**Ejercicio Grupal 4.1 - Módulo 06.**

**Adalid - Awakelab**

**Curso: 0014: Desarrollo de aplicaciones Full Stack Java Trainee**

Integrantes:

* Ricardo Cea
* Aracily Morales
* Mauricio Morales
* Simon Zelada
* Eduardo Torres

A continuación, se adjuntarán imágenes con su respectiva explicación de los procesos de añadir funcionalidad al método crearCapacitacion(), la generación de un log en el sistema al llamar a los métodos y la generación de un archivo .war del proyecto.

En nuestro proyecto, hasta el momento, contamos con 4 controladores con sus respectivas vistas, estos son “Inicio”, “Contacto” “ListarCapacitacion” y “CrearCapacitacion”.

1. Lo primero que se nos solicita es crear un método “crearCapacitacion()” para que al momento que el usuario apriete en su navegador “Crear capacitación”, se envíe un “submit” de la capacitación y este sea mostrado por consola. Como se muestran en las siguientes imágenes:



Imagen01: Imagen de la página web html para crear una capacitación

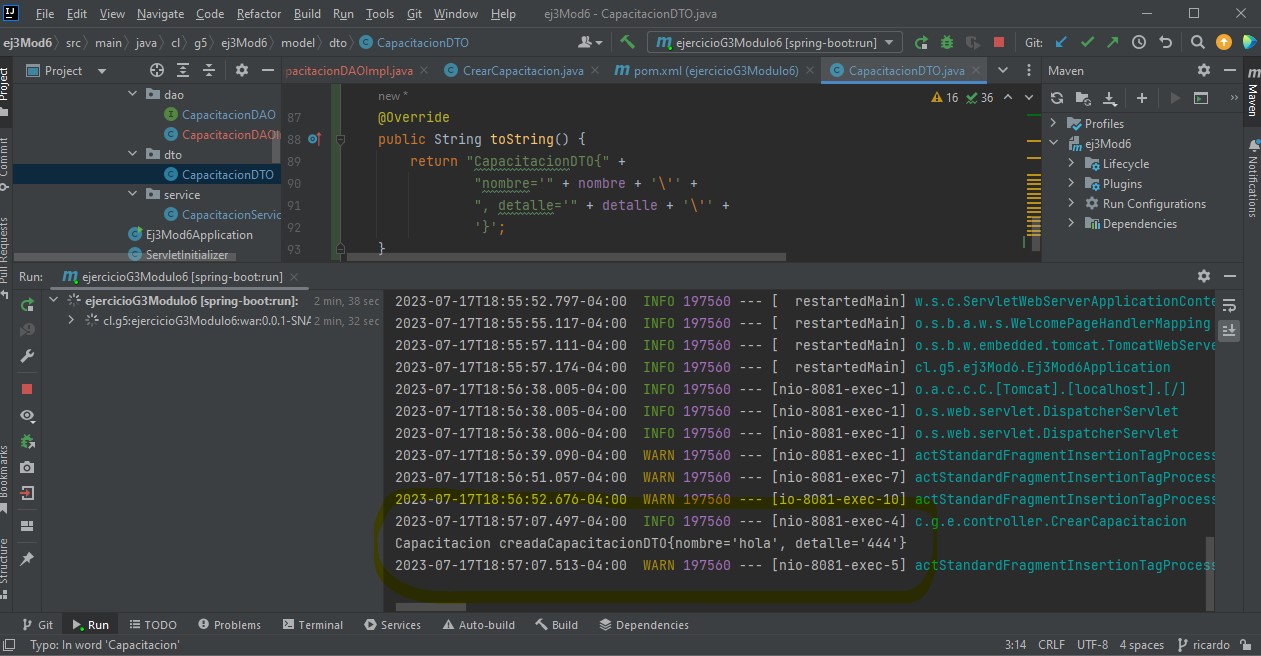


Imagen02: System.out.println generado al presionar “submit” en nuestro sitio web”

1. Posteriormente nos solicitan lo mismo, pero generando un log en el sistema, tal como se muestra a continuación, la consola nos arroja el sgte. mensaje:

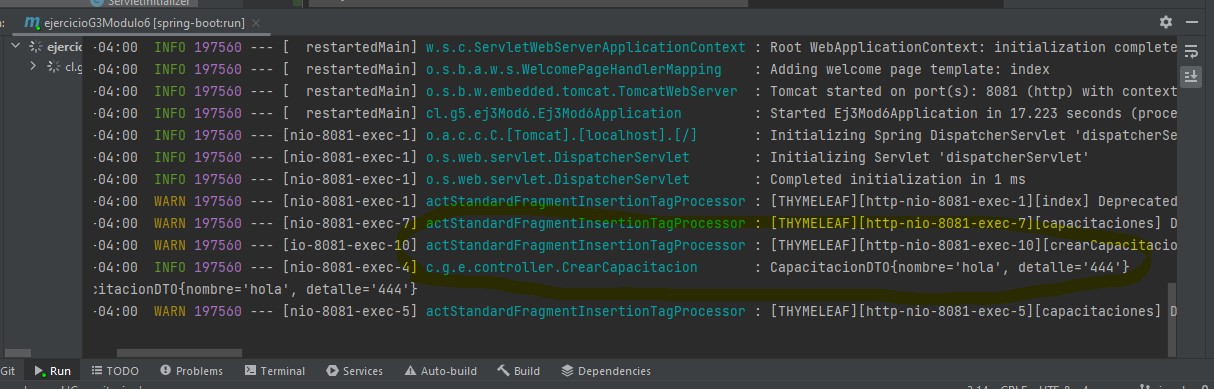


Imagen03: Log del sistema, mostrado por la terminal de IntelliJ

1. Para finalizar, generaremos el archivo . war del proyecto en IntelliJ, mediante Maven “Package”, tal como se muestra en las siguientes imagenes del resultado por consola y el archivo generado en nuestro local:

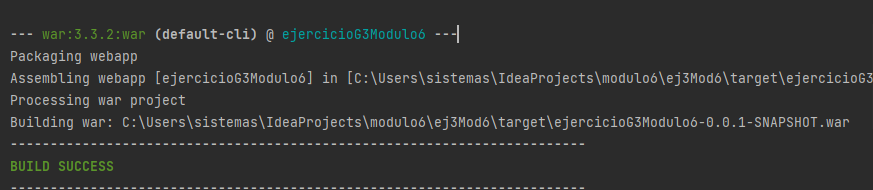


Imagen04: Generación del archivo .war, mediante Maven. se muestra la terminal de IntelliJ

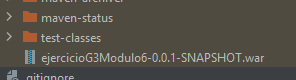


Imagen05: Archivo .war almacenado en nuestro equipo local