

## PRUEBA BACKEND

Utilizando Django REST Framework, desarrollé los endpoints para el sistema de reservas de habitación de un hotel.  
CONDICIONES:

- Las reservas pueden tener 3 estados: Pendiente, Pagado y Eliminado.
- Los datos a almacenar para la reserva son: los detalles del cuarto reservado, los días de estadía, los datos de facturación e identificación del cliente, el monto pagado y el método de pago.
- Proponé los endpoints a crearse para tratar de cubrir el flujo normal de operación de reserva y explicar por qué.

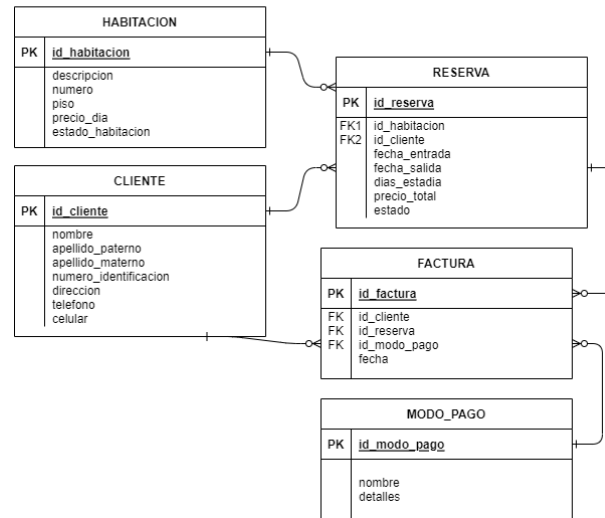
Luego que tengas ya todo el código

- Crear un repositorio para la entrega del código y en un readme del repositorio la justificación de los endpoints creados.

