

# HOTELES DECAMERON DE COLOMBIA

## Dirección de Desarrollo

Hola ingenier@, que bueno tenerlo en la empresa, le comento que el gerente de operaciones hoteleras requiere un sistema que permita ingresar los hoteles con los que cuenta la compañía, además de los nombres básicos del hotel, se deben ingresar los datos tributarios básicos.

Adicional a eso el gerente hotelero requiere que a los hoteles con los que cuenta la compañía, se les pueda asignar los tipos de habitación (Estándar, Junior y Suite). Este sistema debe validar que únicamente se pueda asignar las acomodaciones según el tipo:

- Si el tipo de habitación es Estándar: la acomodación debe ser Sencilla o Doble.
- Si el tipo de habitación es Junior: la acomodación debe ser Triple o Cuádruple
- Si el tipo de habitación es Suite: la acomodación debe ser Sencilla, Doble o Triple

Ejemplo:

Nombre:	<b>DECAMERON CARTAGENA</b>	DIRECCIÓN:	<b>CALLE 23 58-25</b>
Ciudad:	<b>CARTAGENA</b>	NIT:	<b>12345678-9</b>
Numero de Hab:	<b>42</b>		

CANTIDAD	TIPO HABITACIÓN	ACOMODACIÓN
25	ESTANDAR	SENCILLA
12	JUNIOR	TRIPLE
5	ESTANDAR	DOBLE

### Criterios de aceptación:

- La cantidad de habitaciones configuradas, no deben superar el máximo por hotel
- No deben existir hoteles repetidos
- No debe existir tipos de habitaciones y acomodaciones repetidas para el mismo hotel
- No se requieren administradores para datos catálogos, como ciudades, tipos de habitación o acomodación.
- Los gerentes usan portátiles de 15 y algunos casos de 13 pulgadas.

### Criterios técnicos de aceptación:

- La aplicación debe ser totalmente RESTFULL.
- El back y el front deben estar desacoplados
- El back debe ser en PHP
- Entregar la documentación del proyecto que considere importante (por ejemplo: diagramas UML).
- La base de datos debe ser postgresql
- La aplicación debe desplegarse en la nube (puede utilizar la nube que desee). Se debe enviar el link para acceder a la aplicación.
- Se debe enviar el código realizado en un archivo comprimido o compartirlo en un repositorio GIT público para poder descargarlo.
- La aplicación se usará en navegadores Firefox y Chrome

- Considere buenas prácticas como: patrones de diseño, principios SOLID, código documentado, otros).

Favor no olvidar:

- Enviar dump de la BD, lista para instalar.
- Se debe enviar el paso a paso para ejecutar la aplicación, como si su abuelita quisiera realizar el despliegue 😊.

## **Documentación del Proyecto: Sistema de Gestión Hotelera**

### **1. Diseño del Sistema**

#### **1.1 Modelo de Datos (Base de Datos en PostgreSQL)**

La base de datos de este sistema está diseñada para almacenar información relacionada con los hoteles, las habitaciones, los tipos de habitación y las acomodaciones. La estructura de las tablas se ha optimizado para reflejar la relación entre estos elementos, asegurando integridad y eficiencia en las consultas.

##### **1.1.1 Tablas**

###### **1. Hoteles**

- **Descripción:** Almacena los datos de los hoteles.
- **Campos:**
  - id (int): Identificador único del hotel.
  - nombre (varchar): Nombre del hotel.
  - direccion (varchar): Dirección completa del hotel.
  - ciudad (varchar): Ciudad en la que se ubica el hotel.
  - nit (varchar): Número tributario único para el hotel.
  - num\_habitaciones (int): Número máximo de habitaciones que el hotel puede tener.
- **Restricciones:**
  - nit debe ser único.
  - id es la clave primaria.

###### **2. Tipos de Habitación**

- **Descripción:** Almacena los tipos de habitaciones disponibles.
- **Campos:**
  - id (int): Identificador único del tipo de habitación.
  - nombre (varchar): Nombre del tipo de habitación (Ej. Estándar, Junior, Suite).
- **Restricciones:**
  - id es la clave primaria.

