



“Un hombre con una idea nueva es un loco hasta que la idea triunfa” (Mark Twain)

Como empresa, **Alquimiasoft** tiene el objetivo de crear soluciones tecnológicas innovadoras, que aporten al crecimiento de pequeños y medianos negocios. Buscamos estar presentes con nuestras soluciones tecnológicas empresariales en 1M de negocios de Latinoamérica.

Como producto, **Mi negocio** es un sistema contable y de facturación electrónica, que optimiza la administración y control de las pequeñas y medianas empresas en el Ecuador.

Te invitamos a formar parte de este equipo conformado por profesionales comprometidos no solo con el crecimiento de la institución sino con el desarrollo de nuestro país.

Para el desarrollo del ejercicio deberá crear un repositorio público para compartir el código a realizar.

Objetivo:

Desarrollar un servicio con el lenguaje java en su versión 8 o superior, que se conecte a una base de datos local y que exponga un listado de API REST.

Indicaciones:

- Utilizar las siguientes tecnologías
 - Lenguaje java 8+
 - Spring boot
 - PostgreSQL
- Aplicar las mejores prácticas, patrones, estándares, etc. que considere (se toma muy en cuenta el uso de patrones, buenas prácticas y estándares).
- Implementar pruebas unitarias con JUnit y Mockito
- Subir el código a un repositorio GIT y enviarlo antes de la fecha y hora indicada

Historia de usuario

El sistema Mi negocio permite registrar de manera rápida una factura a consumidor final o cualquier otro cliente registrando los datos al momento de la facturación. Pero se necesita mejorar este proceso y es necesario tener una base de clientes por cada empresa que utiliza el sistema, de tal manera que al momento de facturar se pueda buscar un cliente por número de identificación o nombre.

Como usuario del sistema Mi negocio necesito poder registrar nuevos clientes, editarlos, eliminarlos y/o realizar una búsqueda. Adicional a esto, es necesario tomar en cuenta que un cliente puede tener más de una dirección o sucursal y se necesita poder registrar esas direcciones adicionales.

El cliente deberá tener asignado una dirección matriz. (No puede haber más de una dirección matriz).

Si el cliente desea que se le facture a otra dirección diferente a matriz se podrá seleccionar una dirección de un listado o registrar una nueva en el momento.

Implementar las siguientes funcionalidades:

- Funcionalidad para buscar y obtener un listado de clientes.
- Funcionalidad para crear un nuevo cliente con la dirección matriz
- Funcionalidad para editar los datos del cliente
- Funcionalidad para eliminar un cliente
- Funcionalidad para registrar una nueva dirección por cliente
- Funcionalidad para listar las direcciones adicionales del cliente

Criterios de aceptación

Caso: *Funcionalidad para buscar y obtener un listado de clientes*

Dado: que se necesita buscar un cliente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se le pueda indicar un parámetro de búsqueda (Número de identidad o Nombre) y retorne un listado de clientes, incluyendo la dirección matriz cuyos datos coincidan con los buscados.

Ejemplo de respuesta.



```
1 {
2   {
3     "id": null,
4     "identificationType": "DNI",
5     "identificationNumber": "0505555555",
6     "names": "Daniel Sandoval",
7     "email": "daniel@test.com",
8     "cellphone": "0999999999",
9     "mainProvince": "Cotopaxi",
10    "mainCity": "Salcedo",
11    "mainAddress": "Calle Ana paredes"
12  },
13  {
14    "id": null,
15    "identificationType": "RUC",
16    "identificationNumber": "0505555555001",
17    "names": "Marcelo Sandoval",
18    "email": "marcelo@test.com",
19    "cellphone": "0999999999",
20    "mainProvince": "Cotopaxi",
21    "mainCity": "Latacunga",
22    "mainAddress": "el salto"
23  }
24 }
```

Nota: los campos y datos de ejemplo son una guía. Si lo requiere puede agregar o reemplazar campos, siempre y cuando el resultado siga cumpliendo el requerimiento.

Caso: *Funcionalidad para crear un nuevo cliente con la dirección matriz*

Dado: que se necesita crear un nuevo cliente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se envíe un objeto json con los datos para que el servicio lo guarde y retorne el cliente creado, validando que no exista más de un cliente con el mismo número de identificación.

Nota: Este proceso crea un nuevo cliente y registra la dirección matriz. Si lo cree pertinente puede agregar validaciones adicionales al proceso.

Caso: *Funcionalidad para editar los datos del cliente*

Dado: que se necesite editar los datos de un cliente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se envíe un objeto json con los datos del cliente para que el servicio los actualice y retorne el cliente actualizado, validando que con la actualización no exista más de un cliente con el mismo número de identificación.

Nota: Este proceso solo requiere que se pueda editar los datos del cliente. No es necesario una actualización de la dirección matriz o direcciones adicionales.

Caso: *Funcionalidad para eliminar un cliente*

Dado: que se necesite eliminar un cliente existente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se envíe el id del cliente para que el servicio lo elimine y retorne una respuesta de éxito.

Caso: *Funcionalidad para registrar una nueva dirección por cliente*

Dado: que se necesite registrar una dirección para un cliente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se envíe un objeto json para que el servicio guarde los datos y retorne una respuesta de éxito.

Caso: *Funcionalidad para listar las direcciones adicionales del cliente*

Dado: que se necesite conocer las direcciones de un cliente

Como: evaluador

Requiero: Que exista una API REST para llamarla con POSTMAN, en donde se envíe el id de un cliente para que el servicio retorne un listado con sus direcciones asignadas, incluyendo la dirección matriz.

Notas:

- La entidad clientes deberá tener los siguientes campos:
(id, tipo identificación (RUC o Cédula), número de identificación, nombres, correo y número de celular).
Si cree pertinente agregar o reemplazar campos puede hacerlo.
- La entidad dirección cliente o sucursal deberá tener los siguientes campos:
(id, provincia, ciudad, dirección)
Si cree pertinente agregar o reemplazar campos puede hacerlo.



- Todos los objetos retornados o enviados por parámetro deben ser en formato json.
- Se debe implementar buenas prácticas (clean code, SOLID)
- Se debe implementar TDD
- Está permitido utilizar herramientas de versionamiento de base datos. (Liquibase, otras)
- Está permitido utilizar herramientas o frameworks para la conexión y persistencia de datos. (Hibernate, JPA, Jdbc, otras)
- Si es necesario una configuración adicional para ejecutar la aplicación describirla en un archivo README