated_at | +---
-----+

```