



Certified Tech Developer

The Ultimate Degree

paci

Back end I

Consigna para las mesas de trabajo

Crear un proyecto Spring Boot

Objetivo

Tal como te pedimos en la clase asincrónica, ahora vamos a crear otro proyecto con Spring Boot.

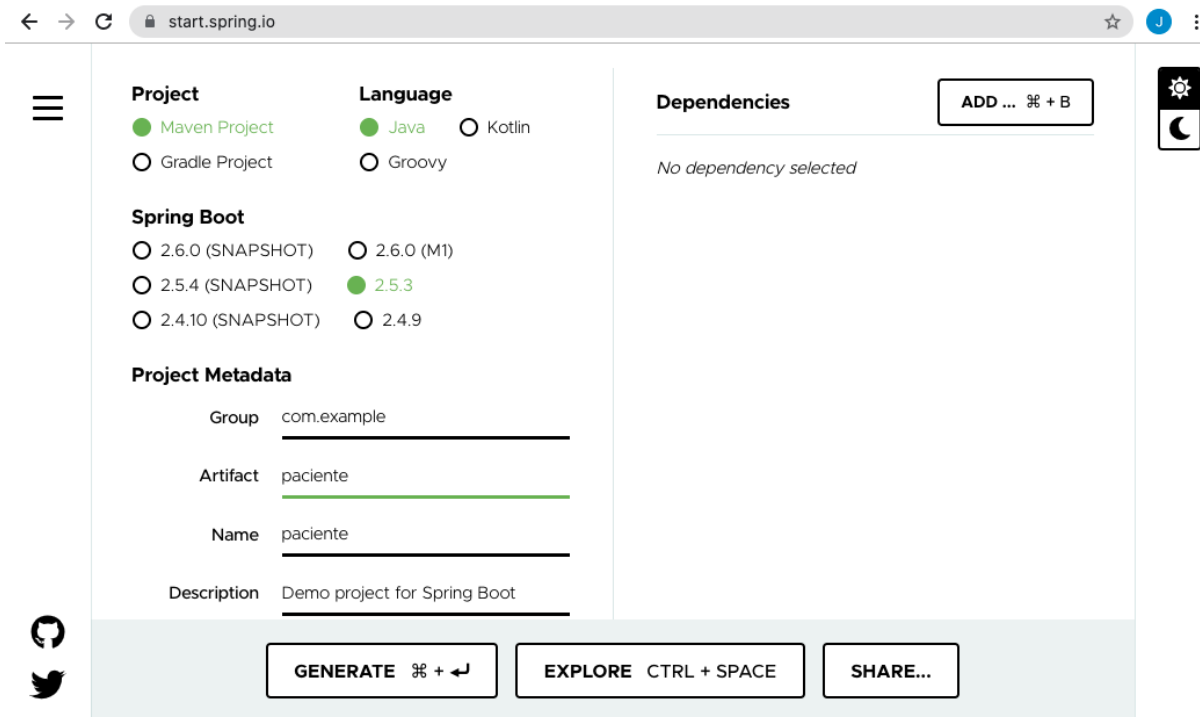
A tener en cuenta:

