

# Prueba Práctica Desarrollador Back-End

## Descripción del problema

Una empresa necesita una solución para gestionar su lista de clientes de manera eficiente. Actualmente, enfrentan dificultades para agregar, actualizar, eliminar y consultar información de los clientes de forma rápida y segura.

Para resolver este desafío, se propone desarrollar un servicio RESTful que permita realizar operaciones CRUD sobre la lista de clientes. Este servicio proporcionará una interfaz uniforme y fácil de usar para gestionar todos los aspectos relacionados con los clientes, mejorando así la eficiencia y la calidad de la gestión de la información del cliente.

Es esencial para la empresa garantizar la validez y confiabilidad de los datos en todo momento. Por tanto, es necesario implementar medidas para validar los datos y evitar inconsistencias.

## Requisitos

A continuación los requisitos que debe de cumplir la solución:

### Requisitos Funcionales

1. Creación de un nuevo cliente.
2. Obtener todos los clientes existentes.
3. Obtener todos los clientes existentes que pertenezcan a un país específico.
4. Obtener un cliente específico por su identificador.
5. Actualizar un cliente existente, permitiendo solo la modificación de su correo, dirección, teléfono y país.
6. Eliminar un cliente por su identificador.

### Requisitos No Funcionales

1. Utilizar Java y el framework *Quarkus* para desarrollar el servicio.
2. Utilizar una base de datos para almacenar los clientes.
3. Incluir pruebas unitarias para las funciones principales.
4. Proporcionar un diagrama de arquitectura que muestre la estructura del proyecto.
5. Utilizar JSON para el intercambio de datos entre el cliente y el servidor.
6. Utilizar buenas prácticas de nomenclatura y estructura del proyecto.
7. Documentar la solución e incluir diagramas de apoyo.
8. El sistema debe mantener la integridad y consistencia de los datos almacenados.

## Datos de apoyo

- Un cliente tiene los siguientes campos:
  - Primer nombre **REQUERIDO**
  - Segundo nombre ( **OPCIONAL** )
  - Primer apellido **REQUERIDO**
  - Segundo apellido ( **OPCIONAL** )
  - Correo **REQUERIDO**
  - Dirección **REQUERIDO**
  - Teléfono **REQUERIDO**
  - País ( **REQUERIDO** , código *ISO 3166* )
  - Gentilicio (Buscar desde el servicio externo de países)
- El API disponible en <https://restcountries.com> proporciona la capacidad de obtener el gentilicio a partir del código del país.
- Para la elaboración de diagramas, se pueden utilizar herramientas como <https://mermaid.live>.

- *Quarkus* es un framework de desarrollo de aplicaciones Java diseñado para facilitar la creación de aplicaciones en la nube de manera eficiente y ágil (ver <https://quarkus.io>).

## Evaluación


La evaluación se basará en los siguientes criterios:

1. Estructura del proyecto y organización del código.
2. Calidad del código (claridad, legibilidad y buenas prácticas de programación).
3. Funcionalidad y cumplimiento de los requisitos.
4. Implementación correcta de las operaciones CRUD.
5. Implementación correcta de RESTful.
6. Documentación clara del código y explicación de las decisiones de diseño.
7. Manejo de errores, excepciones y validación de datos de entrada.
8. Calidad de las pruebas unitarias.

## Entrega

El candidato debe realizar las siguientes entregas mediante un enlace al repositorio de control de versiones ubicado en la plataforma de su preferencia:

1. Código fuente del proyecto.
2. Documentación detallada que incluya una descripción de la solución, instrucciones de configuración y uso del servicio, así como cualquier consideración de diseño o implementación relevante.
3. Diagramas que ilustren la estructura del proyecto, incluyendo las capas de la aplicación, las dependencias principales y la interacción entre los componentes.

 **Nota:** El candidato puede emplear cualquier biblioteca o herramienta que estime conveniente para llevar a cabo la tarea, siempre y cuando satisfaga los requisitos mencionados anteriormente.