



# Documentación de Servicios REST

**Materia:** Arquitectura de Servicios

**Alumnos:** Gonzales Herrera Cristian  
Garcia Garcia Braulio Yosef

**Carrera:** Ingeniería en Tecnologías de la Información y Comunicaciones

**Especialidad:** Desarrollador Fullstack

**Facilitador(a):** Roberto Suarez Zinzun

**Semestre:** 8° A

**Turno:** Matutino

# Índice.

<b>Servicios REST Citas .....</b>	<b>2</b>
Operación: Generar Cita.....	2
Operación: Consultar Citas.....	3
Operación: Consultar Detalles de una Cita .....	3
Operación: Confirmar Cita .....	4
Operación: Cancelar Cita.....	4
Operación: Agregar Dispositivos .....	5
Operación: Consultar Dispositivos .....	5
Operación: Consultar Detalles Dispositivo .....	6
Operación: Actualizar Datos Dispositivos .....	6
Operación: Eliminar Dispositivos.....	6
<b>Servicios REST Reparación.....</b>	<b>7</b>
Operación: Agregar Reparación .....	7
Operación: Consultar Reparaciones.....	8
Operación: Consultar Detalles de una Reparación .....	9
Operación: Actualizar Datos Reparación.....	9
<b>Servicios REST Refacción .....</b>	<b>10</b>
Operación: Agregar Refacción.....	10
Operación: Consultar Refacciones .....	11
Operación: Consultar Detalles Refacción .....	11
Operación: Actualizar Datos Refacción .....	11
Operación: Eliminar Refacción .....	12
<b>Servicios REST Usuario .....</b>	<b>12</b>
Operación: Registrar Usuario .....	13
Operación: Consultar Usuarios.....	13
Operación: Consultar Detalles de Usuario .....	14
Operación: Actualizar Usuario.....	14
Operación: Eliminar Usuario .....	15
Operación: Validar Credenciales .....	15

## Servicios REST Citas.

**Descripción:** Este servicio tiene la función de realizar las operaciones de generar, visualizar, confirmar y eliminar las Citas almacenadas en la base de datos.

**Tipo de Servicio:** Es considerado un servicio de Tarea y Utilidad ya que se usará para generar las citas, así como dar soporte a las operaciones CRUD (Create, Read, Update, Delete) de las mismas.

### Operaciones:

1. Generar Cita.
2. Consultar Citas.
3. Consultar Detalles de una Cita.
4. Confirmar Cita.
5. Cancelar Cita.
6. Agregar Dispositivos.
7. Consultar Dispositivos.
8. Consultar Detalles de un Dispositivo.
9. Actualizar Datos Dispositivos.
10. Eliminar Dispositivos.

### Operación: Generar Cita.

Elemento	Valor
Método de Acceso	POST
Ruta	/citas
Protocolo	HTTP
Actor(es)	Clientes
Entrada	<pre>{   "idCita": int,   "nombreCliente": string,   "dispositivo": [     {       "idDispositivo": int,       "marca": string,       "modelo": string,       "característicasHardware": text,       "fallas": text,       "fotosDispositivo": img     }   ] }</pre>

	<pre> "horaCita": time, } </pre>
Proceso	Debe realizar el registro de una nueva Cita en la base de datos, para ello además de los datos de entrada, también debe generar de manera automática un campo llamado: fechaRegistro, usando la fecha del momento de registro de la cita como valor para este campo.
Salida	<pre> {   "estatus": boolean,   "fechaEntrega": date } </pre>

### Operación: Consultar Citas.

Elemento	Valor
Método de Acceso	GET
Ruta	/citas
Protocolo	HTTP
Actor(es)	Clientes, Administradores
Entrada	N/A
Proceso	Recuperar todas las citas registradas en la base de datos.
Salida	<pre> {   "citas": [     {       "idCita": int,       "nombreCliente": string,       "horaCita": time,       "fechaRegistro": date,       "fechaEntrega": date,       "estatus": string     }   ] } </pre>

### Operación: Consultar Detalles de una Cita.

Elemento	Valor
Método de Acceso	GET
Ruta	/citas/{idCita}
Protocolo	HTTP
Actor(es)	Clientes, Administrador
Entrada	idCita (int) en la consulta
Proceso	Recuperar los detalles de una cita por su ID