##### **Tipos de habitación precargados:**

- Estándar
- Junior
- Suite

### 3. Acomodaciones

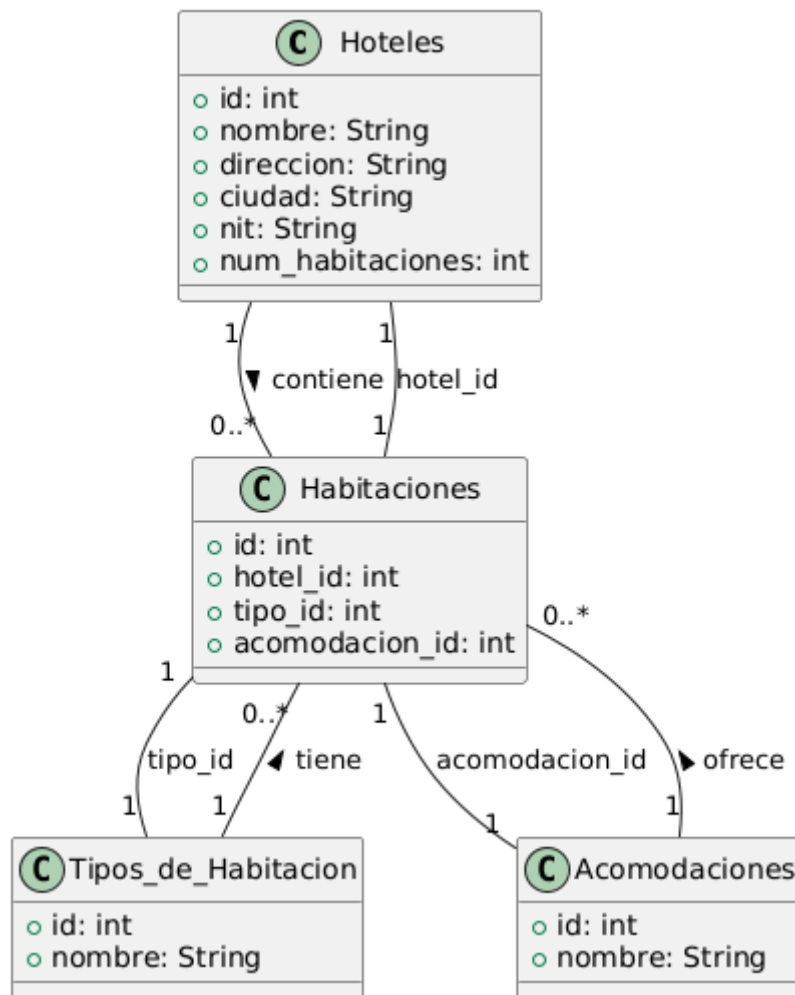
- **Descripción:** Almacena las diferentes acomodaciones disponibles.
- **Campos:**
  - id (int): Identificador único de la acomodación.
  - nombre (varchar): Nombre de la acomodación (Ej. Sencilla, Doble, Triple, Cuádruple).
- **Restricciones:**
  - id es la clave primaria.

#### Acomodaciones precargadas:

- Sencilla
- Doble
- Triple
- Cuádruple

### 4. Habitaciones

- **Descripción:** Almacena información sobre las habitaciones en cada hotel.
- **Campos:**
  - id (int): Identificador único de la habitación.
  - hotel\_id (int): Relación con la tabla hoteles, indicando a qué hotel pertenece la habitación.
  - tipo\_id (int): Relación con la tabla tipos\_habitacion, indicando el tipo de habitación.
  - acomodacion\_id (int): Relación con la tabla acomodaciones, indicando la acomodación de la habitación.
  - numero (int): Número de habitaciones de ese tipo en el hotel.
- **Restricciones:**
  - id es la clave primaria.
  - hotel\_id es una clave foránea que hace referencia a la tabla hoteles.
  - tipo\_id es una clave foránea que hace referencia a la tabla tipos\_habitacion.
  - acomodacion\_id es una clave foránea que hace referencia a la tabla acomodaciones.



## 1.2 Relaciones entre las Tablas

- **Relación entre Hoteles y Habitaciones:**
  - Cada habitación está asociada a un solo hotel, pero un hotel puede tener muchas habitaciones.
- **Relación entre Habitaciones y Tipos de Habitación:**
  - Cada habitación está asociada a un tipo de habitación específico, como Estándar, Junior, o Suite.
- **Relación entre Habitaciones y Acomodaciones:**
  - Cada habitación tiene una acomodación asociada, que puede ser Sencilla, Doble, Triple, o Cuádruple.

## 1.3 Restricciones de Integridad

- **No permitir tipos de habitación y acomodaciones duplicadas para el mismo hotel:**
  - Para garantizar que un hotel no tenga múltiples registros con el mismo tipo de habitación y acomodación, se ha aplicado una restricción de unicidad en las combinaciones de hotel\_id, tipo\_id y acomodacion\_id en la tabla habitaciones.

