

CAPTURAS DEL BACKEND

Regístrate a PeliReview

Únete a la comunidad más crítica y apasionada del cine.

Registrarme

<input type="checkbox"/>	Editar	Copiar	Borrar	3	Admin	NULL	Admin	NULL	NULL	NULL	daniel2@daniel2.com	\$2a5105vrsptEmLQDZ3RP32ONaElZEDIT8kZzQvJU4uZpYr...	1
<input type="checkbox"/>	Editar	Copiar	Borrar	4	Daniel		Murillo		2000-09-28	21	danielmurillomoraes01@gmail.com	123456	2

id	Primer Nombre	Primer Apellido	Nacimiento	Edad	Email	Eliminar
1	Daniel	Murillo	2022-09-10 00:00:00.0	21	daniel@daniel.com	Eliminar
2	Yesid	Murillo	2022-09-10 00:00:00.0	13	daniel1@daniel1.com	Eliminar
3	Admin	Admin			daniel2@daniel2.com	Eliminar
4	Daniel	Murillo	2022-09-09 00:00:00.0	21	dasdas@asdasd.com	Eliminar

id	primer_nombre	segundo_nombre	primer_apellido	segundo_apellido	fecha_nacimiento	edad	email	contrasena	rol_id
1	Daniel	NULL	Murillo	NULL	2022-09-10	21	daniel@daniel.com	123456789	2
2	Yesid	NULL	Murillo	NULL	2022-09-10	13	daniel1@daniel1.com	123456789	2
3	Admin	NULL	Admin	NULL	NULL	NULL	daniel2@daniel2.com	123456789	1
4	Daniel		Murillo		2022-09-09	21	dasdas@asdasd.com	12345678	2
	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL	NULL

id	Primer Nombre	Primer Apellido	Nacimiento	Edad	Email	Eliminar
1	Daniel	Murillo	2022-09-10 00:00:00.0	21	daniel@daniel.com	Eliminar
2	Yesid	Murillo	2022-09-10 00:00:00.0	13	daniel1@daniel1.com	Eliminar
3	Admin	Admin			daniel2@daniel2.com	Eliminar

[illegible]

dymmdaniel +1

@GetMapping("/")

```
public String listarUsuarios(Model model){  
    var usuarios : List<Usuario> = usuarioService.listarUsuarios();  
    model.addAttribute( attributeName: "usuarios", usuarios);  
    return "admin/Usuarios";  
}
```

dymmdaniel +1

@GetMapping("/registrar")

```
public String register(Usuario usuario){  
  
    return "auth/createUser";  
}
```

dymmdaniel +1

@PostMapping("/crear")

```
public String crearUsuario(@Valid Usuario usuario, Errors errores){  
    if(errores.hasErrors()){  
        return "auth/createUser";  
    }else{  
        Rol rol=rolService.buscar( id: 2);  
        usuario.setRol(rol);  
        usuarioService.guardar(usuario);  
        return "redirect:/";  
    }  
}
```

dymmdaniel +1

@GetMapping("/agregar")

```
public String agregar(Usuario usuario) { return "auth/createUser"; }
```

dymmdaniel

@GetMapping("/eliminar")



```
public String eliminar(Usuario usuario){  
    usuarioService.eliminar(usuario);  
    return "redirect:/api/usuario/";  
}
```

repl:generic

Mision Tic Smart Academy Plataformas Stream... Udemey React WhatsApp Cine Calidad GitHub

id	Nombre	Editar	Eliminar
1	Terror	Editar	Eliminar
2	Romance	Editar	Eliminar
3	Drama	Editar	Eliminar
4	Accion	Editar	Eliminar

<

Result Grid   Filter Rows:

	id	nombre
▶	1	Terror
	2	Romance
	3	Drama
	4	Accion
✱	NULL	NULL

```

1 usage
@Autowired
private GeneroService generoService;
@dymmdaniel
@GetMapping("/")
public String listarGeneros(Model model){
    var generos : List<Genero> = generoService.listarGeneros();
    model.addAttribute( attributeName: "generos", generos);
    return "admin/generos";
}

