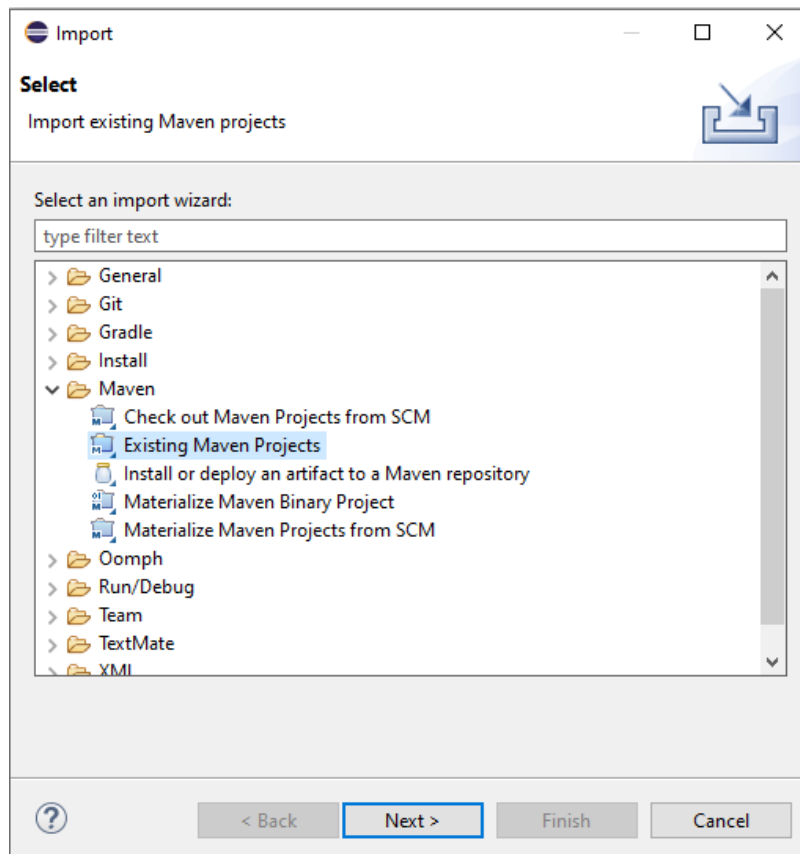
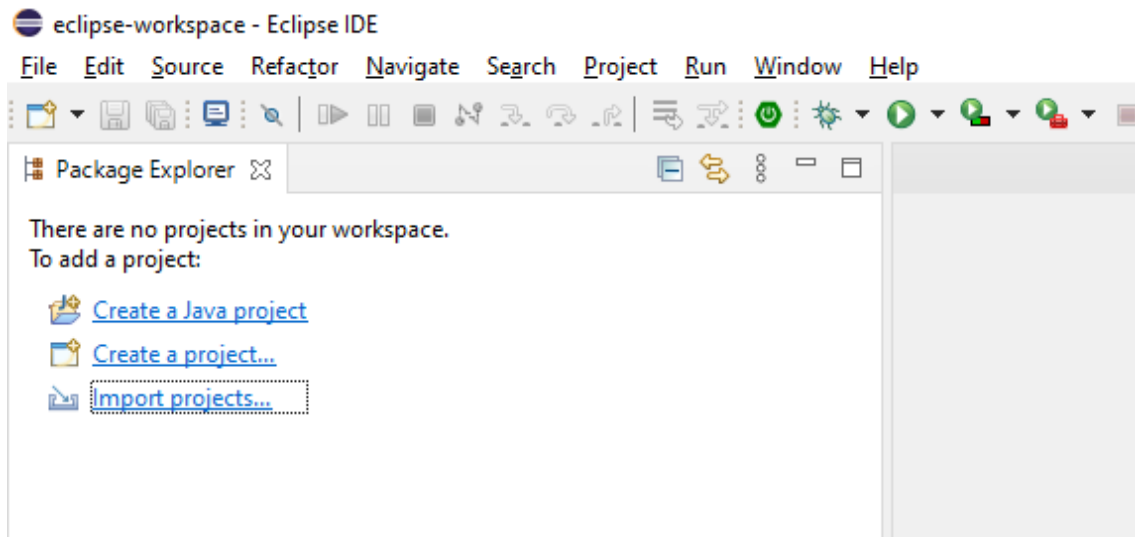


