**DESARROLLO DE APLICACIONES WEB**

**LABORATORIO N° 10**

**Spring Boot**



|  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| ***Alumno(s):*** | ***Escobedo Carrasco Diego Jaime***  ***Quispe Flores Sandra Paola*** | | | | | | ***Nota*** |  | |
| ***Grupo:*** | ***C24 – A*** | | ***Ciclo: III*** | | | | | | |
| ***Criterio de Evaluación*** | | ***Excelente (4pts)*** | | ***Bueno (3pts)*** | ***Requiere mejora (2pts)*** | ***No acept. (0pts)*** | | | ***Puntaje Logrado*** |
| Instalacion de IDE Spring Tool y creación de proyectos | |  | |  |  |  | | | ***3*** |
| Reconocimiento de estructura de proyecto | |  | |  |  |  | | | ***3*** |
| Desarrolla adecuadamente los ejercicios propuestos | |  | |  |  |  | | | ***6*** |
| Realiza observaciones y conclusiones que aporten un opinión crítica y técnica | |  | |  |  |  | | | ***3*** |
| Es puntual y redacta el informe adecuadamente sin copias de otros autores | |  | |  |  |  | | | ***2*** |
| Evidencia avance en laboratorio | |  | |  |  |  | | | ***3*** |

**Laboratorio 10: Spring Boot**

**Objetivos:**

Al finalizar el laboratorio el estudiante será capaz de:

* Reconoce IDE Spring Tool
* Identifica la estructura de proyecto
* Determina requisitos para instalacion de IDE

**Seguridad:**

* Ubicar maletines y/o mochilas en el gabinete del aula de Laboratorio.
* No ingresar con líquidos, ni comida al aula de Laboratorio.
* Al culminar la sesión de laboratorio apagar correctamente la computadora y la pantalla, y ordenar las sillas utilizadas.

**Equipos y Materiales:**

* Una computadora con:
* Windows 7 o superior
* VMware Workstation 10+ o VMware Player 7+
* Conexión a la red del laboratorio
* Máquinas virtuales:
* Windows 8.1 Pro 64bits Español - Plantilla

**Procedimiento:**

**Lab Setup**

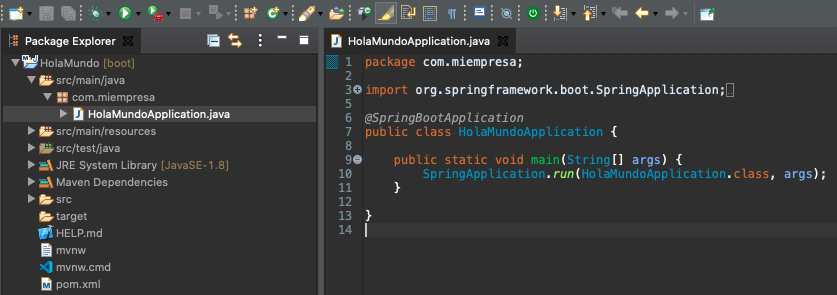
1. **Inicio del equipo virtual**
   1. Encender el equipo
   2. Acceder empleando la cuenta de **usuario**: Tecsup, **contraseña**: \_\_\_\_\_\_\_\_\_\_\_\_\_\_\_
   3. Iniciar el Software VMWare.
   4. Abrir el clon del equipo virtual **C15-DAW** creado la sesión anterior
   5. Iniciar el equipo virtual **C15-DAW**
   6. Identifíquese con la cuenta de usuario: **Redes**. Contraseña: **RCDTecsup2**
2. **Instalacion de IDE Oficial**
   1. Instale el JDK de java (version 8.x)
   2. Verifique que se reconoce el comando **java -version** en una consola
   3. Ingrese al siguiente enlace y descargue **Spring Tool 4 for Eclipse** según el tipo de sistema operativo: <https://spring.io/tools>

**Interfaz de usuario gráfica, Aplicación

Descripción generada automáticamente**

* 1. Instale la IDE descargada

1. **Creacion de Proyecto Spring Boot**
   1. Abra **Spring Tool Suite 4**
   2. Cree un nuevo proyecto:
   3. **File🡪New🡪Spring Started Project.** 
      1. **Name**: HolaMundo
      2. **Type**: Maven
      3. **Packaging**: jar
      4. **Java version**: 8
      5. **Language**: java
      6. **Group**: com.miempresa
      7. **Artifact**: dejar por defecto
      8. **Version**: dejar por defecto
      9. **Description**: Hola Mundo
      10. **Package**: com.miempresa
      11. Todas las demas opciones dejar por defecto
      12. Clic en **Next**
      13. **Spring Boot version**: 2.5.6 (puede variar dependiendo de la version que este utilizando)
      14. En cuadro **Available** escriba “**web”** y active el check de **Spring Web**
      15. Clic en **Next** 2 veces y clic en **Finish**
   4. Expanda el nodo **HolaMundo/src/main/java/com.miempresa/HolaMundoAplication.**java y verifique el codigo generado



* 1. ¿Cuál es la funcion de esta clase creada por el proyecto?

