

Solución Desafío - Administración Bodega

Descargar proyecto base desde [Spring Initalizr](#).

Clase Bodega en dto

```
package cl.desafiolatam.challengesecurity1.dto;

import lombok.Getter;
import lombok.Setter;

@Getter
@Setter
public class Bodega {

    private String nombre;

}
```

Clase Bodega en dto

```
package cl.desafiolatam.challengesecurity1.dto;

import lombok.Getter;
import lombok.Setter;

import java.util.Date;

@Getter
@Setter
public class Material {

    private String nombre;
    private Long precio;
    private Date ingreso;
    private Bodega bodega;

}
```

Controlador MaterialController en carpeta controller

```
package cl.desafiolatam.challengesecurity1.controller;

import cl.desafiolatam.challengesecurity1.dto.Material;
import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.*;
import org.springframework.web.servlet.ModelAndView;
import org.springframework.web.servlet.view.RedirectView;

import javax.servlet.http.HttpSession;
import java.util.ArrayList;
import java.util.Collections;
import java.util.List;

@Controller
public class MaterialController {

    @GetMapping("/materiales")
    public ModelAndView home(HttpSession session) {
        ModelAndView modelAndView = new ModelAndView("materiales");
        modelAndView.addObject("material", new Material());
        modelAndView.addObject("materiales",
session.getAttribute("materiales"));
        return modelAndView;
    }

    @PostMapping("/materiales/agregar")
    public RedirectView home(HttpSession session, @ModelAttribute
Material material) {
        List<Material> materials = new ArrayList<>();
        if (session.getAttribute("materiales") != null)
            materials.addAll((List<Material>)
session.getAttribute("materiales"));
        materials.add(material);
        session.setAttribute("materiales", materials);

        return new RedirectView("/materiales");
    }
}
```

Plantilla materiales.html en resources/templates

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:th="http://www.thymeleaf.org">

  <head>
    <title>Detalle Materiales</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.m
in.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0hcWr7x9Jv
oRxT2MZw1T" crossorigin="anonymous">
  </head>

  <body>
    <ul class="nav justify-content-end">
      <li class="nav-item">
        <a class="nav-link active" href="#">Bodega</a>
      </li>
      <li class="nav-item">
        <a class="nav-link" th:href="@{/logout}">Cerrar sesión</a>
      </li>
    </ul>

    <div class="container">

      <div class="col-sm-4">
      </div>
      <div class="col-sm-4">
        <form th:action="@{/materiales/agregar}" method="post"
th:object="${material}">
          <div class="form-group">
            <input type="text" th:field="*{precio}"
class="form-control" placeholder="Ingresa precio">
          </div>
          <div class="form-group">
            <input type="text" th:field="*{nombre}"
class="form-control" placeholder="Ingresa nombre">
          </div>
          <div class="form-group">
```

```
        <input type="text" th:field="*{bodega.nombre}"
class="form-control" placeholder="Ingresa nombre bodega">
    </div>
    <button type="submit" class="btn
btn-primary">Agregar</button>
</form>
</div>
<div class="col-sm-4">
</div>
<div class="col-sm-12">
    <table class="table">
        <thead>
            <tr>
                <th scope="col">precio</th>
                <th scope="col">nombre</th>
                <th scope="col">bodega</th>
                <th scope="col">ingreso</th>
            </tr>
        </thead>
        <tbody>
            <tr th:each="detalle: ${materiales}">
                <td th:text="${detalle.precio}"/>
                <td th:text="${detalle.nombre}"/>
                <td th:text="${detalle.bodega.nombre}"/>
                <td th:text="${#dates.format(new
java.util.Date(detalle.ingreso), 'dd-MM-yyyy')}"/>
            </tr>
        </tbody>
    </table>
</div>

</div>
</body>
</html>
```

Crear configuración de seguridad

```
package cl.desafiolatam.challengesecurity1.configuration;

import org.springframework.context.annotation.Bean;
import org.springframework.context.annotation.Configuration;
import
org.springframework.security.config.annotation.authentication.builders.A
uthenticationManagerBuilder;
import
org.springframework.security.config.annotation.web.builders.HttpSecurity
;
import
org.springframework.security.config.annotation.web.configuration.EnableW
ebSecurity;
import
org.springframework.security.config.annotation.web.configuration.WebSecu
rityConfigurerAdapter;
import
org.springframework.security.crypto.bcrypt.BCryptPasswordEncoder;

@EnableWebSecurity
@Configuration
public class WebSecurityConfig extends WebSecurityConfigurerAdapter
{

    @Override
    public void configure(AuthenticationManagerBuilder auth) throws
Exception {
        auth.inMemoryAuthentication()
            .withUser("jbl@mail.com")
            .password(passwordEncoder().encode("bar"))
            .roles("BODEGA")
        .and()
        .withUser("bose@mail.com")
            .password(passwordEncoder().encode("msc"))
            .roles("BODEGA");
    }

    @Override
    public void configure(HttpSecurity http) throws Exception {
        http.csrf().disable()
    }
}
```

```
        .authorizeRequests()
        .antMatchers("/materiales/**").hasRole("BODEGA")
        .antMatchers("/login").permitAll()
        .anyRequest().authenticated()
        .and()
        .formLogin()
        .loginPage("/login")
        .failureUrl("/login?error=true")
        .usernameParameter("email")
        .passwordParameter("password")
        .defaultSuccessUrl("/materiales");
    }

    @Bean
    public BCryptPasswordEncoder passwordEncoder() {
        return new BCryptPasswordEncoder();
    }
}
```

Controlador Login

```
package cl.desafiolatam.challengesecurity1.controller;

import org.springframework.stereotype.Controller;
import org.springframework.web.bind.annotation.GetMapping;
import org.springframework.web.servlet.ModelAndView;

@Controller
public class LoginController {

    @GetMapping("/login")
    public ModelAndView login(){
        return new ModelAndView("login");
    }
}
```

Plantilla en html para login

```
<!DOCTYPE html>
<html xmlns="http://www.w3.org/1999/xhtml"
      xmlns:th="http://www.thymeleaf.org">

  <head>
    <title>Iniciar Sesión</title>
    <link rel="stylesheet"
href="https://stackpath.bootstrapcdn.com/bootstrap/4.3.1/css/bootstrap.m
in.css"
integrity="sha384-ggOyR0iXCbMQv3Xipma34MD+dH/1fQ784/j6cY/iJTQU0hcWr7x9Jv
oRxT2MZw1T" crossorigin="anonymous">
  </head>
  <body>

    <div class="container">
      <div class="row">
        <div class="col-sm-4">
        </div>
        <div class="col-sm-4">
          <form th:action="@{/login}" method="post"
class="form-signin">
            <h3 class="form-signin-heading"
th:text="Credenciales"></h3>
            <br/>
            <input type="text" id="email" name="email"
th:placeholder="Email"
class="form-control"/> <br/>
            <input type="password" th:placeholder="Password"
id="password" name="password"
class="form-control"/> <br/>

            <div align="center" th:if="${param.error}">
              <p style="font-size: 20; color:
#FF1C19;">Verifica las credenciales.</p>
            </div>
            <button class="btn btn-lg btn-primary btn-block"
name="Submit" value="Login" type="Submit"
th:text="Login">
            </button>
```



```
        </form>

    </div>
    <div class="col-sm-4">
    </div>
</div>
</div>
</body>
</html>
```