## Utilización del proyecto

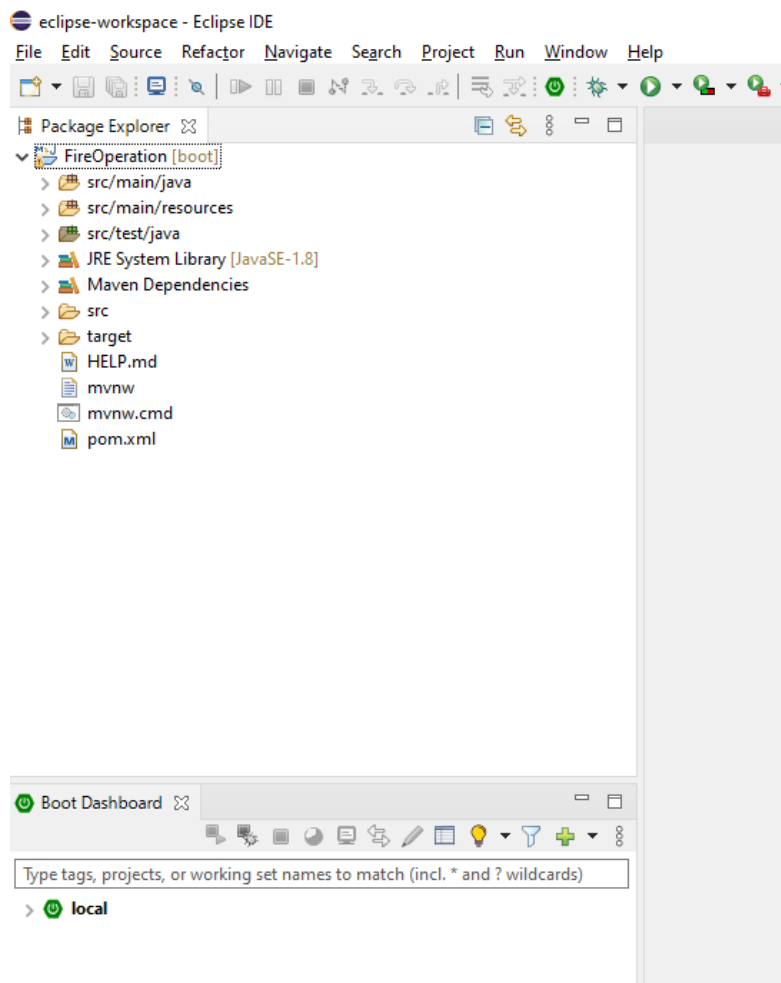
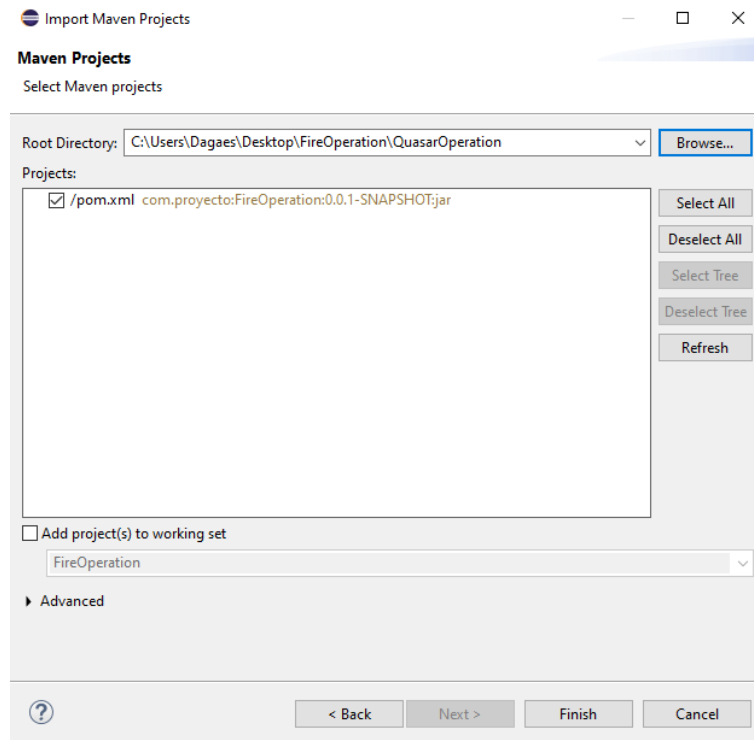
El proyecto fue elaborado en Java versión 1.8, bajo tecnología Maven-2 y springboot versión 2.5.3, soportado en Java JRE 8. Para inicializar la aplicación, bajo el IDE de Eclipse, se deben seguir los siguientes pasos:

