

Link del repositorio usado: <https://github.com/Ivan-Ruiz00/Consultorio-Ramiro-Fdz/tree/master>

☆ com.sofka.sofkaU.qa.consultorio:consultorio Passed						Last analysis: 3 days ago	
🐛 Bugs	🔒 Vulnerabilities	🛡️ Hotspots Reviewed	🕒 Code Smells	Coverage	Duplications	Lines	
4 C	0 A	– A	34 A	0.0% 	0.0% 	660 XS	Java, XML

Bugs encontrados

src/.../consultorio/controller/HistoriasController.java	
<input type="checkbox"/> Call "Optional#isPresent()" before accessing the value. 🐛 Bug Major Open Not assigned 10min effort Comment	3 days ago L26 🔗 🔍 🔽 🐞 cwe ▼
<input type="checkbox"/> Call "Optional#isPresent()" before accessing the value. 🐛 Bug Major Open Not assigned 10min effort Comment	3 days ago L33 🔗 🔍 🔽 🐞 cwe ▼
<input type="checkbox"/> Call "Optional#isPresent()" before accessing the value. 🐛 Bug Major Open Not assigned 10min effort Comment	3 days ago L40 🔗 🔍 🔽 🐞 cwe ▼
src/.../consultorio/controller/PacienteExistenteController.java	
<input type="checkbox"/> Call "Optional#isPresent()" before accessing the value. 🐛 Bug Major Open Not assigned 10min effort Comment	3 days ago L22 🔗 🔍 🔽 🐞 cwe ▼
4 of 4 shown	

<input type="checkbox"/> Bulk Change	<div><div></div><div></div><div></div></div> 1 / 34 issues 4h 55min effort
src/.../qa/consultorio/consultorio/ConsultorioApplication.java	
<input type="checkbox"/> Rename this package name to match the regular expression <code>^[a-z_](\.[a-z_][a-z0-9_]*)*\$</code> . 🕒 Code Smell Minor Open Not assigned 10min effort Comment	3 days ago L1 🔗 🔍 🔽 🐞 convention ▼
src/.../consultorio/controller/CitaController.java	
<input type="checkbox"/> Rename this package name to match the regular expression <code>^[a-z_](\.[a-z_][a-z0-9_]*)*\$</code> . 🕒 Code Smell Minor Open Not assigned 10min effort Comment	3 days ago L1 🔗 🔍 🔽 🐞 convention ▼
src/.../consultorio/controller/HistoriasController.java	
<input type="checkbox"/> Rename this package name to match the regular expression <code>^[a-z_](\.[a-z_][a-z0-9_]*)*\$</code> . 🕒 Code Smell Minor Open Not assigned 10min effort Comment	3 days ago L1 🔗 🔍 🔽 🐞 convention ▼
src/.../consultorio/controller/PacienteController.java	
<input type="checkbox"/> Rename this package name to match the regular expression <code>^[a-z_](\.[a-z_][a-z0-9_]*)*\$</code> . 🕒 Code Smell Minor Open Not assigned 10min effort Comment	3 days ago L1 🔗 🔍 🔽 🐞 convention ▼
src/.../consultorio/controller/PacienteExistenteController.java	
<input type="checkbox"/> Rename this package name to match the regular expression <code>^[a-z_](\.[a-z_][a-z0-9_]*)*\$</code> . 🕒 Code Smell Minor Open Not assigned 10min effort Comment	3 days ago L1 🔗 🔍 🔽 🐞 convention ▼

Vemos que la mayoría de “*Code Smells*” se debe a nombres o declaraciones de variable en una misma línea

4 Bugs

```
    @PostMapping("/actualizar/{cedula}")
    public ResponseEntity agregar(@PathVariable String cedula, @RequestBody HistoriaMedica historiaMedica) {
        historiaMedicaIService.agregar(pacienteRepository.getPacientes().stream()
            .filter(h -> h.getCedula().equals(cedula))
            .findFirst().get(), historiaMedica);
    }
```

🚩 Call "Optional#isPresent()" before accessing the value.

Vemos que los cuatro Bugs siguen esta estructura haciendo un valor booleano antes de acceder al valor

```
29     @DeleteMapping("/eliminar/{cedula}")
30     public ResponseEntity eliminar(@PathVariable String cedula) {
31         historiaMedicaIService.eliminar(pacienteRepository.getPacientes().stream()
32             .filter(h -> h.getCedula().equals(cedula))
33             .findFirst().get());
    }
```

🚩 Call "Optional#isPresent()" before accessing the value.

```
37     public ResponseEntity actualizar(@PathVariable String cedula, @RequestBody HistoriaMedica historiaMedica) {
38         historiaMedicaIService.actualizar(pacienteRepository.getPacientes().stream()
39             .filter(h -> h.getCedula().equalsIgnoreCase(cedula))
40             .findFirst().get(), historiaMedica);
    }
```