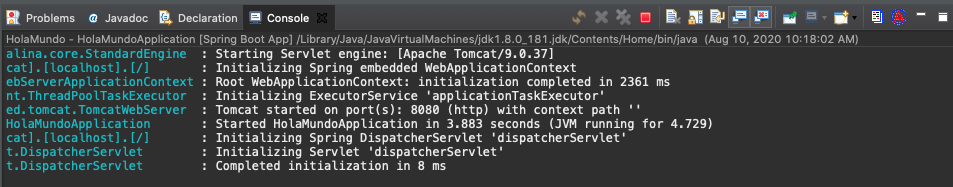
|  |
| --- |
| Esta clase nos esta creando un constructor el cual hace que podamos correr nuestra aplicación como vemos dentro del constructor hay una función run la cual ejecuta la aplicación. |

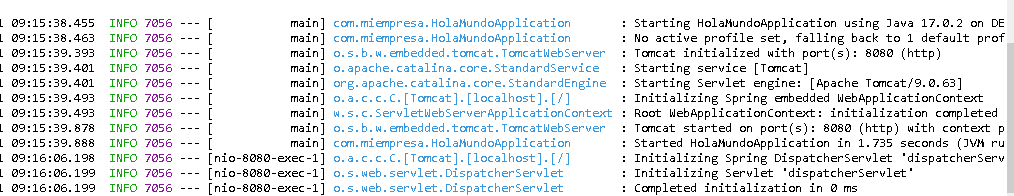
* 1. Haga clic derecho sobre el proyecto creado y elija **Run As🡪 Spring Boot App**
  2. Abra su navegador y digite **localhost:8080** y verifique el resultado mostrado





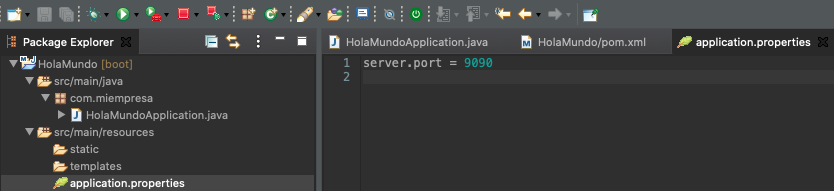
* 1. En su IDE verifique lo mostrado y comente los detalles mas importantes (Que servidor usa, en que puerto esta trabajando)





|  |
| --- |
| Como podemos ver esta utilizando un servidor el cual ya se nos hace conocidos este es TomCat específicamente es Apache TomCat 9.0.63 y esta utilizando el puerto 8080 como ya es clásico, además nos podemos percatar de todas las librerías que se ejecutan en el main. |

* 1. Abra el archivo src/main/resources/**application.properties** y agrege la siguiente linea

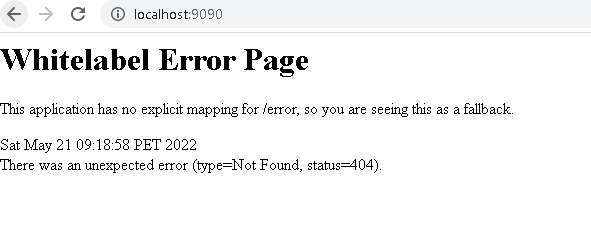


* 1. En la parte inferior izquierda de su IDE(Boot Dashboard) expanda local , clic derecho en su proyecto HolaMundo y y elija Restart

Interfaz de usuario gráfica, Texto, Aplicación

Descripción generada automáticamente

* 1. Verifique en su navegador que que ya no puede acceder por el puerto 8080, intente acceder con el nuevo puerto configurado.



* 1. Comente los detalles mas importantes

|  |
| --- |
| Podemos ver que se esta ejecutando con error porque no hay nada en nuestra clase es decir la clase esta basia nos percatamos que el tipo es no encontrado y el estatus es 404. |

* 1. Investigue sobre que configuraciones mas se pueden realizar desde el archivo **application.properties**

|  |
| --- |
| Categorías de propiedades de Spring Boot  Hay dieciséis categorías de Spring Boot Property que son las siguientes:   * Propiedades principales * Propiedades de caché * Propiedades de correo * Propiedades JSON * Propiedades de datos * Propiedades de transacción * Propiedades de migración de datos * Propiedades de integración * Propiedades web * Propiedades de plantillas * Propiedades del servidor * Propiedades de seguridad * Propiedades de RSocket * Propiedades del actuador * Propiedades de herramientas de desarrollo * Propiedades de prueba |

* 1. Vamos a generar una impresión de nuestros nombres; para esto se debe utilizar controladores. Crearemos un controlador basico solo para entender el proceso de cómo trabaja SpringBoot
     1. En la carpeta **src/main/java** dentro del package **com.miempresa** cree una nueva clase denominada **HolaMundo.java** con el siguiente codigo (Las importaciones puede realizar automaticamente, una vez digite las anotaciones **@RestController** y **@GetMapping**)

