

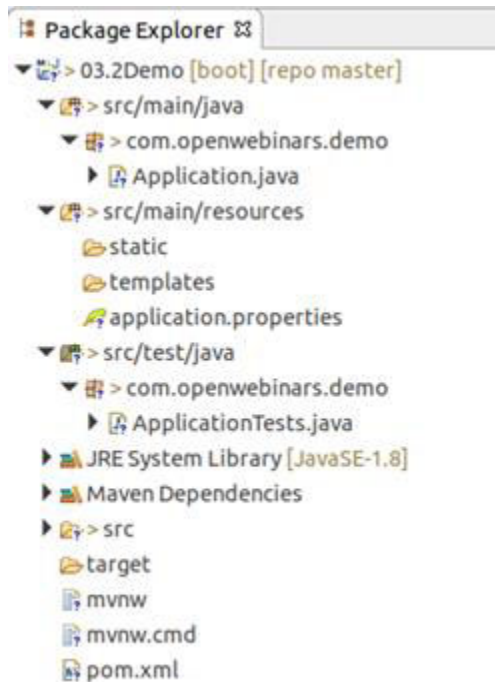
ESTRUCTURA DE UN PROYECTO WEB



APLICACIÓN WEB



PROYECTO DE **SPRING BOOT** GENERADO CON **SPRING INITIALIZR**



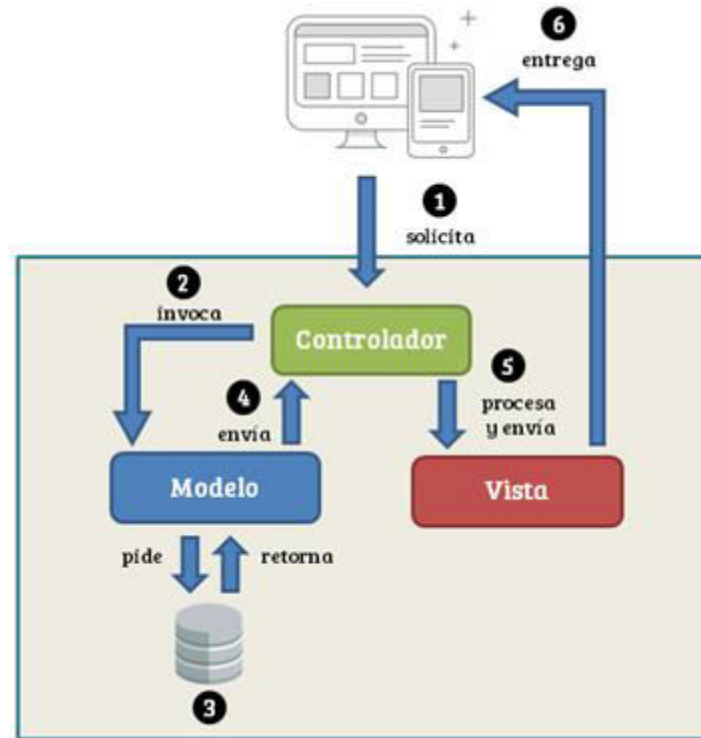
¿Qué elementos va a incluir un proyecto para el desarrollo de una aplicación web?
El patrón MVC nos dará pistas.

ESTRUCTURA SEGÚN MVC

- ▶ Controladores
- ▶ Modelo
 - ▶ Entidades
 - ▶ Repositorios
 - ▶ Servicios
- ▶ Vistas

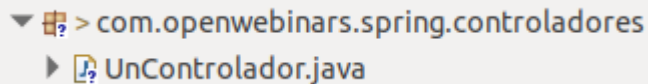
Podemos añadir

- ▶ Configuración
- ▶ Seguridad
- ▶ Utilidades
- ▶ ...



CONTROLADORES

- ▶ Clases con métodos que atenderán las peticiones desde el navegador.
- ▶ Acceden al modelo, y lo retornan a la vista.
- ▶ Al menos un paquete para almacenarlos.

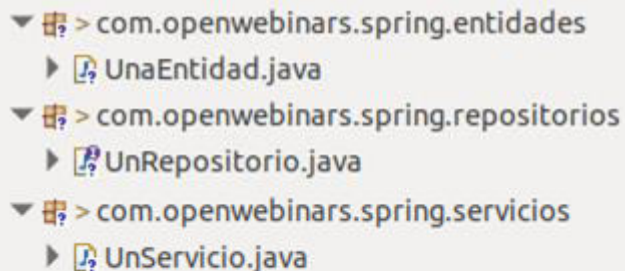


A screenshot of an IDE's package explorer showing a package structure. The package path is `com.openwebinars.spring.controladores`, and it contains a class named `UnControlador.java`. The package icon is a folder with a question mark, and the class icon is a Java file with a question mark.

```
▼ com.openwebinars.spring.controladores
  ▶ UnControlador.java
```

MODELO

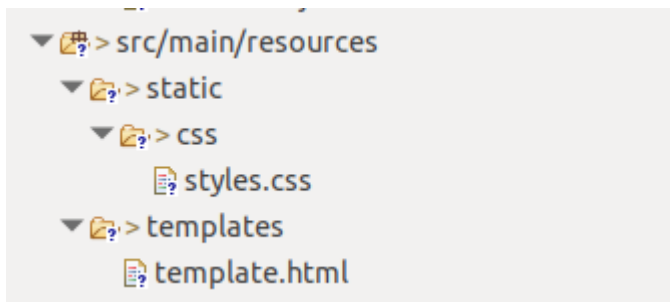
- ▶ Entidades: clases que modelan los objetos de nuestro modelo de negocio.
- ▶ Repositorios (almacenes)
- ▶ Servicios (Lógica de negocio)
- ▶ Un paquete por cada tipo



```
▼ [icon] > com.openwebinars.spring.entidades
  ▶ [icon] UnaEntidad.java
▼ [icon] > com.openwebinars.spring.repositorios
  ▶ [icon] UnRepositorio.java
▼ [icon] > com.openwebinars.spring.servicios
  ▶ [icon] UnServicio.java
```

VISTAS

- ▶ Plantillas (normalmente, HTML) que, tras ser procesadas, serán visualizadas por el navegador.
- ▶ Recursos estáticos (CSS, imágenes, JS, ...)



Plantillas y recursos pueden ser organizados en carpetas, según nos convenga.

¿Y EL RESTO DE PAQUETES?

- ▶ Configuración
- ▶ Seguridad
- ▶ Utilidades

CONFIGURACIÓN

- ▶ Clases anotadas con @Configuration
- ▶ Modifican aspectos de configuración por defecto.
- ▶ Ejemplo: configuración de las propiedades de la base de datos.

SEGURIDAD

- ▶ Útil si usamos Spring Security
- ▶ Clase que extiende a `WebSecurityConfigurerAdapter`
- ▶ Si nuestro manejo de seguridad es complejo, podemos tener subpaquetes (modelo, servicios, configuración, filtros, ...)

UTILIDADES

- ▶ Clases auxiliares que nos sirven para algunas tareas.
- ▶ Pueden ser o no componentes Spring.