- Ejercicio individual
- Complejidad: baja 🔥

Instrucciones

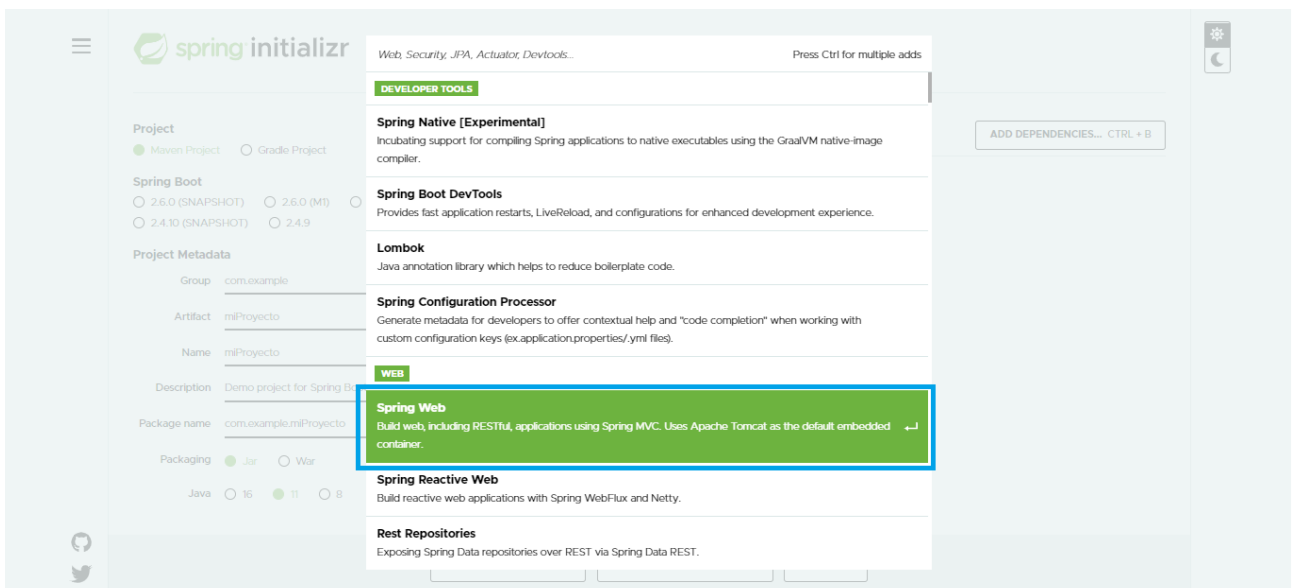
1. Primero, debemos ingresar al siguiente link: <https://start.spring.io>.
2. Allí visualizarás la siguiente pantalla. En el apartado "Name" y "Artifact" escribí "paciente" (atajo: si lo escribís primero en "Artifact", se completa por defecto el "Name").

Los otros campos debés dejarlos tal cual aparecen.



The screenshot shows the start.spring.io web application. It has a sidebar with a hamburger menu and social media icons. The main content area is divided into three sections: Project, Language, and Dependencies. The Project section has radio buttons for Maven Project (selected) and Gradle Project. The Language section has radio buttons for Java (selected) and Kotlin, and Groovy. The Spring Boot section has radio buttons for 2.6.0 (SNAPSHOT), 2.5.4 (SNAPSHOT), 2.4.10 (SNAPSHOT), 2.6.0 (M1), 2.5.3 (selected), and 2.4.9. The Project Metadata section has input fields for Group (com.example), Artifact (paciente), Name (paciente), and Description (Demo project for Spring Boot). The Dependencies section has an 'ADD ...' button and the text 'No dependency selected'. At the bottom, there are three buttons: GENERATE, EXPLORE, and SHARE...