\* El IDE utilizado debe tener el plugin correspondiente a Springboot

## Importar el proyecto



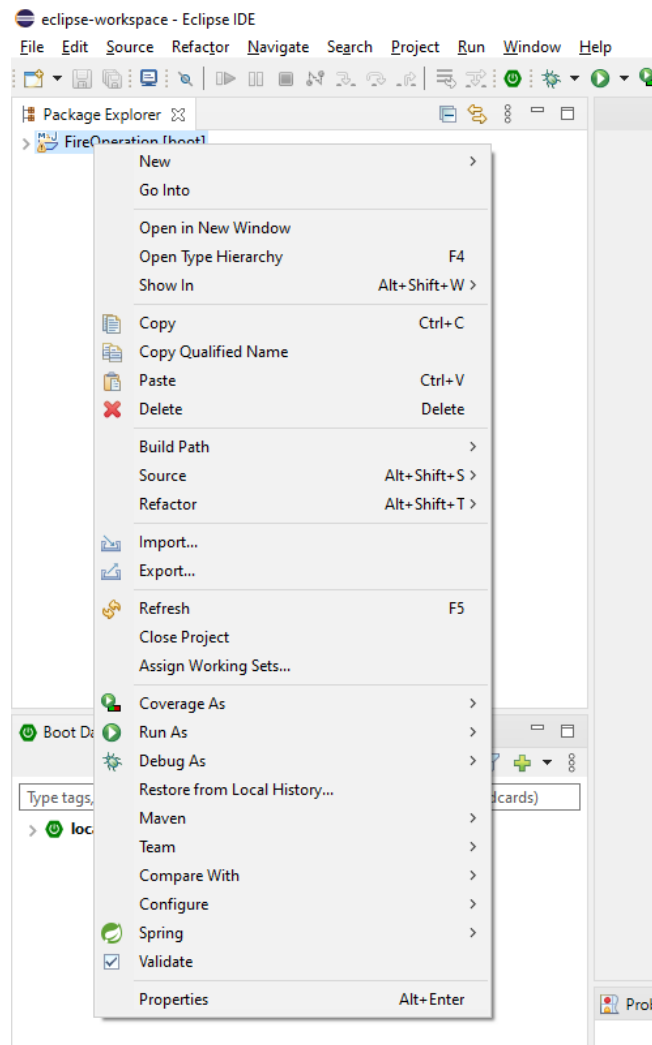
## Mercado Libre - Realizado por David Galeano Espitia