## ANALISIS

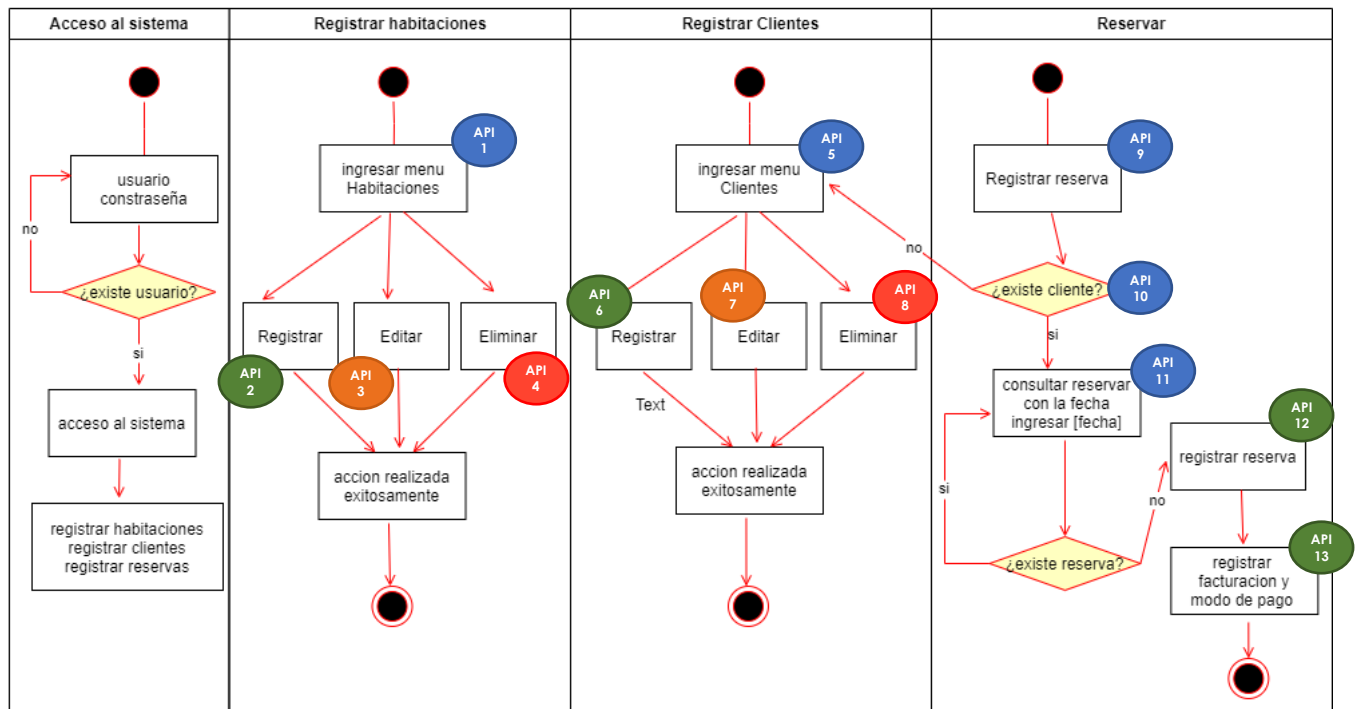
### BASE DE DATOS

#### Reservas habitación



### FLUJO Reservas

#### Rol Administrador



## EXPLICACION ENDPOINTS

### API – 1 Lista Habitaciones

Identificador:	API - 1	Petición: GET
<b>Proceso:</b> Habitaciones <b>Acción:</b> Listar habitaciones	<b>Descripción API:</b> Devuelve todas las habitaciones registradas en el hotel del modelo [Habitación]	
	GET	http://127.0.0.1:8000/reservar/v1/listaHabitacion/
<pre>[   {     "id_habitacion": 1,     "descripcion": "Habitacion Nro 1",     "numero": 1,     "piso": 1,     "precio_dia": "50.000",     "estado_habitacion": "Bueno"   },   {     "id_habitacion": 4,     "descripcion": "Habitacion Nro 4",     "numero": 4,     "piso": 2,     "precio_dia": "70.000",     "estado_habitacion": "Bueno"   } ]</pre>		<b>Parámetros:</b> (sin parámetros)

### API – 2 Registra Habitación

Identificador:	API - 2	Petición: POST
<b>Proceso:</b> Habitaciones <b>Acción:</b> Registrar Habitación	<b>Descripción API:</b> Crea un nuevo registro en el modelo [Habitación].	
	POST	http://127.0.0.1:8000/reservar/v1/HabitacionGuardar/
<b>Body:</b> <pre>{   "descripcion": "Habitacion Nro 1",   "numero": 1,   "piso": 1,   "precio_dia": "50.000",   "estado_habitacion": "Bueno" }</pre>		

### API – 3 Edita los datos de una Habitación

Identificador:	API - 3	Petición: PUT
<b>Proceso:</b> Habitaciones <b>Acción:</b> Editar Habitación	<b>Descripción API:</b> Edita los datos en el modelo [Habitación].	
	PUT	http://127.0.0.1:8000/reservar/v1/HabitacionEditar/{id_habitacion}/
<b>Parámetros:</b> habitación que será editado {id_habitacion} <b>Body:</b> datos que serán editados		
<pre>{   "descripcion": "Habitacion Nro 1",   "numero": 5,   "piso": 3,   "precio_dia": "50.000",   "estado_habitacion": "Bueno" }</pre>		

#### API – 4 Elimina Habitación

Identificador:	API - 6	Petición: PUT
Proceso: Habitaciones Acción: Eliminar Habitación	<b>Descripción API:</b> Elimina una habitación del modelo [Habitación].	
		<b>DELETE</b> <a href="http://127.0.0.1:8000/reservar/v1/HabitacionEliminar/{id_habitacion}/">http://127.0.0.1:8000/reservar/v1/HabitacionEliminar/{id_habitacion}/</a>
Parámetros: habitación que será eliminado {id_habitacion}		