```

id	Título	duracion	Año	Puntuacion	Clasificacion	Editar	Eliminar
1	Spider Man	130 Min	2005	4.5	Jovenes	<button>Editar</button>	<button>Eliminar</button>
2	Spider Man-2	160 Min	2009	3.0	Jovenes	<button>Editar</button>	<button>Eliminar</button>
3	RBD	170 Min	2021	2.0	Adultos	<button>Editar</button>	<button>Eliminar</button>
4	Buscando a nemo	116 Min	2017	4.5	Niños	<button>Editar</button>	<button>Eliminar</button>

[illegible]

dymmdaniel

```
@GetMapping("/")
public String listarPeliculas(Model model){
    var peliculas :List<Pelicula> =peliculaService.listarPeliculas();
    model.addAttribute( attributeName: "peliculas",peliculas);
    return "admin/Peliculas";
}
```

dymmdaniel +1

```
@GetMapping("/{id}")
public String buscarPelicula(Model model,@PathVariable Long id){
    var pelicula :Pelicula =peliculaService.buscar(id);
    if(pelicula==null){
        log.info("No existe la pelicula");
        model.addAttribute( attributeName: "titulo", attributeValue: "No existe");
    }else {
        model.addAttribute( attributeName: "titulo",pelicula.getTitulo());
        model.addAttribute( attributeName: "pelicula", pelicula);
    }
    return "movie/pelicula";
}
```

dymmdaniel

```
@GetMapping("/titulo/{titulo}")
public String buscarPeliculaTitulo(@PathVariable String titulo,Model model){
    var peliculas :List<Pelicula> =peliculaService.findByTitulo(titulo);
    if(peliculas.isEmpty()){
        // Colocar error
    }else {
        model.addAttribute( attributeName: "peliculas", peliculas);
    }
    return "movie/index";
}
```

dymmdaniel

```
@GetMapping("/clasificacion/{clasificacion}")
public String buscarPeliculaClasificacion(@PathVariable String clasificacion,Model model){
    var peliculas :List<Pelicula> =peliculaService.findByClasificacion(clasificacion);
    if(peliculas.isEmpty()){
        // Colocar error
    }else {
        model.addAttribute( attributeName: "peliculas", peliculas);
    }
}
```

London 11 Smart Academy Platanillas Sirena... Odemy React Whatsapp Cine Calicut Github

id	Nombre	Edad minima	Editar
1	Niños	0	<button>Editar</button>
2	Jovenes	13	<button>Editar</button>
3	Adultos	18	<button>Editar</button>

