

Sistema de Gestión Hotelero - Documentación por Módulos - Habitaciones

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 Habitación

- **Nombre del Archivo:** RoomModel.js, RoomController.js, RoomRoutes.js, RoomAdmin.jsx
- **Descripción:** Permite registrar una nueva habitación en el sistema.
- **Datos de entrada:**
 - "id_room": "número",
 - "room_number": "texto",
 - "rate": "número",
 - "id_room_type": " número ",
 - "id_room_status": " número ",
 - "id_floor": " número ",
 - "photo_path": "texto"
- **Proceso:**
 - El usuario administrativo presiona el botón “Crear habitación”, se abre el modal, diligencia los datos del formulario y presiona el botón “Crear”.
 - El frontend envía los datos registrados al backend y espera una respuesta.
 - El backend recibe los datos y valida que se hayan ingresado todos.
 - Si no se ingresaron todos envía un mensaje al frontend.
 - Si se ingresaron todos los datos verifica, a través del room_number, que la habitación no exista.
 - Si la habitación ya existe envía un mensaje al frontend.
 - Si la habitación no existe se comunica con la base de datos.
 - La base de datos recibe los campos, ejecuta la solicitud SQL y envía una respuesta al backend.
 - El backend envía una respuesta al frontend, el frontend muestra un mensaje al usuario y actualiza la tabla de habitaciones.
- **Datos de salida:**
 - Si no se ingresaron todos los datos del formulario: mensaje de error.
 - Si la habitación ya existe: mensaje de error.

- Si se ingresaron todos los datos y la habitación no existe: mensaje exitoso.

2. Módulo: Consultar Habitaciones

- **Nombre del Archivo:** RoomModel.js, RoomController.js, RoomRoutes.js, RoomAdmin.jsx
- **Descripción:** Permite consultar todos los datos asociados a una habitación.
- **Datos de entrada:** id_room
- **Proceso:**
 - El usuario administrativo presiona el botón “Consultar”, el frontend envía una solicitud al backend con el id_room de la habitación seleccionada.
 - El backend recibe el valor y verifica si la habitación existe.
 - Si no existe envía una respuesta al frontend con un mensaje de error.
 - Si la habitación existe se comunica con la base de datos para realizar la consulta SQL.
 - La base de datos envía al backend el resultado de la consulta. El backend envía la respuesta al frontend.
 - El frontend muestra los datos obtenidos del backend en el modal.
- **Datos de salida:**
 - Si el id_room no existe: mensaje de error.
 - Si el id_room existe: se muestran los datos de la habitación.

3. Módulo: Modificar Habitaciones

- **Nombre del Archivo:** RoomModel.js, RoomController.js, RoomRoutes.js, RoomAdmin.jsx
- **Descripción:** Permite modificar o actualizar los datos de una habitación.
- **Datos de entrada:** id_room
- **Proceso:**
 - El usuario presiona el botón “Modificar”, el Modal se abre y el usuario cambia los datos en el formulario.

- Cuando el usuario presiona el botón “Guardar”, el frontend realiza una solicitud al backend y envía los datos de la habitación actual y aquellos diligenciados por el usuario.
 - El backend recibe los datos y verifica que se hayan ingresado todos los campos necesarios.
 - Si no se ingresaron todos los campos envía al frontend un mensaje de error.
 - Si se proporcionaron todos los campos necesarios, los recorre uno por uno para verificar en cuáles hubo cambios y elimina aquellos que no hacen parte de la tabla de habitaciones.
 - El backend verifica si hay al menos un valor nuevo para actualizar la base de datos, si no lo encuentra envía al frontend un mensaje de error.
 - Si hay valores válidos para actualizar se comunica con la base de datos.
 - La base de datos recorre cada uno de los campos y valores proporcionados, y los modifica en la habitación que cumple con el id_room. Después, envía un resultado al backend.
 - El backend envía una respuesta al frontend, el frontend muestra un mensaje al usuario y actualiza la tabla de habitaciones.
- **Datos de salida:**
 - Si no se ingresaron todos los campos necesarios: mensaje de error.
 - Si se enviaron todos los campos, pero no se realizaron cambios en ningún valor: mensaje de error.
 - Si no existe el id_room: mensaje de error.
 - Si existe el id_room, se proporcionaron todos los campos obligatorios y se realizó cambios en al menos un valor: mensaje de éxito.

4. Módulo: Eliminar Habitaciones

- **Nombre del Archivo:** RoomModel.js, RoomController.js, RoomRoutes.js, RoomAdmin.jsx
- **Descripción:** Permite eliminar una habitación del sistema.
- **Datos de entrada:** room_number
- **Proceso:**
 - El usuario administrativo selecciona el botón “Eliminar” de una habitación existente. Se abre el modal de verificación.

- Si el usuario presiona “Eliminar” el frontend realiza una consulta al backend y envía el room_number de dicha habitación.
 - El backend recibe el room_number, se comunica con la base de datos y envía el dato recibido.
 - La base de datos ejecuta la petición SQL y envía una respuesta al backend.
 - El backend verifica si alguna de las filas de la base de datos fue afectada, si ninguna fue afectada envía al frontend un mensaje de error.
 - Si alguna de las filas fue afectada envía un mensaje de éxito al frontend.
 - El frontend recibe la respuesta, muestra un mensaje en la pantalla del usuario y actualiza la tabla de habitaciones.
- **Datos de salida:**
 - Si existe el room_number y no tiene reservas en curso la habitación: mensaje de éxito.
 - Si la habitación tiene reservas en curso: mensaje de error.
 - Si no existe el room_number: mensaje de error.