





Tutorial de API con SpringBoot-JWT y Postman

> PEV: Javier G. Palacios ARGENTINA PROGRAMA

Prerrequisitos

- Descargar e instalar Sprint Tool 4: https://spring.io/tools
- Descargar e instalar Postman: https://www.postman.com/

Nota: este api se puede usar para el portfolio web, solo hay que agregar las clases entidad, repositorio, servicios, dto, y controladores correspondientes al portfolio.

Toda la parte de seguridad ya esta implementada.

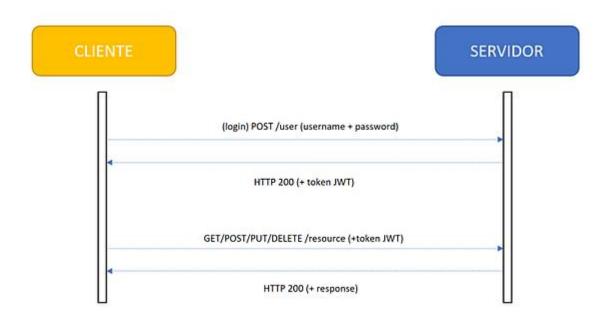
JWT

Es un estándar basado en JSON para crear tokens de acceso que nos permiten agregar seguridad en una comunicación cliente/servidor. El token JWT actúa como un contendor que almacena toda la información que hace referencia a la autentificación de un usuario y/o intercambio de datos del mismo.

¿Qué es el proceso de autenticación?

Es un proceso mediante el cual podemos garantizar la identidad de un cliente. Cuando hablamos de cliente hablamos de quien consume nuestra API Rest.

En forma gráfica, tenemos:



- 1. El cliente se autentica, ejecutando una operación de login. Esta operación puede ejecutarla la misma API que vamos a consumir u otra API o herramienta encargada de llevar a cabo este proceso. El software que realiza el proceso de login se llama **Servidor de Autenticación**.
- 2. Si el login es exitoso y se valida correctamente los datos del usuario, el servidor de autenticación genera un token JWT. Este token contiene, entre otros datos, el nombre del usuario y sus permisos.
- 3. El cliente usa ese token para acceder a los recursos protegidos que se publican mediante una API Rest.

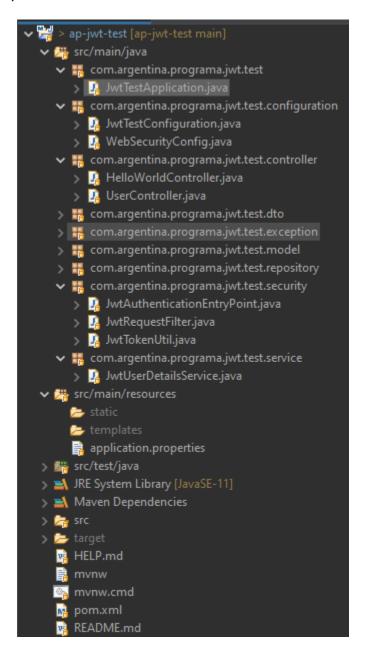
4. En cada petición, el servidor valida el token para comprobar su validez y los permisos del usuario para acceder a un recurso particular.

Hice una API Rest con Spring Boot de ejemplo, que contiene tres endpoints:

- /register: Permite registrar un usuario. Los datos de entrada son nombre de usuario y contraseña.
- /login: Valida las credenciales del usuario y, en caso de ser exitosas, genera el token jwt.
- /hello: Retorna un String con un mensaje. Este endpoint tiene seguridad, lo que significa que para consumirlo hay que enviar un token JWT válido.

Esta API utiliza una base de datos en memoria, por lo que solo tenes que abrirlo con eclipse o netbeans y ejecutar el servicio desde la clase principal JwtTestApplication. Deje algunos comentarios en el código fuente, explicando algunos detalles.

La estructura del proyecto es:

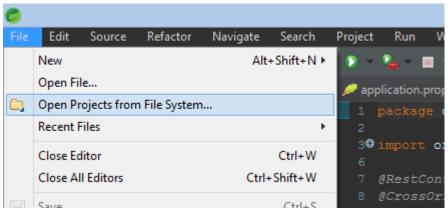


Las clases más importantes son:

- JwtTestApplication: Clase principal, desde esta clase se ejecuta el servicio.
- JwtTokenUtil: Clase que tiene la lógica para crear y validar un Token JWT.
- JwtRequestFilter: Filtro HTTP que valida el Token JWT en cada request.
- WebSecurityConfig: Clase de configuración. En esta clase configuramos que endpoints tienen seguridad y cuales no, entre otras cosas.

