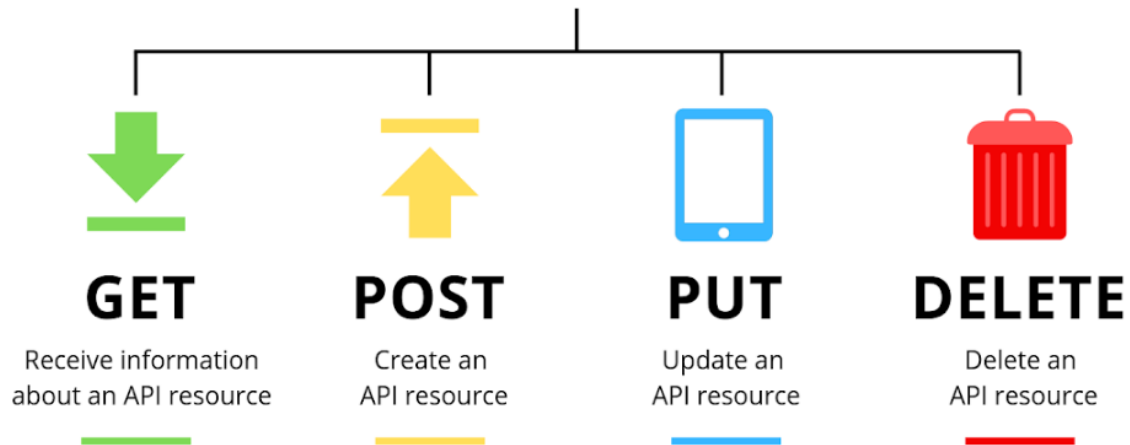


INFORME DE PROYECTO SEMESTRAL REST W.S.

GRUPO Z

REST API Methods



Integrantes:

- Edgar I. Matus S
- Alex P. Bidart Orellana
- Nicolás A. Jiménez J.

Profesor:

- Sebastián Salazar M.

12/07/2021

Introducción	3
Metodología	4
Enfrentamiento del problema:	4
Desarrollo:	4
Resultados:	6
Tecnología	7
Conclusiones	8

Introducción

Este informe contiene las especificaciones del desarrollo del proyecto realizado para la asignatura de Computación Paralela y Distribuida, que consistía en la creación o desarrollo de una *Api-Restful* aplicada a una página web html casi estática de la Agencia Geofísica de la Universidad de Chile con la finalidad de obtener y almacenar en una base de datos la información histórica de los últimos sismos producidos en territorio nacional y poder entregarla a aplicaciones consumidoras.

Contextualizando un poco más con respecto al servicio creado, se realizó una *Api-Restful* mediante el *framework* de desarrollo *Spring-boot* dentro de la cual se realiza la práctica de *scraping* y autenticación mediante JWT para consumirla por la aplicación móvil que se tiene contemplada dentro de la asignatura utilizando también un gestor de motor de base de datos a través de dependencias útiles para cada contexto de desarrollo dentro del proyecto.

Metodología

Enfrentamiento del problema:

La manera de abordar el problema, fue por una parte a un nivel semi profesional de ponerse en el lugar de desarrolladores que por alguna razón tendrían que desarrollar esta interfaz para dar soporte a alguna otra aplicación (el cual era así el caso) a través de un servicio web. Es por esto que se realizó un estudio de los temas relacionados, como por ej. de códigos de estados HTTP para luego así poder analizar y estudiar lo realizado por el docente en clase a través de las grabaciones de estas, como también del repositorio del estimado.

Desarrollo:

Siguiendo un poco con lo anterior, se realizó un repositorio dentro del cual se importó la estructura o esqueleto obtenido de <https://start.spring.io/> necesario para desarrollar una aplicación de *Spring-boot*, ya que por motivos académicos y didácticos, que comprometían aprender este framework como también del lenguaje de programación Java, pero se optaron también producto de las circunstancias contextuales como grupo.

Se trabajó en un repositorio privado Github, que al momento de leer el informe, el docente podrá ver, en donde se siguió una estructura genérica muy parecida a la realizada por este, siguiendo el paradigma del orden en harás de las mejores prácticas para por nuestro desarrollo como muy próximos futuros profesionales.