## 2. Lógica de Negocio

### 2.1 Validación de la Adición de Habitaciones

- **Verificación de Habitaciones Duplicadas:** No se permiten múltiples entradas para el mismo tipo de habitación y acomodación en un hotel. Esto asegura que el sistema no registre más de una habitación con las mismas características en el mismo hotel.
- **Validación de Capacidad del Hotel:** El sistema verifica que el número total de habitaciones no supere el límite definido por num\_habitaciones del hotel. Si el total de habitaciones excede este valor, no se permite agregar más habitaciones.

### 2.2 Cálculo de la Suma de Habitaciones

- Al agregar una nueva habitación, el sistema calcula la suma de las habitaciones existentes en el hotel y valida que el nuevo número de habitaciones no supere el total permitido.

### 2.3 Relación de Tipos de Habitaciones con Acomodaciones

- Se aplica una validación para garantizar que una acomodación sea válida solo para ciertos tipos de habitaciones. Por ejemplo, la acomodación "Sencilla" solo es válida para tipos de habitación "Estándar" o "Suite".

## 3. API RESTful

La API RESTful implementa los siguientes endpoints principales:

1. **POST /api/hoteles:** Crear un nuevo hotel.
2. **GET /api/hoteles:** Obtener la lista de todos los hoteles.
3. **POST /api/habitaciones:** Crear una nueva habitación en un hotel (requiere hotel\_id, tipo\_id, acomodacion\_id, numero).
4. **GET /api/habitaciones/{hotelId}:** Obtener las habitaciones asociadas a un hotel.

## 4. Manual de Usuario

El manual de usuario debe detallar cómo los administradores o gerentes pueden interactuar con el sistema para crear hoteles, asignar habitaciones y realizar la gestión de tipos y acomodaciones.

## 5. Recomendaciones de Implementación

- **Back-End:** Laravel para manejar las peticiones HTTP, validaciones y la lógica de negocio.
- **Front-End:** React para la interfaz de usuario, con Axios para consumir los endpoints de la API.
- **Base de Datos:** PostgreSQL para almacenar la información de hoteles, habitaciones, tipos y acomodaciones

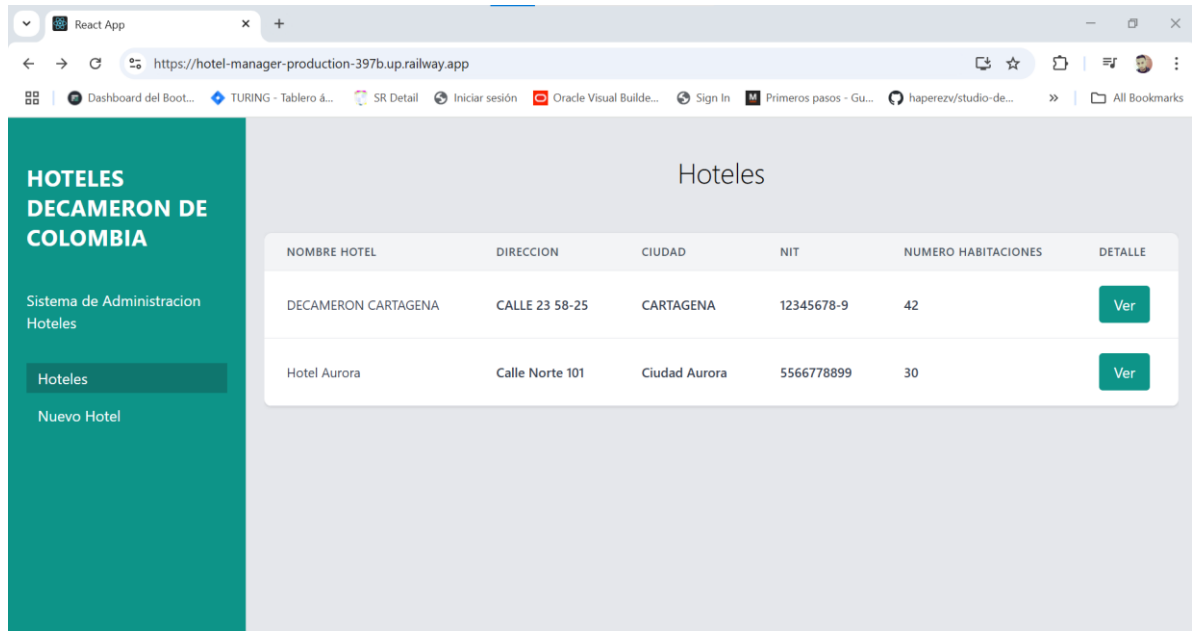
Ejecutar los 2 links para que la aplicación pueda subir:

