# **Documentación del Servicio REST/API**

## **Controladores**

**ClientController:** Para acceder en el HTML se carga la dirección <http://localhost:55100/api/client>. Este controlador se encarga de la creación, modificación, eliminación y comprobación de los clientes.

* Iniciar Sesión: Para iniciar sesión se hace un POST desde el html en formato JSON con el código C00, el nombre de usuario(cédula) y contraseña. El Servicio Web retorna el código 200 junto con la información del usuario solicitado, retorna un código de Error01 si el usuario no existe o Error02 si la contraseña es incorrecta.

Entrada:

* codigo = “C00”
  + {“user”:1111111, “pass”:”xxxxxxx”}

Salidas:

* + {“codigo”:200, “Nombre1”:”xxx”, “Nombre2”:”xxx”, ….}
  + {“codigo”:”201”, “descripcion”:”Error”}
* Crear Cliente: Para crear un usuario se hace un POST desde el html en formato JSON con el código C01, el nombre1, nombre2, apellido1, apellido2, cedula, fNacimiento, contraseña, provincia, canton, distrito, indicaciones, telefono. El Servidor Web retorna el código 200 si la inserción a la base de datos se completó correctamente, retorna 201 y una descripción del error si algo sale mal durante la inserción.

Entrada:

* codigo = “C01”
  + {“nombre1”:”xxx”, “nombre2”:”xxx”, ….}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”descrip”:”xxxxx”}
* Eliminar Cliente: Para eliminar un usuario se hace un POST desde el html con el código C03 y la cédula y retorna 200 si la eliminación fue un éxito de lo contrario retorna 201 y una descripción del error.

Entradas:

* codigo = “C02”
  + {“cedula”:”xxxxxxx”}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”descrip”:”xxxxx”}
* Actualizar Cliente: Para actualizar un usuario se hace un PUT desde el html con el código C02 y todos los datos del cliente, incluyendo los que no se modificaron y retorna 200 si la actualización fue un éxito de lo contrario retorna 201 y una descripción del error.

Entradas:

* codigo = “C02”
  + {“nombre1”:”xxx”, “nombre2”:”xxx”, ….}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”descrip”:”Insercion Exitosa”}

**EmpleadoController:** Para acceder en el HTML se carga la dirección <http://localhost:55100/api/empleado> . Este controlador se encarga de la creación, modificación, eliminación de los empleados.

* Obtener todos los empleados: Para obtener todos los empleados se realiza un POST con el código E00 y un JSON con el ID de la compañía de donde se quieren obtener los empleados. El Servicio Web retorna el código 200 junto con la información solicitada en formato JSON, retorna un código 201 si sucede algún error durante la búsqueda.

Entradas:

* codigo = “E00”
  + {“comp” = X}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”Empleados”:[{JSON}] }
* Insertar Empleado: Para insertar un empleado se realiza un POST con el código E01 y un JSON el nombre1, nombre2, apellido1, apellido2, cedula, fNacimiento, contraseña, provincia, canton, distrito, indicaciones, teléfono, rol, sucursal, compañía. El Servicio Web retorna el código 200 si la inserción fue exitosa, retorna un código 201 si sucede algún error durante la búsqueda.

Entradas:

* codigo = “E01”
* {“nombre1”:”xxx”, “nombre2”:”xxx”, ….}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”descripción”:”Insercion exitosa” }
* Actualizar Empleado: Para actualizar un empleado se hace un PUT desde el html con el código E02 y todos los datos del empleado, incluyendo los que no se modificaron y retorna 200 si la actualización fue un éxito de lo contrario retorna 201 y una descripción del error.

Entradas:

* codigo = “E02”
  + {“nombre1”:”xxx”, “nombre2”:”xxx”, ….}

Salidas:

* + {“codigo”:”201”, “descripcion”:”Error”}
  + {“codigo”:”200”:”descrip”:”Insercion Exitosa”}
* Iniciar Sesión: Para iniciar sesión se hace un POST desde el html en formato JSON con el código E04, el nombre de usuario(cédula) y contraseña. El Servicio Web retorna el código 200 junto con la información del empleado solicitado, retorna un código de Error01 si el usuario no existe o Error02 si la contraseña es incorrecta.

Entrada:

* codigo = “E04”
  + {“user”:1111111, “pass”:”xxxxxxx”}

Salidas:

* + {“codigo”:200, “Nombre1”:”xxx”, “Nombre2”:”xxx”, ….}
  + {“codigo”:”201”, “descripcion”:”Error”}