La forma en que el Proyecto se compone, en resumidas cuentas, es el siguiente: la aplicación debe contar con alguna conexión a alguna base de datos con las tablas necesarias (que son dos) en orden de funcionar con las tablas "`credencial_z`" y "`sismo_z`". Dicho esto, si se desea consumir el servicio principal de este, que es la de obtener un histórico de los sismos que han sido *scrapeados* desde http://www.sismología.cl/links/ultimos_sismos.html, se debe contar con la autorización necesaria para ello como también de un *token* (JWT) que debe solicitar con la restricción de

420 minutos para usarse después de solicitado. No existe una restricción física más que estas como de dirección IP por ej., por motivos académicos, ya que no nos parece necesarias en estas circunstancias pero si se tienen en consideración.

Resultados:

Los resultados de la realización de este proyecto son bastantes satisfactorios, pues se pudo lograr lo mínimo solicitado en el enunciado con resultados locales de ejecución bastante decentes. Sabemos que estos discrepan cuando se utilice una base de datos remota, pero la forma de llevar a cabo el desarrollo evita un excesivo uso del motor aún así teniendo la certeza de que se puede mejorar dichas interacciones. No está demás decir que una de las limitaciones que resultó de la forma de abordar así el proyecto fue la de no mostrar todos los sismo, debido a que estos se extraen de la página al momento de ser solicitados y si existen (porque existirán) sismos durante ninguna solicitud por parte del cliente del servicio para consumirlo durante un tiempo muy prolongado de horas, los sismos que estuvieron dentro de ese intervalo hasta que se hizo un nuevo consumo del servicio no habrán sido guardados en la base de datos.

El código está respaldado en el repositorio público:

<https://github.com/garimatus/proyecto-rest-cpyd>

Y la documentación puede ser accedida mediante el acceso a:

<http://localhost:8080/swagger-ui/>

Tecnología

Como se dijo anteriormente, las justificaciones del uso de la tecnología utilizada fueron principalmente dos: aprender el framework y consecuentemente intentar dominar el lenguaje de programación Java y poder aprovechar el tiempo destinado a la asignatura en aprenderlos para lograr así una mejor preparación para enfrentar el mundo laboral, por el hecho de ser un entorno de desarrollo aún popular en sistemas de informáticos empresariales.

Dentro del grupo de trabajo se tenía relativa experiencia trabajando con servicios parecidos, ya sea por experiencia propia en algún proyecto personal o producto de lo aprendido en la asignatura de Redes y comunicación de Datos, dentro de los cuales ciertos participantes tenían cierta experiencia usando *.NET* y *ExpressJS*, para servicios y aplicaciones web respectivamente.

La ventaja en todo caso de usar *Spring-boot* por sobre los otros dos anteriormente mencionados, radica en que se puede compilar un ejecutable bastante portable entre diferentes arquitecturas de sistemas y poder dockerizarlo con relativa facilidad gracias a las herramientas que nos ofrece como por ej. *Maven*.

Conclusiones

Las conclusiones que podemos obtener de este proyecto son bastante alentadoras, ya que se pudo aprender, en gran medida, a usar el framework utilizado en el proyecto junto con su lenguaje y sistema de repositorios de dependencias bastantes útiles. Además cabe agregar que los resultados (a nivel local) son bastante satisfactorios por el hecho de contar con un ejecutable propio y no estar dependiendo de terceros utilizando un lenguaje de programación de bastante nivel.

Consideramos que la existencia de este tipo de herramientas que pueden ser integradas en proyectos de mayor escala son bastante demandadas en el mundo laboral e interactuar con ellas en trabajos prácticos como este y de este nivel, con los resultados obtenidos, nos parece de mucha utilidad dentro de nuestra formación académica y eso se agradece. Pero lo que nos llama la atención fue la relación entre la asignatura y el proyecto, ya que es algo a lo uno estaría más familiarizado con Redes y si bien hay aspectos de aplicaciones web y dentro del lenguaje Java para sacarle provecho a este tipo de aplicaciones, como los es la asincronía dentro de *Node* por ej., en este proyecto no se utilizó ninguna más los resultados son satisfactorios según la percepción dentro del presente grupo.