1. Backend: <https://backend-laravel-production-664f.up.railway.app/api/hoteles>

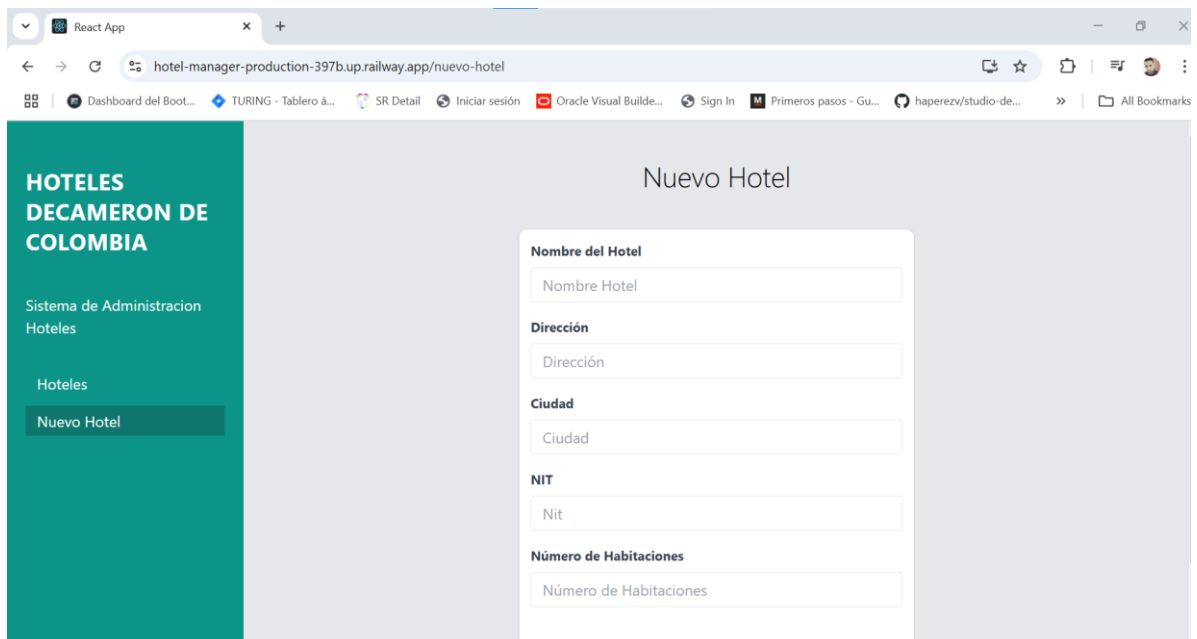
2. Frontend: <https://hotel-manager-production-397b.up.railway.app/>

## Manual de usuario

- Para ingresar a la aplicación lo puede realizar accediendo al siguiente link: <https://hotel-manager-production-397b.up.railway.app/>



- Para crear un nuevo hotel acceda a crear Nuevo Hotel



- En la opción Ver podrá Agregar la habitación, tipo y acomodación.

React App

hotel-manager-production-397b.up.railway.app/hotel/1

NIT: 12345678-9

Número de Habitaciones: 42

### Agregar Nueva Habitación

**Tipo de Habitación**

Seleccione un tipo

**Acomodación**

Seleccione una acomodación

**Número de Habitaciones**

Cantidad de habitaciones

**Guardar Habitación**

#### Asignación de Habitaciones

Cantidad	Tipo de Habitación	Acomodación
25	Estándar	Sencilla
12	Junior	Triple
5	Estándar	Doble

- Para acceder al api, se puede acceder desde esta url: <https://backend-laravel-production-664f.up.railway.app/api/hoteles>

backend-laravel-production-66

<https://backend-laravel-production-664f.up.railway.app/api/hoteles>

Dashboard del Boot... TURING - Tablero á... SR Detail Iniciar sesión Oracle Visual Bui... Sign In Primeros pas...

Pretty-print ☒

```
[
  {
    "id": 1,
    "nombre": "DECAMERON CARTAGENA",
    "direccion": "CALLE 23 58-25",
    "ciudad": "CARTAGENA",
    "nit": "12345678-9",
    "num_habitaciones": 42,
    "created_at": "2024-11-28T21:39:52.000000Z",
    "updated_at": "2024-11-28T21:39:52.000000Z"
  },
  {
    "id": 2,
    "nombre": "Hotel Aurora",
    "direccion": "Calle Norte 101",
    "ciudad": "Ciudad Aurora",
    "nit": "5566778899",
    "num_habitaciones": 30,
    "created_at": "2024-11-28T22:50:34.000000Z",
    "updated_at": "2024-11-28T22:50:34.000000Z"
  }
]
```