Salida	<pre>{   "idCita": int,   "nombreCliente": string,   "horaCita": time,   "fechaRegistro": date,   "fechaEntrega": date,   "estatus": string,   "dispositivo": {     "idDispositivo": int,     "marca": string,     "modelo": string,     "característicasHardware": text,     "fallas": text,     "fotosDispositivo": img   } }</pre>
--------	---

### Operación: Confirmar Cita.

Elemento	Valor
Método de Acceso	PUT
Ruta	/citas/{idCita}/confirmar
Protocolo	HTTP
Actor(es)	Administrador
Entrada	idCita (int) en la consulta
Proceso	Confirmar una cita específica según su ID.
Salida	<pre>{   "estatus": boolean,   "mensaje": string }</pre>

### Operación: Cancelar Cita.

Elemento	Valor
Método de Acceso	DELETE
Ruta	/citas/{idCita}/eliminar
Protocolo	HTTP
Actor(es)	Cliente, Administrador
Entrada	idCita (int) en la consulta
Proceso	Cancelar una cita específica según su ID, cambiando su estado a "Cancelada"
Salida	<pre>{   "estatus": boolean,   "mensaje": string }</pre>

	}
--	---

### Operación: Agregar Dispositivos.

Elemento	Valor
Método de Acceso	POST
Ruta	/dispositivos
Protocolo	HTTP
Actor(es)	Cliente
Entrada	{ "marca": string, "modelo": string, "característicasHardware": text, "fallas": text, "fotosDispositivo": img }
Proceso	Agregar un nuevo dispositivo al sistema
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Consultar Dispositivos.

Elemento	Valor
Método de Acceso	GET
Ruta	/dispositivos
Protocolo	HTTP
Actor(es)	Cliente
Entrada	N/A
Proceso	Recuperar todos los dispositivos registrados en el sistema
Salida	{ "dispositivos": [ { "idDispositivo": int, "marca": string, "modelo": string, "característicasHardware": text, "fallas": text, "fotosDispositivo": img } ] }

### Operación: Consultar Detalles de un Dispositivo.

Elemento	Valor
Método de Acceso	GET
Ruta	/dispositivos/{idDispositivo}
Protocolo	HTTP
Actor(es)	Cliente
Entrada	idDispositivo (int) en la consulta.
Proceso	Recuperar los detalles de un dispositivo registrado en la base de datos por su id.
Salida	{ "idDispositivo": int, "marca": string, "modelo": string, "característicasHardware": text, "fallas": text, "fotosDispositivo": img }

### Operación: Actualizar Datos Dispositivos.

Elemento	Valor
Método de Acceso	PUT
Ruta	/dispositivos/{idDispositivo}
Protocolo	HTTP
Actor(es)	Cliente
Entrada	{ "marca": string, "modelo": string, "característicasHardware": text, "fallas": text, "fotosDispositivo": img }
Proceso	Actualizar los detalles de un dispositivo según su ID
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Eliminar Dispositivos.

Elemento	Valor
Método de Acceso	DELETE

Ruta	/dispositivos/{idDispositivo}
Protocolo	HTTP
Actor(es)	Clientes
Entrada	idDispositivo (int) en la consulta
Proceso	Eliminar un dispositivo del sistema según su ID
Salida	{ "estatus": boolean, "mensaje": string }

## Servicios REST Reparación.

**Descripción:** Este servicio tiene la función de realizar las operaciones de confirmar, actualizar y visualizar los detalles de las Reparaciones almacenadas en la base de datos.

**Tipo de Servicio:** Es considerado un servicio de Tarea ya que esta únicamente enfocado en las operaciones de la Reparación.

### Operaciones:

1. Agregar Reparación.
2. Consultar Reparaciones.
3. Consultar Detalles de una Reparación.
4. Actualizar Datos Reparación.
5. Agregar Refacción.
6. Consultar Refacciones.
7. Consultar Detalles de una Refacción.
8. Actualizar Datos Refacción.
9. Eliminar Refacción.

