



Centro Educativo Técnico Laboral Kinal

Materia: 5to Perito Informática

Profesor: Eduardo Hor

Sección: IN5CM

Manual de Usuario

Nombre: Josué Daniel Jolón Motta

Carné: 2022205

Fecha de Entrega:

06/02/2026



Manual de Usuario – Repuestos Automotrices

Introducción

Este manual explica cómo usar mi proyecto hecho en Spring Boot para manejar empleados de un sistema de repuestos automotrices. El sistema permite crear, ver, buscar, actualizar y eliminar empleados usando herramientas como Postman o el navegador.

Descripción General del Sistema

El proyecto funciona como una API REST. Esto significa que se trabaja usando enlaces (URLs) y métodos como GET, POST, PUT y DELETE. Toda la información de empleados se guarda en una base de datos y se puede manipular desde herramientas para probar APIs.

Requisitos para usar la aplicación

- Computadora con Windows, Linux o macOS
- Java JDK instalado
- IntelliJ IDEA
- Maven
- Base de datos MySQL
- Proyecto configurado correctamente

Inicio de la aplicación

1. Abrir el proyecto en IntelliJ IDEA
2. Ejecutar la clase principal del proyecto
3. Esperar que el sistema inicie correctamente
4. Usar la dirección <http://localhost:8080> para probar los endpoints

Endpoints del Sistema

Obtener empleados:

localhost:8080/api/empleados



Función:

- Devuelve todos los empleados registrados.

Buscar empleado por ID:

localhost:8080/api/empleados/{id}

Resultado:

- Muestra los datos del empleado.
- Si no existe devuelve error 404.

Crear empleado:

localhost:8080/api/empleados

Body JSON ejemplo:

```
{  
  "nombre_empleado": "Juan",  
  "apellido_empleado": "Perez",  
  "puesto_empleado": "Vendedor",  
  "email_empleado" : "juanperez@gmail.com"  
}
```

Actualizar empleado:

localhost:8080/api/empleados/{id}

Body JSON:

```
{  
  "nombre_empleado": "Carlos",  
  "apellido_empleado": "Lopez",  
  "puesto_empleado": "Administrador",  
  "email_empleado" : "carlolopez@gmail.com"  
}
```



Eliminar empleado:

localhost:8080/api/empleados/{id}

Recomendaciones de Uso

- Revisar bien el ID antes de actualizar o eliminar
- Enviar bien los datos en formato JSON
- Verificar que la base de datos esté activa
- Revisar errores en consola si algo falla

Conclusión

Este proyecto permite practicar cómo funcionan las APIs REST usando Spring Boot. También ayuda a entender cómo hacer un CRUD conectado a base de datos y cómo probar los endpoints usando herramientas externas.

Link de POSTMAN: <https://jolon-2022205-7672540.postman.co/workspace/jolon-2022205's-Workspace~ac7d40e5-4045-4b8b-8531-0d80d3a9fb32/collection/52082090-f5ff90f6-4f44-4b8e-9241-05dd6782ee31?action=share&creator=52082090>

Link de trello:

<https://trello.com/invite/b/6983f8d374b92b3ef2068137/ATTI81fc6ee5eae4905d2b2ead9069aad8bD8267C1E/ta5t3josuejolon>