

## Proyecto FrontEnd con SPRING BOOT.

El proyecto a desarrollar en el ciclo IV se descomponer en microservicios, cada uno de ellos tiene una responsabilidad y funcionalidad dentro del sistema general. Vamos a hacer un ejercicio en donde separemos el proyecto en front y back para después realizar la conexión entre ellos, en este documento abordaremos el contenido del frontend.

### 1. Acceso de Usuario (FrontEnd).

En este módulo se debe dar acceso al sistema y a la vez tendremos una fase de frontend y otra de backend que debemos implementar por separado o agrupar según lo que deseemos.

En nuestra parte front se debe construir el formulario que captura el ingreso al sistema y el menú de navegación del mismo.

## Requerimientos

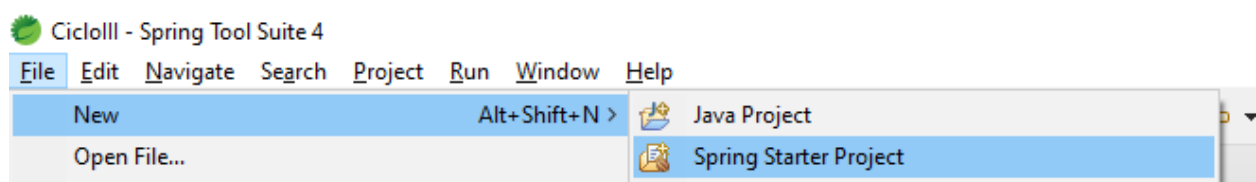
1. Maven 3.0+
2. IDE(STS o Eclipse o Netbeans o Visual Estudio Code)
3. JDK 1.8+

## Contenido

1. Cree el proyecto Spring Boot.
2. Revisar la base de datos MYSQL y defina sus configuraciones en el proyecto.
3. Creación de la capa de presentación.
4. Crear clase de modelo de entidad
5. Crear clase de controladores.
6. Compile y ejecute el proyecto

### 1. Crear el proyecto Spring Boot.

Desde STS 4.0 creamos un nuevo proyecto, seleccionando la opción que se muestra en la figura siguiente:



Ahora definimos los datos deseados o similares a los que se muestran en la siguiente pantalla, vamos a crear un proyecto Maven, jar, con jdk 11 entre las características más importantes.

### New Spring Starter Project

Service URL	<input type="text" value="https://start.spring.io"/>		
Name	<input type="text" value="TiendaFront"/>		
<input checked="" type="checkbox"/> Use default location			
Location	<input type="text" value="C:\ProyectoCicloI\Proyecto\TiendaFront"/>	<input type="button" value="Browse"/>	
Type:	<input type="text" value="Maven"/>	Packaging:	<input type="text" value="Jar"/>
Java Version:	<input type="text" value="11"/>	Language:	<input type="text" value="Java"/>
Group	<input type="text" value="com.mintic.tiendafront"/>		
Artifact	<input type="text" value="TiendaFront"/>		
Version	<input type="text" value="0.0.1-SNAPSHOT"/>		
Description	<input type="text" value="Demo project for Spring Boot"/>		
Package	<input type="text" value="com.mintic.tiendafront"/>		

Tenemos que especificar solo algunas dependencias: Spring Boot Starter Web, Spring Boot DevTools y ThymeLeaf.