#### API – 5 Lista Clientes

Identificador:	API - 5	Petición: GET
Proceso: Clientes Acción: Listar clientes	<b>Descripción API:</b> Devuelve todos los clientes registrados en el hotel del modelo [Cliente]	
		<b>GET</b> <a href="http://127.0.0.1:8000/reservar/v1/listaCliente/">http://127.0.0.1:8000/reservar/v1/listaCliente/</a>
<pre>[   {     "id_cliente": 1,     "nombre": "William",     "apellido_paterno": "Luque",     "apellido_materno": "Rojas",     "numero_identificacion": "78547114",     "direccion": "Zona. Ballivian, Calle. Potosi, Nro. 158",     "telefono": 61212476,     "celular": 61212476   },   {     "id_cliente": 2,     "nombre": "Andrea",     "apellido_paterno": "Amusquivar",     "apellido_materno": "Baldivezo",     "numero_identificacion": "45784747",     "direccion": "Zona. Villa Dolores, Calle. Sucre, Nro 1954",     "telefono": 71589946,     "celular": 71589946   } ]</pre>		Parámetros: (sin parámetros)

#### API – 6 Registra Cliente

Identificador:	API - 2	Petición: POST
Proceso: Clientes Acción: Registrar Cliente	<b>Descripción API:</b> Crea un nuevo registro en el modelo [Cliente].	
		<b>POST</b> <a href="http://127.0.0.1:8000/reservar/v1/ClienteGuardar/">http://127.0.0.1:8000/reservar/v1/ClienteGuardar/</a>
<b>Body:</b> <pre>{   "nombre": "Maria",   "apellido_paterno": "Huampo",   "apellido_materno": "Quispe",   "numero_identificacion": "12514470",   "direccion": "Zona. Los Andes, Calle. Santa Cruz, Nro. 150",   "telefono": 7626487,   "celular": 76262548 }</pre>		

#### API – 7 Edita los datos de un Cliente

Identificador:	API - 7	Petición: PUT
Proceso: Clientes Acción: Editar Cliente	<b>Descripción API:</b> Edita los datos en el modelo [Cliente].	
		<b>PUT</b> <a href="http://127.0.0.1:8000/reservar/v1/ClienteEditar/{id_cliente}/">http://127.0.0.1:8000/reservar/v1/ClienteEditar/{id_cliente}/</a>
Parámetros: cliente que será editado {id_cliente} Body: datos que serán editados <pre>{     "nombre": "Ana Maria",     "apellido_paterno": "Huampo",     "apellido_materno": "Quispe",     "numero_identificacion": "12514470",     "direccion": "Zona. Los Andes, Calle. Santa Cruz, Nro. 150",     "telefono": 7626487,     "celular": 76262548   }</pre>		

API – 8 Elimina Cliente

Identificador:	API - 8	Petición: PUT
<b>Proceso:</b> Clientes <b>Acción:</b> Eliminar Cliente	<b>Descripción API:</b> Elimina un cliente del modelo [Cliente].	
	DELETE	http://127.0.0.1:8000/reservar/v1/ClienteEliminar/{id_cliente}/
	Parámetros: cliente que será eliminado {id_cliente}	

API – 9 Lista Reservas

Identificador:	API - 9	Petición: GET
<b>Proceso:</b> Reservar <b>Acción:</b> Listar reservas	<b>Descripción API:</b> Devuelve todas las reservas realizadas en el hotel del modelo [Reserva]	
	GET	http://127.0.0.1:8000/reservar/v1/listaReserva/
<pre>[   {     "id_reserva": 3,     "id_cliente": {       "id_cliente": 2,       "nombre": "Andrea",       "apellido_paterno": "Amusquivar",       "apellido_materno": "Baldiviezo",       "numero_identificacion": "45784747",       "direccion": "Zona. Villa Dolores, Calle. Sucre, Nro 1054",       "telefono": 71509946,       "celular": 71509946     },     "id_habitacion": {       "id_habitacion": 1,       "descripcion": "Habitacion Nro 1",       "numero": 1,       "piso": 1,       "precio_dia": "50.000",       "estado_habitacion": "Bueno"     },     "fecha_entrada": "2022-03-18",     "fecha_salida": "2022-03-13",     "dias_estadia": 15,     "precio_total": "360.000",     "estado": "Pendiente"   },   {     "id_reserva": 1,     "id_cliente": {       "id_cliente": 1,       "nombre": "William",       "apellido_paterno": "Luque",       "apellido_materno": "Rojas",       "numero_identificacion": "78547114",       "direccion": "Zona. Ballivian, Calle. Potosi, Nro. 150",       "telefono": 61212476,       "celular": 61212476     },     "id_habitacion": {       "id_habitacion": 1,       "descripcion": "Habitacion Nro 1",       "numero": 1,       "piso": 1,       "precio_dia": "50.000",       "estado_habitacion": "Bueno"     },     "fecha_entrada": "2022-03-18",     "fecha_salida": "2022-03-07",     "dias_estadia": 7,     "precio_total": "250.000",     "estado": "Pendiente"   } ]</pre>		Parámetros: (sin parámetros)

