



UNIVERSIDAD  
**NACIONAL**  
DE COLOMBIA


Universidad Nacional de Colombia - sede Bogotá  
Facultad de Ingeniería  
Departamento de Sistemas e Industrial  
Curso: Ingeniería de Software 1 (2016701)

## COOKTRACK

### Linter

Linter: CheckStyle 

#### Configuración:

Se utilizaron las reglas de estilo de Google para Java, conocidas como Google Java Style Guide , son un conjunto de convenciones que especifican cómo debe escribirse el código Java para garantizar consistencia, legibilidad y mantenibilidad.

- **Líneas largas permitidas en imports, URLs o etiquetas HTML**

Se permite que ciertas líneas excedan el límite de 100 caracteres si coinciden con el patrón: `^package.*|^import.*|a href|href|http://|https://|ftp://.`

- **Supresión personalizada con XPath**

Se utiliza un filtro `SuppressionXPathSingleFilter` con una consulta XPath específica para suprimir casos particulares del módulo `RightCurlyAlone`.

Esto desactiva advertencias en estructuras de llaves `{ }` solas, si cumplen ciertas condiciones del AST.

- **Separadores como DOT, COMMA, ELLIPSIS, ARRAY\_DECLARATOR, METHOD\_REF tienen reglas de salto de línea personalizadas**

Cada tipo de separador tiene una política distinta (`EOL`, `n1`, etc.) definida explícitamente, imponiendo un estilo específico sobre cómo deben colocarse los separadores en el código.

- **Estilo de nombres de tipos genéricos (type parameters)**

Formato personalizado para parámetros de tipo (`^[A-Z][0-9]?$` o `([A-Z][a-zA-Z0-9]*[T]$)`), distinto al típico uso de una sola letra. Indica una convención de nombres de tipo muy específica y controlada.

- **Indentación personalizada**

La indentación está ajustada a 2 espacios para la mayoría de elementos (en lugar de los típicos 4).

- **Solo se documentan métodos públicos en Javadoc**

`JavadocMethod` y `MissingJavadocMethod` están configurados para revisar solo métodos públicos, y permiten la omisión de tags si están anotados con `@Override` o `@Test`.

- **Empty blocks permitidos en más contextos**

El `EmptyBlock` permite bloques vacíos en estructuras como `try`, `catch`, `if`, etc., pero requiere que tengan al menos contenido textual (`TEXT`), no necesariamente código.

- **Excepción para bloques vacíos con `{}` solos**

Mensaje personalizado en `WhitespaceAround` para aceptar llaves `{}` como bloques vacíos, siempre que no estén en declaraciones múltiples.

## Resultados obtenidos

Durante la ejecución, el linter arrojó múltiples advertencias relacionadas con la indentación. Esto se debe a que, aunque la configuración de Google espera una indentación de 2 espacios, el IDE utilizado (IntelliJ IDEA) ha formateado automáticamente el código a una indentación de 4 espacios, generando inconsistencias respecto a la guía configurada. Entre los errores y advertencias relevantes se destacan:

- Incorrect indentation level for class definitions.
- Incorrect indentation level for method definitions.
- Incorrect indentation level for if-else statements.

## Evidencia de ejecución

```
[INFO] Scanning for projects...
[INFO]
[INFO] -----< org.example:CookTrack >-----
[INFO] Building CookTrack 1.0-SNAPSHOT
[INFO]   from pom.xml
[INFO] -----[ jar ]-----
[INFO]
[INFO] --- checkstyle:3.3.0:check (default-cli) @ CookTrack ---
[INFO] You have 0 Checkstyle violations.
[INFO]
[INFO] BUILD SUCCESS
[INFO]
[INFO] Total time:  2.013 s
[INFO] Finished at: 2025-07-19T22:08:51-05:00
[INFO]
C:\Users\juane\unal\ingesoft 1\course\Proyecto\CookTrack>
```