### New Spring Starter Project Dependencies

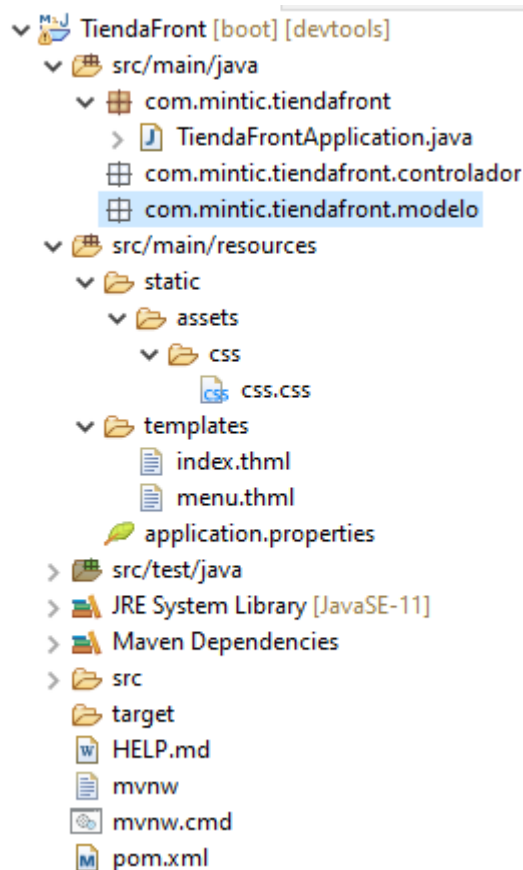
Spring Boot Version:	<input type="text" value="2.5.6"/>
Available:	Selected:
<input type="text"/>	<input checked="" type="checkbox"/> Spring Boot DevTools <input checked="" type="checkbox"/> Thymeleaf <input checked="" type="checkbox"/> Spring Web
<input type="checkbox"/> Developer Tools <input type="checkbox"/> Google Cloud Platform	

## 2. Revisar la base de datos MYSQL y defina sus configuraciones en el proyecto.

Vamos a trabajar con una base de datos similar a la usada en el ciclo anterior, comparto el archivo sql para construirla llamada tiendagenerica.sql, en el front no necesitamos conexión a la base de datos está se realizará por lo pronto en el backend.

## 3. Creación de la capa de presentación.

Crearemos primero el index.html de forma sencilla en donde agregamos Bootstrap con el CDN y a su vez agregamos un css de propio que anexamos dentro de la carpeta static como se aprecia en la siguiente figura:



Contenido de archivo css.css:

```

1 @charset "ISO-8859-1";
2
3 body {
4     text-align: center;
5 }
6
7 table {
8     width: 80%;
9     border: 1px solid black;
10    margin-left: auto;
11    margin-right: auto;
12 }
13
14 td {
15     border: 1px solid black;
16 }
17
18
19 #formulario {
20     text-align: left;
21     margin-left: auto;
22     margin-right: auto;
23 }

```

Contenido de index.html:

```

1 <!DOCTYPE html>
2 <!--
3 Inicio del Proyecto
4 Iniciar Sesión solicitando el nombre del usuario y su clave
5 -->
6 <html xmlns="https://www.thymeleaf.org">
7 <head>
8 <meta charset="ISO-8859-1">
9 <title>login</title>
10 <link
11     href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
12     rel="stylesheet"
13     integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTWfSpd3yD65VohhpuuCOMLASjC"
14     crossorigin="anonymous">
15 <script
16     src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
17     integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
18     crossorigin="anonymous"></script>
19 <link rel="stylesheet" href="assets/css/css.css" />
20 </head>
21 <body>
22 <div class="container">
23
24 <div class="row">
25 <h1>Login Usuario</h1>
26 <form action="#" method="post">
27
28 <div class=" col-xl-4 col-lg-3 col-md-4 col-sm-6" id="formulario">
29 <div class="form-group">
30 <label for="Usuario">Usuario</label> <input type="text"
31     class="form-control" name="nombreUsuario" id="Usuario">
32 </div>

```

```

33
34<div class="form-group">
35  <label for="pwd">Password:</label> <input type="password"
36    class="form-control" name="password" id="pwd">
37  </div>
38  <!-- <div style="color: red">${error}</div>-->
39  <br>
40  <button type="submit" class="btn btn-primary">Ingresar</button>
41  <br><br>
42  </div>
43  </form>
44  </div>
45  </div>
46 </body>
47 </html>