Pasos para probar la API

1. Importar la API con Spring Tool Suite 4



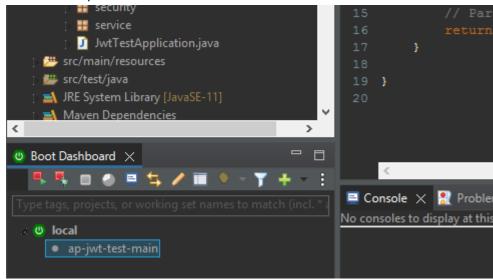
b. Les tiene que quedar así:

```
workspace-sts-v4.16 - ap-jwt-test-main/src/main/java/com/argentina/programa/jwt/test/con
    Edit Source Refactor Navigate Search Project Run Window Help
📷 - 🔡 🖷 : 📑 : 🐚 : 💇 : 💋 🗗 🗐 - 🖫 : 🤻 - 🕟 - 🚰 - 🔳 - 🖺 - : 👪 🐧 - : 😂 🚀 - : 🛂
📱 Package Explorer 🗙 🍱 JUnit - 📙 ≒ 🚦 📅 🗖 - 🗓 HelloWorldController.java 🗙
 ap-jwt-test-main [boot]
                                                 package com.argentina.programa.jwt.test.controller;
   src/main/java
                                               30 import org.springframework.web.bind.annotation.CrossOrio
      # com.argentina.programa.jwt.test
        # configuration
        HelloWorldController.java
          UserController.java
        ₩ dto
        # exception
        ₩ model
        # repository
        # security
        JwtTestApplication.java
   src/main/resources
   👺 src/test/java

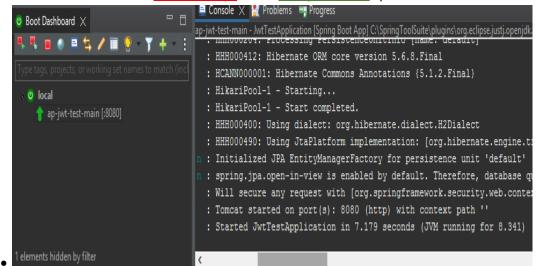
➡ JRE System Library [JavaSE-11]

   Maven Dependencies
```

2. Iniciar la API desde la pestaña Boot Dasboard:



b. Luego hacer clic en el botón del cuadrado rojo con el triángulo verde, quedaría así:



3. Abrir Postman

a.

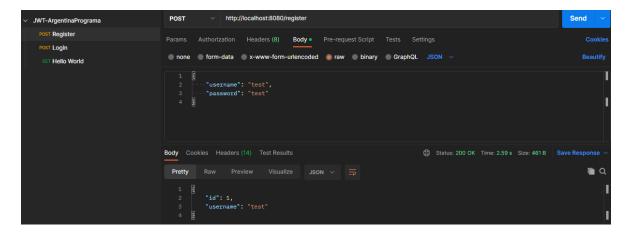
Dentro de la carpeta ap-jwt-test\postman

Se encuentra el archivo **ap_jwt_test.postman_collection.json** que es el que deben importar con Postman para realizar las pruebas de la API.

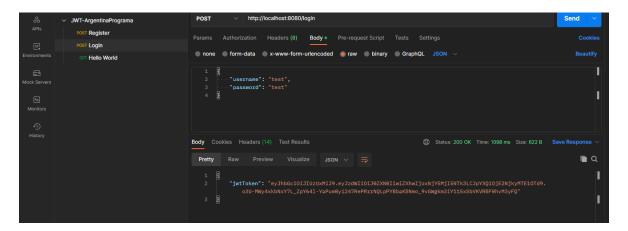
Les tiene que quedar así:



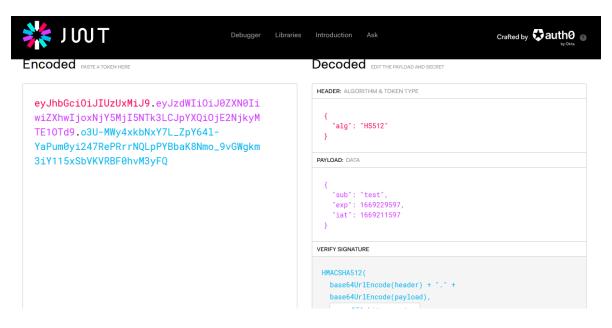
1-Seleccionar el método Register luego hacer Clic en el botón Send, y la API respondera con un 200 Ok.



2- Seleccionar el método Login(usando el username y password registrado en el paso anterior), luego hacer Clic en el botón Send, y la API respondera con un 200 Ok y nos devolverá el jwtToken de autorizacion.

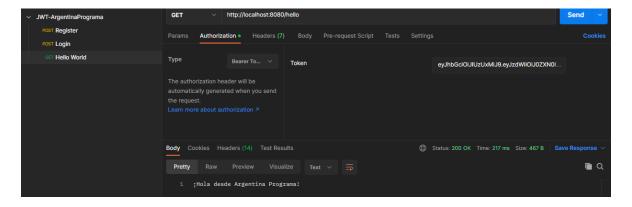


En este punto si queremos ver qué información tiene el token, seleccionamos el código alfanumerico que esta entre comillas dobles y lo pegamos en https://jwt.io/



3-Finalmente seleccionamos y copiamos el token devuelto por la API, y ahora seleccionamos el Metodo Hello World

Vamos a la pestaña en el Type seleccionamos Bearer Token, y en Token pegamos el nuestro.



Ahora cada vez que implementemos nuestros controladores devemos enviar el token de seguridad para que se autorize y nos devuelva o guarde datos en la base de datos.

Nota: código fuente adaptado de https://github.com/leandrounsl/ap-jwt-test