

# Sistema de Gestión Hotelero - Documentación por Módulos - Reservas

---

**Programa:**

Tecnólogo en Análisis y Desarrollo de Software  
2977341

**Sede:**

SENA Regional Boyacá  
Centro de Desarrollo Agropecuario y Agroindustrial

**Evidencia:**

GA8-220501096-AA1-EV02

**Realizado por:**

María Jimena Sánchez Pinilla

**Instructor:**

Jorge Eliecer Niño Ochoa

**Fecha:**

15 de junio de 2025

# Documentación por Módulos

## 1. Módulo: Registrar una Reserva

**Nombre del Archivo:** BookingModel.js, BookingController.js, BookingRoutes.js, BookingsAdmin.jsx

**Descripción:** Permite registrar una nueva reserva.

**Datos de entrada:**

- "id\_user": "número",
- "check\_in": "fecha",
- "check\_out": "fecha",
- "room\_number": "texto",
- "total": "número"

**Proceso:**

- El usuario presiona el botón “Crear reserva”, el modal con el formulario se abre, ingresa los datos y presiona “Guarda”.
- El frontend envía los datos al backend. El backend valida que se hayan ingresado todos los datos.
- Si no se ingresaron todos los datos responde un mensaje de error.
- Si se ingresaron todos los datos, verifica que el room\_number exista.
- Si no existe el room\_number responde un mensaje de error.
- Si existe, procede a generar un código único para identificar la reserva.
- Después, a partir del room\_number, se obtiene el id\_room con el objetivo de crear la entidad en booking\_room.
- El backend envía los datos de la reserva a la base de datos.
- Después, obtiene el id\_booking de la nueva reserva para crear la entidad en booking\_room.
- Luego, se envían los datos a la base de datos para crear el booking\_room.
- La respuesta de este proceso se envía al frontend.

**Datos de salida:**

Si no se ingresaron todos los datos del formulario: mensaje de error.

Si no existe el room\_number: mensaje de error.

Si se ingresaron todos los datos del formulario, el room\_number existe y el proceso para crear la reserva y el booking\_room es exitoso: mensaje de éxito.

## 2. Módulo: Consultar Reservas

**Nombre del Archivo:** BookingModel.js, BookingController.js, BookingRoutes.js, BookingsAdmin.jsx

**Descripción:** Permite consultar los datos de una reserva existente.

**Datos de entrada:** code

**Proceso:**

- El usuario presiona el botón “Consultar”, el frontend realiza una consulta al backend y envía el code de la reserva que se eligió.
- El backend recibe el código, se comunica con la base de datos, se realiza la consulta y genera una respuesta.
- Si no encuentra un booking a partir de dicho código envía un mensaje de error al frontend.
- Si encuentra el booking envía dichos datos al frontend.
- El modal se abre y muestra la información al usuario.

**Datos de salida:**

- Si el código no existe: mensaje de error.
- Si el código existe: mensaje de éxito.

## 3. Módulo: Modificar Reservas

**Nombre del Archivo:** BookingModel.js, BookingController.js, BookingRoutes.js, BookingsAdmin.jsx

**Descripción:** Permite modificar los datos de una reserva existente.

**Datos de entrada:** code

**Proceso:**

- El usuario administrativo presiona el botón modificar, se abre un modal que muestra los datos actuales y un formulario para realizar cambios en los datos.
- Cuando el usuario presiona el botón “Guardar”, el frontend realiza una consulta al backend y envía el code de la reserva que fue seleccionada.
- El backend recibe los datos. Primero consulta a la base de datos la información existente en la reserva.
- Después, almacena los datos enviados desde el frontend y verifica que se hayan ingresado todos los datos obligatorios.
- A partir del user\_id\_number se obtiene el id\_user que se almacena en la tabla booking.
- Se verifica que el nuevo room\_number exista.
- Si no existe se envía un mensaje al frontend.

- Si existe se obtiene el id\_room.
- Teniendo en cuenta que el id\_booking no se puede modificar, se obtiene este dato de la información existente.
- Se verifica si hubo cambios entre el id\_room original y el nuevo id\_room.
- Si no se realizaron cambios no se actualiza la tabla booking\_room.
- Si se realizaron cambios se actualiza la tabla booking\_room.
- Despues, se comparan los valores originales de la reserva con los nuevos valores de la reserva para verificar si hubo cambios.
- Si no se realizaron cambios envía un mensaje al frontend.
- Si se realizaron cambios se eliminan aquellos valores que no hacen parte de la tabla de booking y los restantes se envían a la base de datos para ejecutar la consulta.
- Se recibe una respuesta y se envía al frontend.

**Datos de salida:**

- Si no se ingresan todos los campos obligatorios del formulario: mensaje de error.
- Si el room\_number no existe: mensaje de error.
- Si no se realizaron cambios en los datos: mensaje de error.
- Si se ingresaron todos los datos del formulario, se realizó al menos un cambio y el room\_number existe: mensaje de éxito.