Índice

[ENUNCIADO DEL PROBLEMA: 1](#_Toc72421237)

[ANÁLISIS DEL PROBLEMA: 1](#_Toc72421238)

[Pruebas y testing 4](#_Toc72421239)

# ENUNCIADO DEL PROBLEMA:

De parte de Kodigo Academia, se me ha solicitado la creación de un API para implementar una agenda de citas, dicha agenda debe manejar: contactos, teléfonos, emails y direcciones cada una con sus diferentes tipos, además de manejar las citas de cada usuario.

Para el desarrollo del problema se solicita hacerlo con el framework Spring Boot y la base de datos H2 por sus prestaciones de código Java.

# ANÁLISIS DEL PROBLEMA:

Para dicho desarrollo se utilizarán las siguientes herramientas de software:

* Spring Tool Suite 4
* Postman 8
* Open JDK 11
* Spring framework
* H2 Database

Según los datos solicitados procedemos a la creación de la base de datos, el schema creado es el siguiente:  
Diagrama

Descripción generada automáticamente

La base fue implementada en modo archivo para evitar la creación de esta cada vez que se ejecuta la aplicación, esto generó otros conflictos de conexiones múltiples que fueron resueltos con configuración de accesos.

El API utiliza tecnología Spring Boot Web, H2 Drives y DevTools se ha mapeado todas las actividades de mantenimiento en el framework Spring, el mapeo quedó de la siguiente manera:

|  |  |  |
| --- | --- | --- |
| Acción | Acceso URL | Método |
| ADDRESSES |  |  |
| Consulta General | http://localhost:8080/api/v1/addresses | GET |
| Consulta por ID | http://localhost:8080/api/v1/address/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/address | POST |
| Actualizar datos | http://localhost:8080/api/v1/address | PUT |
| Borrar registro | http://localhost:8080/api/v1/address/{id} | DELETE |
| APPOINTMENTS |  |  |
| Consulta General | http://localhost:8080/api/v1/appointments | GET |
| Consulta por ID | http://localhost:8080/api/v1/appointment/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/appointment | POST |
| Actualizar datos | http://localhost:8080/api/v1/appointment | PUT |
| Borrar registro | http://localhost:8080/api/v1/appointment/{id} | DELETE |
| CONTACTS |  |  |
| Consulta General | http://localhost:8080/api/v1/contacts | GET |
| Consulta por ID | http://localhost:8080/api/v1/contact/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/contact | POST |
| Actualizar datos | http://localhost:8080/api/v1/contact | PUT |
| Borrar registro | http://localhost:8080/api/v1/contact/{id} | DELETE |
| COUNTRIES |  |  |
| Consulta General | http://localhost:8080/api/v1/countries | GET |
| Consulta por ID | http://localhost:8080/api/v1/country/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/country | POST |
| Actualizar datos | http://localhost:8080/api/v1/country | PUT |
| Borrar registro | http://localhost:8080/api/v1/country/{id} | DELETE |
| EMAILS |  |  |
| Consulta General | http://localhost:8080/api/v1/emails | GET |
| Consulta por ID | http://localhost:8080/api/v1/email/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/email | POST |
| Actualizar datos | http://localhost:8080/api/v1/email | PUT |
| Borrar registro | http://localhost:8080/api/v1/email/{id} | DELETE |
| PHONES |  |  |
| Consulta General | http://localhost:8080/api/v1/phones | GET |
| Consulta por ID | http://localhost:8080/api/v1/phone/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/phone | POST |
| Actualizar datos | http://localhost:8080/api/v1/phone | PUT |
| Borrar registro | http://localhost:8080/api/v1/phone/{id} | DELETE |
| TYPE\_ADDRESSES |  |  |
| Consulta General | http://localhost:8080/api/v1/typesAddresses | GET |
| Consulta por ID | http://localhost:8080/api/v1/typesAddress/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/typesAddress | POST |
| Actualizar datos | http://localhost:8080/api/v1/typesAddress | PUT |
| Borrar registro | http://localhost:8080/api/v1/typesAddress/{id} | DELETE |
| USERS |  |  |
| Consulta General | http://localhost:8080/api/v1/users | GET |
| Consulta por ID | http://localhost:8080/api/v1/user/{id} | GET |
| Agregar nuevo registro | http://localhost:8080/api/v1/user | POST |
| Actualizar datos | http://localhost:8080/api/v1/user | PUT |
| Borrar registro | http://localhost:8080/api/v1/user/{id} | DELETE |

# Pruebas y testing

Para las pruebas y desarrollo se utilizó **Postman** API Client en su versión 8 para realizar las pruebas a la aplicación que ejecuta en Tomcat embebido en STS. El uso general es el siguiente:

Después de instalar la aplicación y validar su cuenta de usuario, vamos a entrar en API Client y según indica la siguiente imagen ejecutar la acción indicada para cada enlace.

Interfaz de usuario gráfica, Texto, Aplicación, Correo electrónico

Descripción generada automáticamente

Si existen inconvenientes en esta ejecución podremos depurar la aplicación.