API – 10 Verificar si existe cliente

Identificador:	API - 10	Petición: GET
<b>Proceso:</b> Reservar <b>Acción:</b> Verificar si existe cliente	<b>Descripción API:</b> Devuelve los datos de un cliente si existe en el modelo [Cliente]	
	<b>GET</b> <a href="http://127.0.0.1:8000/reservar/v1/ClienteBuscar/{numero_identificacion}/">http://127.0.0.1:8000/reservar/v1/ClienteBuscar/{numero_identificacion}/</a>	
<pre>{   "id_cliente": 2,   "nombre": "Andrea",   "apellido_paterno": "Amusquivar",   "apellido_materno": "Baldiviezo",   "numero_identificacion": "45784747",   "direccion": "Zona. Villa Dolores, Calle. Sucre, Nro 1054",   "telefono": 71509946,   "celular": 71509946 }</pre>		<b>Parámetros:</b> {numero_identificacion} En el ejemplo numero_identificacion = 45784747

API – 10 Verificar si existe reserva en fecha X

Identificador:	API - 11	Petición: GET
<b>Proceso:</b> Reservar <b>Acción:</b> Verificar si existe reserva	<b>Descripción API:</b> Devuelve los datos de una reserva con fecha X si existe en el modelo [Reserva]	
	<b>GET</b> <a href="http://127.0.0.1:8000/reservar/v1/ReservaBuscar/{fecha_entrada}/">http://127.0.0.1:8000/reservar/v1/ReservaBuscar/{fecha_entrada}/</a>	
<pre>{   "id_reserva": 1,   "fecha_entrada": "2022-03-18",   "fecha_salida": "2022-03-07",   "dias_estadia": 7,   "precio_total": "250.000",   "estado": "Pendiente",   "id_habitacion": 1,   "id_cliente": 1 }</pre>		<b>Parámetros:</b> {fecha_entrada} En el ejemplo Fecha_entrada = 2022-03-18

API – 12 Registrar Reserva


Identificador:	API - 12	Petición: POST
<b>Proceso:</b> Reservar <b>Acción:</b> Registrar Reserva	<b>Descripción API:</b> Crea un nuevo registro en el modelo [Reserva].	
	<div>POSThttp://127.0.0.1:8000/reservar/v1/ReservaGuardar/</div>	
	<b>Body:</b> <pre>{   "fecha_entrada": "2022-03-18",   "fecha_salida": "2022-03-07",   "dias_estadia": 7,   "precio_total": "250.000",   "estado": "Pendiente",   "id_habitacion": 1,   "id_cliente": 1}</pre>	










## API – 13 Registrar Facturación

Identificador:	API - 13	Petición: POST
Proceso: Reservar Acción: Registrar Factura	Descripción API: Crea un nuevo registro en el modelo [Factura].	
	POST	<a href="http://127.0.0.1:8000/reservar/v1/FacturaGuardar/">http://127.0.0.1:8000/reservar/v1/FacturaGuardar/</a>
	Body: <pre>{   "id_factura": 1,   "fecha": "2022-03-18",   "id_cliente": 1,   "id_reserva": 1,   "id_modopago": 3 }</pre>	

Django Los endpoints descritos están realizados con Django REST Framework.