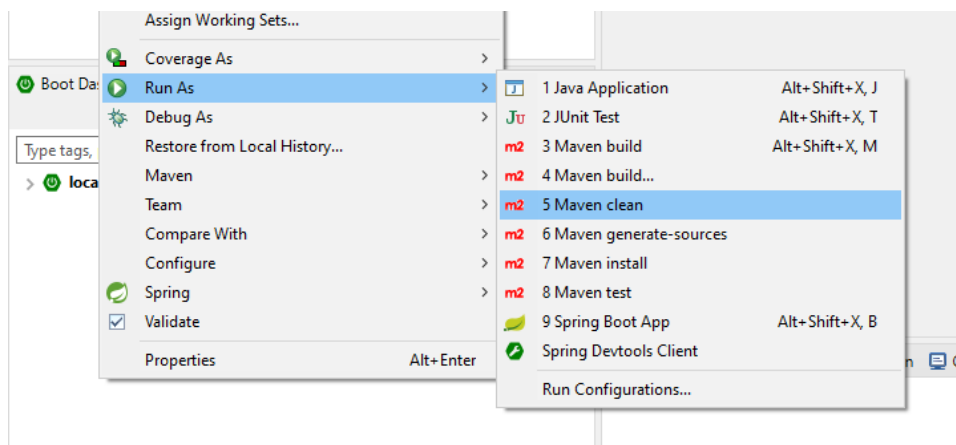
Una vez que finalice la importación, se debe hacer la instalación de las librerías y dependencias del proyecto bajo los siguientes pasos:

## Instalación de dependencias:

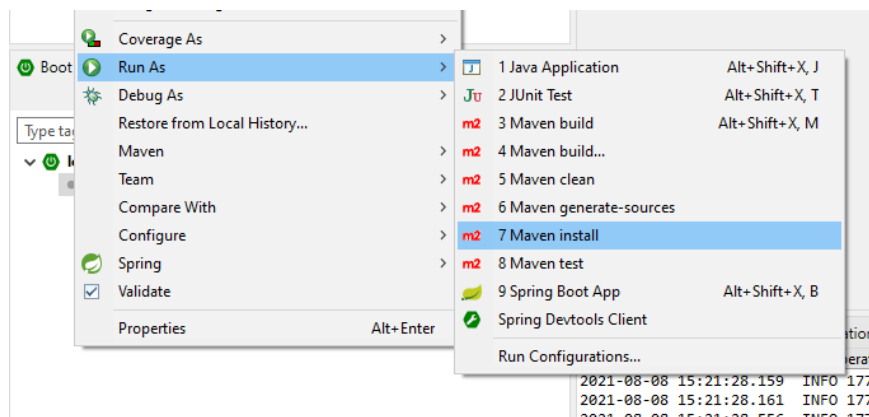
Seleccionar y dar click derecho en la carpeta raíz del proyecto:



En la opción “Run As”, ejecutar la opción “Maven Clean”:

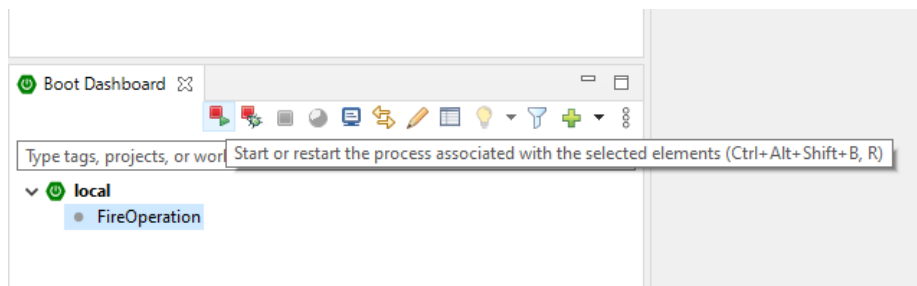


Luego de que finalice, Se deberá hacer lo mismo con la opción “Run As”->”Maven Install”:

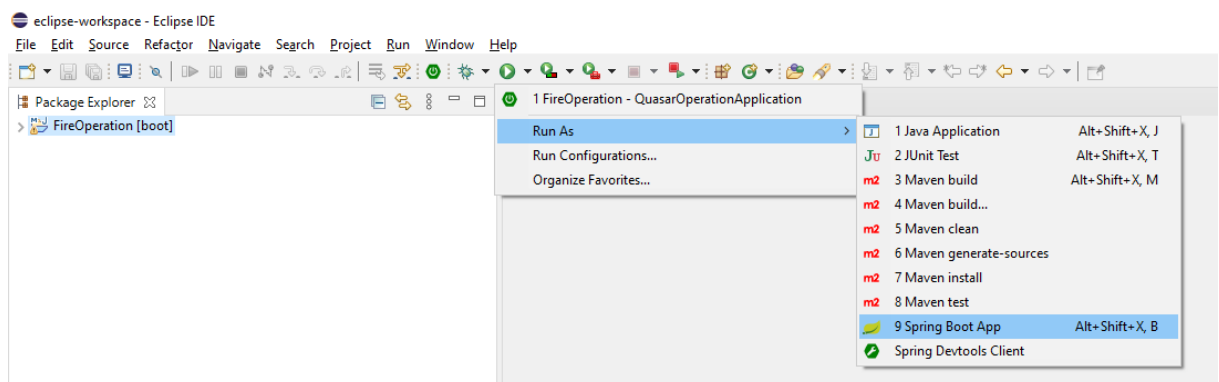


## Ejecución:

Mediante la ventana “Boot dashboard”, seleccionar el proyecto y dar click en el botón “Start - Restart”:



O bien en la barra de tareas, con la opción “Run”->”Run As”->”Spring Boot App”:



## Configuración de Satélites:

Esta aplicación está diseñada para soportar n cantidad de satélites, pero para dar una respuesta, es necesario que haya al menos 3.

Los satélites deben estar configurados en el archivo “**satelite.json**” del proyecto ubicado en:

**{Pathproyecto}\src\main\resources**

La estructura es la siguiente:

```
{
  "satellites": [
    {
      "name": "kenobi",
      "position": {
        "x": -500,
        "y": -200
      }
    },
    {
      "name": "skywalker",
      "position": {
        "x": 100,
        "y": -100
      }
    },
    {
      "name": "sato",
      "position": {
        "x": 500,
        "y": 100
      }
    }
  ]
}
```

Si se incluye un satélite adicional o se cambian los datos de este archivo en tiempo de ejecución, la aplicación tomara los cambios automáticamente.

## Utilización:

Por medio del cliente correspondiente o Postman, se debe usar el siguiente servicio:

1. Método para envío de la información de todos los satélites al tiempo:

**Método:** POST

**URL:** <http://{ip}:8080/topsecret>

\* IP corresponderá a la ip o dirección del servicio.

**Body:** Debe tener la siguiente estructura:

```
{
  "satellites": [
    {
      "name": "name",
      "distance": 0.0,
      "message": []
    }
  ]
}
```

**Respuesta:**

```
{
  "position": {
    "x": 0.0,
    "y": 0.0
  },
  "message": ""
}
```

En caso de éxito, será con un código 200, en caso de error, 404.

2. Método para envío de la información individualmente (no genera respuesta inmediata):

**Método:** POST

**URL:** [http://{ip}:8080/topsecret\\_split/{satellite\\_name}](http://{ip}:8080/topsecret_split/{satellite_name})

\* IP corresponderá a la ip o dirección del servicio.

**Body:** Debe tener la siguiente estructura:

```
{
  "distance": 0.0,
  "message": ""
}
```

**Respuesta:**

En caso de éxito, será con un código 200, en caso de error, 404.

3. Método para obtener la respuesta, basado en la información parcial enviada individualmente por cada satélite:

**Método:** GET

**URL:** [http://{ip}:8080/topsecret\\_split/](http://{ip}:8080/topsecret_split/)

\* IP corresponderá a la ip o dirección del servicio.

**Respuesta:**

```
{  
  "position": {  
    "x": 0.0,  
    "y": 0.0  
  },  
  "message": ""  
}
```

En caso de éxito, será con un código 200, en caso de error, 404 con el mensaje, “Datos insuficientes”.