```

Contenido de menu.html:

```

1 <!DOCTYPE html>
2<html>
3<head>
4  <meta charset="ISO-8859-1">
5  <link
6    href="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/css/bootstrap.min.css"
7    rel="stylesheet"
8    integrity="sha384-EVSTQN3/azprG1Anm3QDgpJLIm9Nao0Yz1ztcQTwFspd3yD65VohhpuuCOMLAsjC"
9    crossorigin="anonymous">
10 <script
11   src="https://cdn.jsdelivr.net/npm/bootstrap@5.0.2/dist/js/bootstrap.bundle.min.js"
12   integrity="sha384-MrcW6ZMFYlzcLA8Nl+NtUVF0sA7MsXsP1UyJoMp4YLEuNSfAP+JcXn/tWtIaxVXM"
13   crossorigin="anonymous"></script>
14 <title>Menu Principal</title>
15 </head>
16<body>
17  <nav class="navbar navbar-expand-lg navbar-light bg-light">
18    <a class="navbar-brand" href="/menu">Inicio</a>
19
20    <div class="collapse navbar-collapse" id="navbarText">
21      <ul class="navbar-nav mr-auto">
22        <li class="nav-item active"><a class="nav-link" href="#">Usuarios</a></li>
23        <li class="nav-item"><a class="nav-link" href="#">Clientes</a>
24      </li>
25      <li class="nav-item"><a class="nav-link" href="#">Proveedores</a></li>
26
27      <li class="nav-item"><a class="nav-link" href="#">Ventas</a></li>
28
29      <li class="nav-item"><a class="nav-link" href="#">Reportes</a></li>
30      <li class="nav-item"><a class="nav-link" href="#">Consolidado</a></li>
31    </ul>
32
33    </div>
34  </nav>
35 </body>
36 </html>

```

Contenido inicial de AppControlador:

Sobre el package controlador creamos una clase que permita acceder a las urls de index y menú.html.

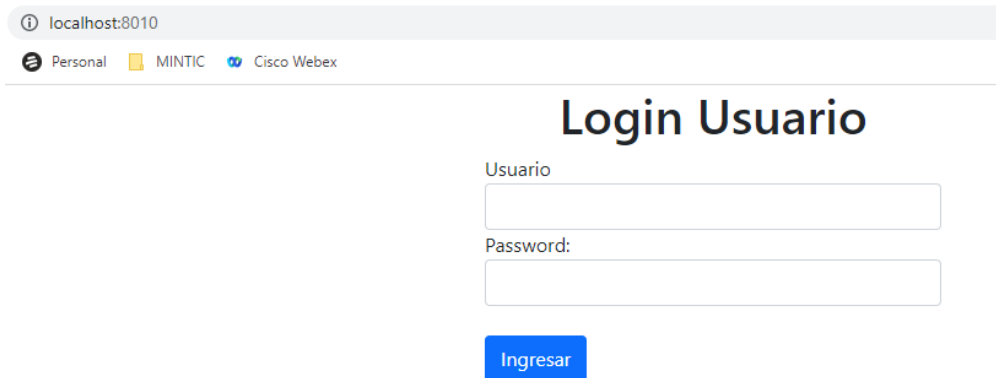
```

1 package com.mintic.tiendafront.controlador;
2
3 import org.springframework.stereotype.Controller;
4 import org.springframework.web.bind.annotation.GetMapping;
5
6 @Controller
7 public class AppControlador {
8
9     @GetMapping("/")
10    public String index() {
11        return "index";
12    }
13
14    @GetMapping("/menu")
15    public String menu() {
16        return "menu";
17    }
18
19 }
20

```

Asignamos un puerto al proyecto 8010 modificando el archivo properties.

Al ejecutar el proyecto debe aparecer una imagen como la siguiente:



Al ejecutar el proyecto en la url /menú se muestra la siguiente figura:



## 4. Crear clase de modelo de entidad

Cree la clase de modelo de dominio del formulario, es decir debe contener solo los atributos que se capturan en el mismo nombreUsuario y password, el nombre de la clase es LoginDto, el contenido de la clase se muestra en la siguiente figura:

```
1 package com.mintic.tiendafront.modelo;
2
3 public class LoginDto {
4
5     private String nombreUsuario;
6
7     private String password;
8
9     public String getNombreUsuario() {
10         return nombreUsuario;
11     }
12
13     public void setNombreUsuario(String nombreUsuario) {
14         this.nombreUsuario = nombreUsuario;
15     }
16
17     public String getPassword() {
18         return password;
19     }
20
21     public void setPassword(String password) {
22         this.password = password;
23     }
24
25 }
```

Esta es una clase de tradicional o simple con el nombre de la clase y los nombres de los campos.