The screenshot displays the Django administration interface. At the top, the header shows 'Django administration' and a welcome message for 'TUGERENTE'. Below the header, the 'Site administration' section is visible, containing two main categories: 'AUTHENTICATION AND AUTHORIZATION' and 'RESERVAR'. Under 'AUTHENTICATION AND AUTHORIZATION', there are links for 'Groups' and 'Users', each with '+ Add' and 'Change' options. Under 'RESERVAR', there are links for 'Clientes', 'Facturas', 'Habitaciones', 'Modo\_Pago', and 'Reservas', each with '+ Add' and 'Change' options. On the right side, the 'Recent actions' section is shown, listing 'My actions' with a list of recent operations: 'Factura object (2)', 'Factura object (1)', 'Reserva object (2)', 'Habitacion object (5)', 'Habitacion object (4)', 'Habitacion object (3)', 'Habitacion object (2)', 'Habitacion object (1)', 'Cliente object (4)', and 'Cliente object (3)'. Each action is accompanied by a green plus icon and a yellow pencil icon.

← → ↻ 127.0.0.1:8000/reservar/v1/ 

        »  Reading list

## Page not found (404)

Request Method: GET  
Request URL: <http://127.0.0.1:8000/reservar/v1/>

Using the URLconf defined in `tu_gerente.urls`, Django tried these URL patterns, in this order:

1. `admin/`
2. `admin2/`
3. `reservar/ v1/ listaCliente/ [name='ListaCliente']`
4. `reservar/ v1/ ClienteGuardar/ [name='ClienteGuardar']`
5. `reservar/ v1/ ClienteEditar/<int:pk>/ [name='ClienteEditar']`
6. `reservar/ v1/ ClienteEliminar/<int:pk>/ [name='ClienteEliminar']`
7. `reservar/ v1/ ClienteBuscar/<str:busca>/ [name='ClienteBuscar']`
8. `reservar/ v1/ listaHabitacion/ [name='ListaHabitacion']`
9. `reservar/ v1/ HabitacionGuardar/ [name='HabitacionGuardar']`
10. `reservar/ v1/ HabitacionEditar/<int:pk>/ [name='HabitacionEditar']`
11. `reservar/ v1/ HabitacionEliminar/<int:pk>/ [name='HabitacionEliminar']`
12. `reservar/ v1/ listaReserva/ [name='ListaReserva']`
13. `reservar/ v1/ ReservaGuardar/ [name='ReservaGuardar']`
14. `reservar/ v1/ ReservaEditar/<int:pk>/ [name='ReservaEditar']`
15. `reservar/ v1/ ReservaEliminar/<int:pk>/ [name='ReservaEliminar']`
16. `reservar/ v1/ ReservaBuscar/<str:fecha>/ [name='ReservaBuscar']`
17. `reservar/ v1/ listaReserva/ [name='ListaReserva']`
18. `reservar/ v1/ ReservaGuardar/ [name='ReservaGuardar']`
19. `reservar/ v1/ ReservaEditar/<int:pk>/ [name='ReservaEditar']`
20. `reservar/ v1/ ReservaEliminar/<int:pk>/ [name='ReservaEliminar']`
21. `reservar/ v1/ listaModoPago/ [name='ListaModoPago']`
22. `reservar/ v1/ ModoPagoGuardar/ [name='ModoPagoGuardar']`
23. `reservar/ v1/ ModoPagoEditar/<int:pk>/ [name='ModoPagoEditar']`
24. `reservar/ v1/ ModoPagoEliminar/<int:pk>/ [name='ModoPagoEliminar']`
25. `reservar/ v1/ listaFactura/ [name='ListaFactura']`
26. `reservar/ v1/ FacturaGuardar/ [name='FacturaGuardar']`
27. `reservar/ v1/ FacturaEditar/<int:pk>/ [name='FacturaEditar']`
28. `reservar/ v1/ FacturaEliminar/<int:pk>/ [name='FacturaEliminar']`

The current path, `reservar/v1/`, didn't match any of these.

You're seeing this error because you have `DEBUG = True` in your Django settings file. Change that to `False`, and Django will display a standard 404 page.

**El código fuente esta alojado en:** [https://github.com/luzmar7/backend\\_tugente](https://github.com/luzmar7/backend_tugente)