### Operación: Agregar Reparación.

Elemento	Valor
Método de Acceso	POST
Ruta	/reparaciones
Protocolo	HTTP
Actor(es)	Administrador
Entrada	{ "idReparacion": int, "idCita": int,



	<pre> "fechaInicio": date, "fechaFin": date, "estado": string, "descripcion": text, "costo": money, "refaccionesUtilizadas": [   {     "idRefaccion": int,     "nombreRefaccion": string,     "cantidad": int   } ] </pre>
Proceso	Agregar una reparación cuando ha sido atendida una cita y se decidió hacer la reparación
Salida	<pre> {   "estatus": boolean,   "mensaje": string } </pre>

### Operación: Consultar Reparaciones.

Elemento	Valor
Método de Acceso	GET
Ruta	/reparaciones
Protocolo	HTTP
Actor(es)	Administrador
Entrada	N/A
Proceso	Recuperar los detalles de todas las reparaciones en la base de datos
Salida	<pre> {   "reparaciones": [     "idReparacion": int,     "idCita": int,     "fechaInicio": date,     "fechaFin": date,     "estado": string,     "descripcion": string,     "costo": decimal,     "refaccionesUtilizadas": [       {         "idRefaccion": int,         "nombre": string,         "cantidad": int       }     ]   ] } </pre>

```

    ]
  ]
}

```

### Operación: Consultar Detalles de una Reparación.

Elemento	Valor
Método de Acceso	GET
Ruta	/reparaciones/{idReparación}
Protocolo	HTTP
Actor(es)	Administrador
Entrada	idReparación (int) en la consulta
Proceso	Recuperar los detalles de una reparación específica según su ID
Salida	<pre> {   "idReparacion": int,   "idCita": int,   "fechaInicio": date,   "fechaFin": date,   "estado": string,   "descripcion": string,   "costo": decimal,   "refaccionesUtilizadas": [     {       "idRefaccion": int,       "nombre": string,       "cantidad": int     }   ] } </pre>

### Operación: Actualizar Datos Reparación.

Elemento	Valor
Método de Acceso	PUT
Ruta	/reparaciones/{idReparación}
Protocolo	HTTP
Actor(es)	Actualizar
Entrada	<pre> {   "idReparación": int,   "costoTotal": float } </pre>
Proceso	Actualizar los detalles de una reparación específica según su ID

Salida	{ "estatus": boolean, "mensaje": string }
--------	--

## Servicios REST Refacción.

**Descripción:** Este servicio tiene la función de realizar las operaciones de agregar, visualizar, actualizar y eliminar los datos de Refacciones almacenados en la base de datos.

**Tipo de Servicio:** Es considerado un servicio de Entidad ya que representa los datos de objetos reales, por lo que hace uso de las operaciones CRUD (Create, Read, Update, Delete) de los mismos.

### Operaciones:

1. Agregar Refacción.
2. Consultar Refacciones.
3. Consultar Detalles de una Refacción.
4. Actualizar Datos Refacción.
5. Eliminar Refacción.

### Operación: Agregar Refacción.

Elemento	Valor
Método de Acceso	POST
Ruta	/refacciones
Protocolo	HTTP
Actor(es)	Administrador
Entrada	{ "idRefaccion": int, "nombre": string, "cantidad": int, "precioUnitario": float, "descripcion": text }
Proceso	Agregar una nueva refacción al inventario
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Consultar Refacciones.

Elemento	Valor
Método de Acceso	GET
Ruta	/refacciones
Protocolo	HTTP
Actor(es)	Administrador
Entrada	N/A
Proceso	Recuperar todas las refacciones disponibles en el inventario
Salida	<pre>{   "refacciones": [     {       "idRefaccion": int,       "nombre": string,       "cantidad": int,       "precioUnitario": float,       "descripcion": text     }   ] }</pre>

### Operación: Consultar Detalles de una Refacción.

Elemento	Valor
Método de Acceso	GET
Ruta	/refacciones/{idRefaccion}
Protocolo	HTTP
Actor(es)	Administrador
Entrada	idRefaccion (int) en la consulta.
Proceso	Recuperar los detalles de una refacción registrada en la base de datos por su id.
Salida	<pre>{   "idRefaccion": int,   "nombre": string,   "cantidad": int,   "precioUnitario": float,   "descripcion": text }</pre>

### Operación: Actualizar Datos Refacción.

Elemento	Valor
----------	-------

Método de Acceso	PUT
Ruta	/refacciones/{idRefaccion}
Protocolo	HTTP
Actor(es)	Administrador
Entrada	{ "idRefaccion": int, "nombre": string, "cantidad": int, "precioUnitario": decimal }
Proceso	Actualizar los detalles de una refacción en el inventario según su ID
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Eliminar Refacción.