Result Grid			
		Filter Rows:	
Edit:			
	id	nombre	edad_minima
▶	1	Niños	0
	2	Jovenes	13
	3	Adultos	18
✱	NULL	NULL	NULL

```

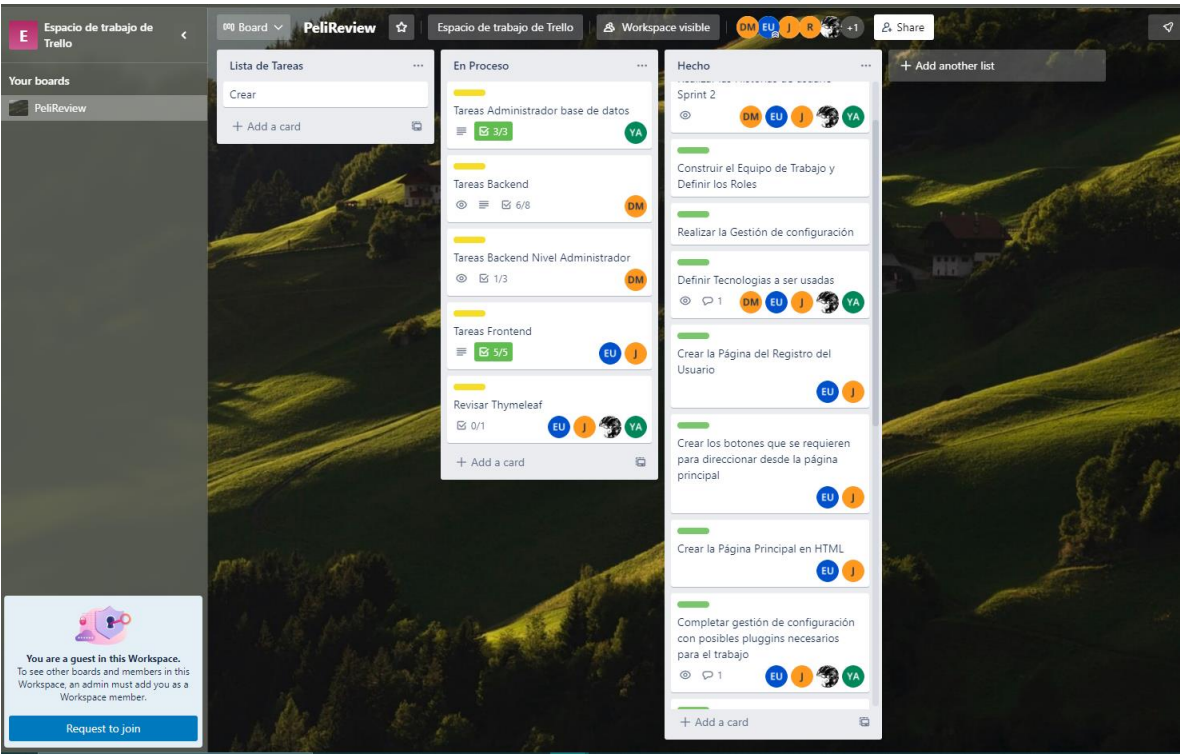
12  @Controller
13  @Slf4j
14  @CrossOrigin
15  @RequestMapping("/api/clasificacion")
16  public class ControladorClasificacion {
17
18      1 usage
19      @Autowired
20      private ClasificacionService clasificacionService;
21
22      @GetMapping("/")
23      public String listarClasificacion(Model model){
24          var clasificaciones : List<Clasificacion> = clasificacionService.listarClasificaciones();
25          model.addAttribute( attributeName: "clasificaciones", clasificaciones);
26          return "admin/Clasificaciones";
27      }
28

```

INFORME RESTROSPECTIVA

Para este tercer sprint teníamos como objetivo la elaboración del CRUD para el proyecto. Respecto a esto, la función de registro de usuario a la página ya se encuentra implementada luego de completar con éxito el primer registro. Para este último sprint nos proponemos terminar la parte del CRUD que resta y comenzar a hacer las pruebas respectivas para la evaluación del proyecto, conectándonos en reuniones más seguidas entre todos.

PANTALLASOS DEL TRELLO



TAREAS PENDIENTES PARA EL SIGUIENTE SPRINT

En Proceso

...

Tareas Administrador base de datos



✓ 3/3

YA

Realizar el Despliegue de la aplicacion



DM

EU

J



YA

Juntar FrontEnd y BackEnd



DM

EU

J



YA

Tareas Backend



✓ 6/8

DM

Tareas Backend Nivel Administrador



✓ 1/3

DM

Tareas Frontend



✓ 5/5

EU

J

Revisar Thymeleaf



0/1

EU

J



YA

+ Add a card



Crear Metodos (CRUD)

Implementar Seguridad

Crear inicio de sesion

Busqueda de pelicula por varios items

33%

Listar Usuarios (Editar y eliminar)

Listar Generos (Editar y eliminar)

Listar Categorias (Editar y eliminar)

Add an item