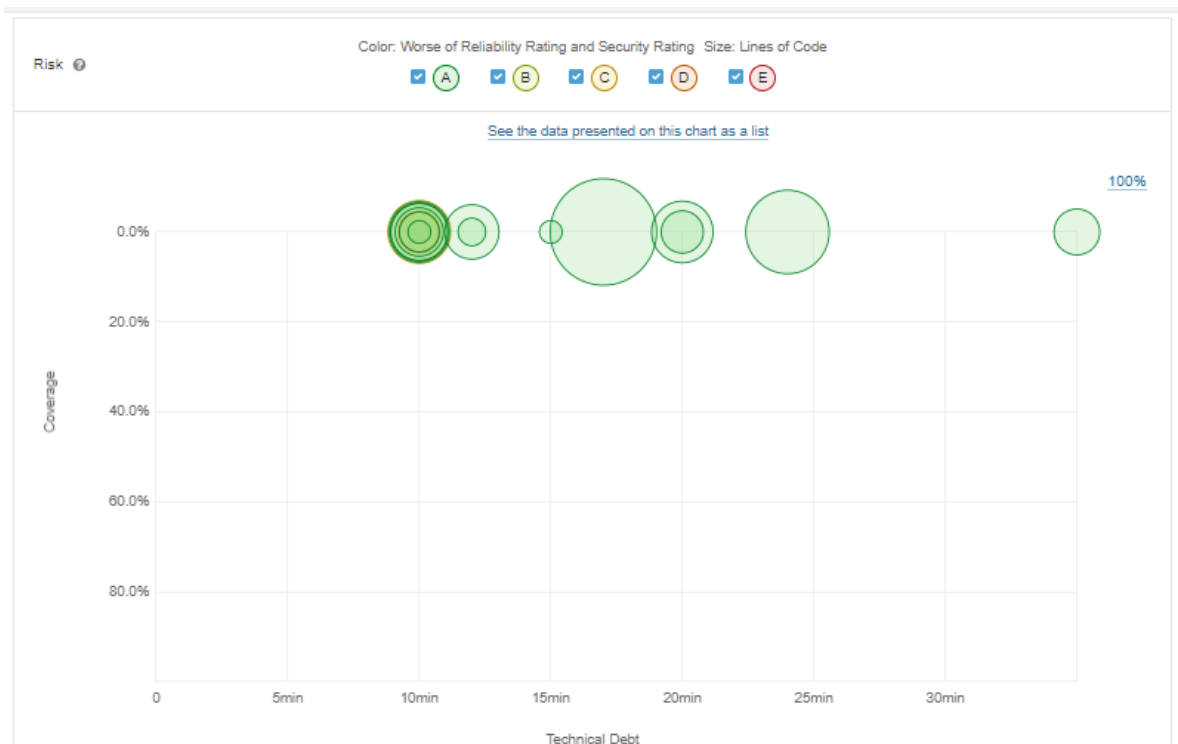
🚩 Call "Optional#isPresent()" before accessing the value.

```
41     return new ResponseEntity(historiaMedica, HttpStatus.ACCEPTED);
    }
```

```
18     @PostMapping("/agendar/citas/{cedula}")
19     public ResponseEntity agendarCita(@PathVariable String cedula, @RequestBody Cita cita){
20         cita.setPaciente(pacienteRepository.getPacientes().
21             stream().filter(p -> p.getCedula()
22             .equals(cedula)).findFirst()
    }
```

🚩 Call "Optional#isPresent()" before accessing the value.

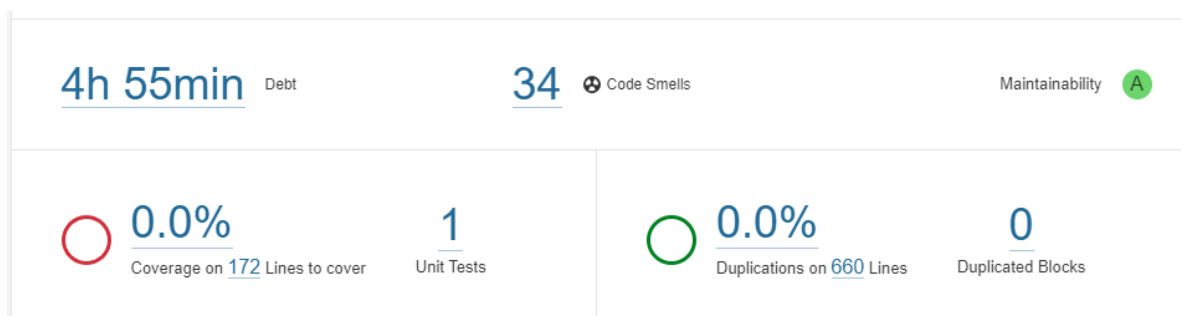
```
23         .get());
    }
```



Vemos que en general no tenemos riesgos solo tenemos los primeros bugs al minuto 10 de correr el código más no son críticos

	Lines of Code	Bugs	Vulnerabilities	Code Smells	Security Hotspots	Coverage	Duplications
com.softka.softkaU.qa.consultorio:consultorio							
L src	553	4	0	34	0	0.0%	0.0%
L target/surefire-reports	60	0	0	0	0	—	0.0%
L pom.xml	47	0	0	0	0	—	0.0%

Vemos un resumen del análisis estático



Mantenibilidad del código y pruebas unitarias