Elemento	Valor
Método de Acceso	DELETE
Ruta	/refacciones/{idRefaccion}
Protocolo	HTTP
Actor(es)	Administrador
Entrada	idRefaccion (int) en la consulta
Proceso	Eliminar una refacción del inventario según su ID
Salida	{ "estatus": boolean, "mensaje": string }

## Servicios REST Usuario.

**Descripción:** Este servicio tiene la función de realizar las operaciones de generación, visualización, actualización, eliminación y consultas de los Usuarios almacenados en la base de datos.

**Tipo de Servicio:** Es considerado un servicio de Entidad y Utilidad ya que se usará en la validación de credenciales de Usuarios, así como dar soporte a las operaciones CRUD (Create, Read, Update, Delete) de los mismos.

### Operaciones:

1. Registrar Usuario.
2. Consultar Usuarios.
3. Consultar Detalles de un Usuario.
4. Actualizar Perfil.
5. Eliminar Perfil.
6. Validar Credenciales.

### Operación: Registrar Usuario.

Elemento	Valor
Método de Acceso	POST
Ruta	/usuarios
Protocolo	HTTP
Actor(es)	Cliente, Administrador
Entrada	{ "nombre": string, "apellidos": string, "telefono": string, "correoElectronico": string, "contraseña": string, "rol": string }
Proceso	Registrar un nuevo usuario en el sistema
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Consultar Usuarios.

Elemento	Valor
Método de Acceso	GET
Ruta	/usuarios
Protocolo	HTTP
Actor(es)	Cliente, Administrador
Entrada	N/A
Proceso	Recuperar los detalles de todos los usuarios registrados en la base de datos.
Salida	{ "usuarios": [ { "idUsuario": int,

```

    "nombre": string,
    "apellidos": string,
    "telefono": string,
    "correoElectronico": string,
    "contraseña": string,
    "rol": string
  }
]
}

```

### Operación: Consultar Detalles de un Usuario.

Elemento	Valor
Método de Acceso	GET
Ruta	/usuarios/{idUserario}
Protocolo	HTTP
Actor(es)	Cliente, Administrador
Entrada	idUserario (int) en la consulta
Proceso	Recuperar los detalles del perfil de usuario según su ID.
Salida	<pre> {   "idUserario": int,   "nombre": string,   "apellidos": string,   "telefono": string,   "correoElectronico": string,   "contraseña": string,   "rol": string } </pre>

### Operación: Actualizar Perfil.

Elemento	Valor
Método de Acceso	PUT
Ruta	/usuarios/{idUserario}
Protocolo	HTTP
Actor(es)	Cliente, Administrador
Entrada	<pre> {   "nombre": string,   "apellidos": string,   "telefono": string,   "correoElectronico": string,   "contraseña": string } </pre>
Proceso	Actualizar los detalles del perfil de usuario según su ID

Salida	{ "estatus": boolean, "mensaje": string }
--------	--

### Operación: Eliminar Perfil.

Elemento	Valor
Método de Acceso	DELETE
Ruta	/usuarios/{idUserio}
Protocolo	HTTP
Actor(es)	Usuarios (Cliente, Administrador)
Entrada	idUserio (int) en la consulta
Proceso	Elimina el perfil de un usuario según su ID
Salida	{ "estatus": boolean, "mensaje": string }

### Operación: Validar Credenciales.

Elemento	Valor
Método de Acceso	POST
Ruta	/usuarios/validar
Protocolo	HTTP
Actor(es)	Sistema
Entrada	{ "nombreUsuario": string, "contraseña": string }
Proceso	Validar las credenciales del usuario
Salida	{ "estatus": boolean, "mensaje": string }