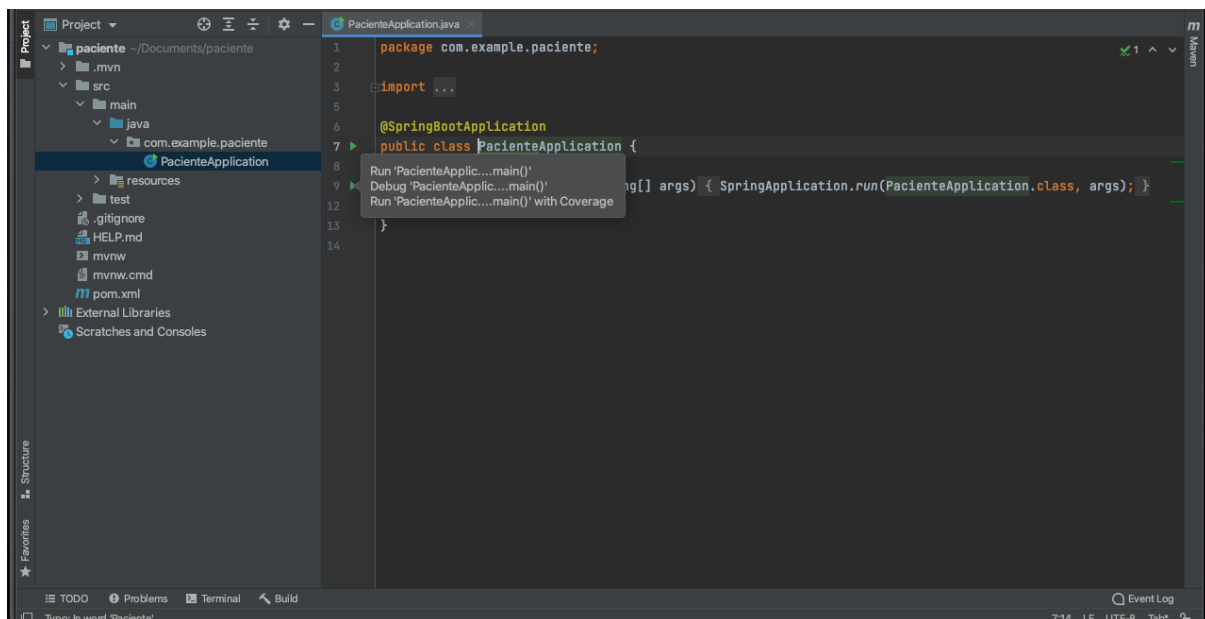
3. Una vez completado lo mencionado anteriormente, presioná el botón “ADD DEPENDENCIES”.
4. Allí se desplegará la siguiente pantalla, donde tenés que elegir la opción “Spring Web”.



The screenshot shows the spring initializr web application. It has a sidebar with a hamburger menu and social media icons. The main content area is divided into three sections: Project, Language, and Dependencies. The Project section has radio buttons for Maven Project (selected) and Gradle Project. The Language section has radio buttons for Java (selected) and Kotlin, and Groovy. The Spring Boot section has radio buttons for 2.6.0 (SNAPSHOT), 2.5.4 (SNAPSHOT), 2.4.10 (SNAPSHOT), 2.6.0 (M1), 2.5.3 (selected), and 2.4.9. The Project Metadata section has input fields for Group (com.example), Artifact (miProyecto), Name (miProyecto), and Description (Demo project for Spring Boot). The Package name is com.example.miProyecto. The Packaging section has radio buttons for Jar (selected) and War. The Dependencies section has an 'ADD DEPENDENCIES...' button. A modal is open showing a list of dependencies. The 'Spring Web' option is highlighted with a blue box. The modal also shows 'Spring Native [Experimental]', 'Spring Boot DevTools', 'Lombok', 'Spring Configuration Processor', 'Spring Reactive Web', and 'Rest Repositories'.



5. Luego, presioná "Generate". Ahora se va a descargar un archivo llamado paciente.zip, el cual deberás descomprimir y abrir IntelliJ IDEA.
6. En IntelliJ IDEA, deberás ir a "Archivo -> Nuevo -> Project from existing source". Luego, tenés que buscar la carpeta "paciente" que descomprimiste y abrirla desde el archivo pom.xml.
7. Buscá la clase PacienteApplication y ejecutá el método main.



8. Ahora, copió y pegá este código para poder crear el Controller:

```
package com.example.paciente;

import org.springframework.boot.SpringApplication;
import org.springframework.boot.autoconfigure.SpringBootApplication;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.bind.annotation.RestController;

@SpringBootApplication
@RestController
public class PacienteApplication {
```



```
public static void main(String[] args) {  
    SpringApplication.run(MiProyectoApplication.class, args);  
}  
  
@GetMapping("paciente")  
public String holaPAciente(){  
    return "Hola Paciente";  
}  
}
```

9. Luego, debés ir al navegador y correr el localhost:8080/paciente.
10. ¿Obtuviste el "Hola Paciente" en pantalla?

Ahora ya tenés un proyecto Spring Boot creado y listo para usarse.