Texto

Descripción generada automáticamente

* + 1. Guarde los cambios y en la parte inferior(Boot Dashboard) reincie su proyecto.
    2. En su navegador verifique que ahora se muestra la vista configurada

Imagen que contiene Texto

Descripción generada automáticamente



* + 1. Comente los detalles mas importantes de la implementacion realizada

|  |
| --- |
| Podemos ver como se ejecuta el programa con las exportaciones getmapping y restcontroller podemos ver crea una variable String con nuestros nombres y finalmente la retorna con un poco de código HTML. |

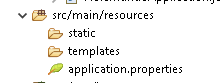
* 1. Investigue y comente para que sirve el archivo y/o directorio. Coloque una captura que muestre lo que se puede visualizar en cada carpeta y/o archivo:
     1. src/main/java

Son recursos estáticos excluidos cuando se ejecuta el programa



* + 1. src/main/resources

En esta carpeta encontramos el static y templates y el archivo application.propierties donde podemos hacer configuraciones a nuestro proyecto.



* + - 1. static

Esta carpeta contiene todos los archivos de estilos: css y scss, archivos de comportamientos JavaScript, imágenes incluidas dentro del theme y el archivo checkout.



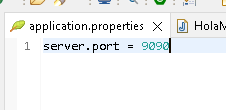
* + - 1. templates

Los templates son los archivos que contienen el código de las páginas de su web site. Se trata, entre otras cosas, de determinar cómo las informaciones van a aparecer en la pantalla (incluso la posición de los placeholders).



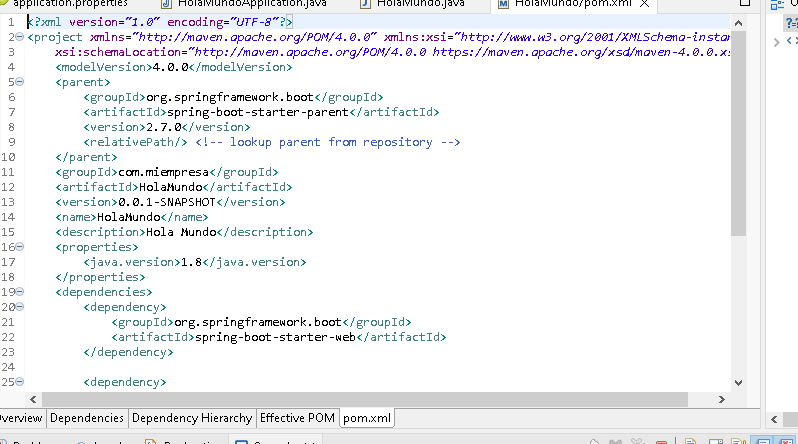
* + 1. application.properties

Las properties son un modo conveniente de proporcionar pares clave-valor que podremos utilizar desde nuestra propia aplicación. De hecho, lo que podemos definir mediante una property es una clave, a la que le asignaremos un valor, y en nuestra aplicación podemos usar esa clave.



* + 1. pom.xml

XML es donde vamos a describir nuestro proyecto. Con packaging se indica el tipo de empaquetado que hay que hacer con el proyecto. Podemos usar jar, war, ear, pom.



**OBSERVACIONES (5 mínimo)**:

* Spring Boot es una tecnología que nos permite crear aplicaciones autocontenidas, con esto nos podemos olvidar de la arquitectura y enfocarnos únicamente en desarrollo.
* Identificamos que el objetivo de Spring Boot es proporcionarnos un conjunto de herramientas para construir rápidamente aplicaciones de Spring que sean fáciles de configurar.
* Al emplear Spring Boot reducimos el tiempo de desarrollo aumentando la productividad.
* Spring Tool Suite es un IDE para desarrollar aplicaciones Spring.
* Es un entorno de desarrollo basado en Eclipse. Proporciona un entorno listo para usar para implementar, ejecutar, implementar y depurar la aplicación.

**CONCLUSIONES (5 mínimo):**

* Al trabajar con Spring Boot podemos delegarle labores como configuración de dependencias, desplegar nuestro servicio o aplicación a un servidor de aplicaciones y enfocarnos únicamente en desarrollar nuestro código.
* Spring Boot nos facilita la creación de aplicaciones independientes basadas en Spring de grado de producción que puede "simplemente ejecutar".
* Spring Boot contiene una infraestructura ligera que elimina la mayor parte del trabajo de configurar las aplicaciones basadas en Spring.
* La versatilidad de SpringBoot hace que podamos abarcar prácticamente cualquier reto en el ámbito profesional de la programación con su ayuda.
* Pensar en Spring Boot es verlo como otro programador que hace todo el trabajo de configurar y administrar las dependencias al detectar de manera inteligente la dirección del proyecto.