

Facultad de matemáticas

Reporte de casos de pruebas para el Módulo Contador de metodos

Versión 1.0

Autores:

Josué Israel Canul Ordoñez

Mónica Garcilazo Cuevas

José Carlos Leo Fernández

Endrick Alfredo Pool Flores

Samuel David Rodríguez Coral

**CONTROL DE DOCUMENTACIÓN**

**Control de Configuración**

| **Título:** | Reporte de casos de pruebas para el Módulo de gestor de archivos |
| --- | --- |
| **Referencia:** |  |
| **Autor(es):** | Samuel David Rodriguez Coral |
| **Fecha:** | 03/04/2025 |

**Histórico de versiones**

| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre de archivo** |
| --- | --- | --- | --- | --- |
| 2.0 | 03/04/25 | A | Samuel David Rodriguez Coral | REP\_CasosPruebasContadorDeMetodos\_v1.0\_2025-04-03 |

**Estado:** (B)orrador, (R)evisión, (A)probado

**Histórico de cambios**

| **Versión** | **Fecha** | **Cambios** |
| --- | --- | --- |
| 1.0 | 03/04/2025 | Primera versión. |

**Índice**

[**1. Introducción 4**](#_1pbytlxhdrny)

[**2. Casos de pruebas unitarias del módulo contador de métodos 4**](#_i2oged897ftd)

[2.1. Contar Métodos en Archivo sin Método 4](#_o7v5mu9u6acb)

[2.1. Contar Métodos en Archivo sin Método 9](#_qn8ytftktqfa)

[2.1. Contar Métodos en Archivo sin Método 14](#_od1uh6fvqxty)

[2.1. Contar Métodos en Archivo sin Método 19](#_xy1lshmsmi0c)

# **Introducción**

Este documento detalla los casos de prueba unitarios para la verificación y validación del módulo de contador de metodos, el cual recibe un archivo java y dentro del archivo identifica la cantidad de metodos que este contiene

# **Casos de pruebas unitarias del módulo contador de métodos**

## **2.1. Contar Métodos en Archivo sin Método**

*Tabla 3 - Caso de prueba 2.1*

| **Caso de prueba** | Leer Archivo Existente | | |
| --- | --- | --- | --- |
| **Identificador caso de prueba** | CP01\_ArchivoSinMetodos | | |
| **Función probar** | MethodLineCounter.count | | |
| **Ubicación de la prueba** | com\example\MethodLineCounterTest.java | | |
| **Metodo donde se realiza la prueba** | testCount\_NoMethods | | |
| **Objetivo** | Verificar que el método devuelve 0 cuando el archivo no contiene métodos. | | |
| **Descripción** | Se proporcionará un archivo Java sin métodos y se validará que el contador retorne 0. | | |
| **Criterios de éxito** | El metodo retorna 0. | | |
| **Criterios de falla** | El método retorna un número distinto de 0 o genera una excepción. | | |
| **Precondiciones** | El archivo de prueba debe existir y no contener métodos. | | |
| **Necesidades para el caso de prueba** | Archivo a leer:  "public class TestClass {",  " private int x;",  " // No methods here",  "}" | | |
| **Autor** | Samuel David Rodriguez Coral | | |
| **Fecha de creación** | 01/04/2025 | | |
| **Flujo del caso de prueba** | **No paso** | **Usuario del sistema** | **Sistema** |
| **1** | **Crear un archivo Java sin métodos.** |  |
| **2** | **Ejecutar el método count().** |  |
| **3** |  | **Verificar que el método retorna 0.** |
| **Post condiciones** | Se obtiene la lista esperada sin errores. | | |

*Tabla 4 - Resultado del caso de prueba 2.1*

| **Número de ejecución** | 1 | |
| --- | --- | --- |
| **Estatus (Pasó / No Pasó)** | Pasó | |
| **Responsable de la ejecución** | Samuel David Rodriguez Coral | |
| **Fecha de ejecución** | 03/04/2025 | |
| **Salida** | La consola indico que el equalsAssert fue true a que la clase no contiene metodos | |
| **Faltas detectadas** | | |
| **Descripción falta** | | **Severidad (Alta, Media, Baja)** |
| N/A | | |

## 

## **2.2. Contar Métodos en Archivo con Un Método**

*Tabla 3 - Caso de prueba 2.1*

| **Caso de prueba** | Leer Archivo Existente | | |
| --- | --- | --- | --- |
| **Identificador caso de prueba** | CP02\_ArchivoUnMetodo | | |
| **Función probar** | MethodLineCounter.count | | |
| **Ubicación de la prueba** | com\example\MethodLineCounterTest.java | | |
| **Metodo donde se realiza la prueba** | testCount\_SingleMethod | | |
| **Objetivo** | Verificar que el método detecta correctamente un único método en un archivo Java. | | |
| **Descripción** | Se proporcionará un archivo con un solo método y se validará que el contador retorne 1. | | |
| **Criterios de éxito** | El método retorna 1. | | |
| **Criterios de falla** | El método retorna un número incorrecto o genera una excepción. | | |
| **Precondiciones** | El archivo de prueba debe existir y contener exactamente un método. | | |
| **Necesidades para el caso de prueba** | Archivo a leer:  "public class TestClass {",  " public void myMethod() { }",  "}"  )); | | |
| **Autor** | Samuel David Rodriguez Coral | | |
| **Fecha de creación** | 01/04/2025 | | |
| **Flujo del caso de prueba** | **No paso** | **Usuario del sistema** | **Sistema** |
| **1** | **Crear un archivo Java con un único método.** |  |
| **2** | **Ejecutar el método count().** |  |
| **3** |  | **Verificar que el método retorna 1.** |
| **Post condiciones** | Se obtiene el valor esperado | | |

*Tabla 4 - Resultado del caso de prueba 2.2*

| **Número de ejecución** | 1 | |
| --- | --- | --- |
| **Estatus (Pasó / No Pasó)** | Pasó | |
| **Responsable de la ejecución** | Samuel David Rodriguez Coral | |
| **Fecha de ejecución** | 03/04/2025 | |
| **Salida** | Retorno el valor esperado que era 1 | |
| **Faltas detectadas** | | |
| **Descripción falta** | | **Severidad (Alta, Media, Baja)** |
| N/A | | |

## 

## **2.3. Contar Métodos en Archivo con Múltiples Métodos**

*Tabla 3 - Caso de prueba 2.1*

| **Caso de prueba** | Leer Archivo Existente | | |
| --- | --- | --- | --- |
| **Identificador caso de prueba** | CP03\_ArchivoMultiplesMetodos | | |
| **Función probar** | MethodLineCounter.count | | |
| **Ubicación de la prueba** | com\example\MethodLineCounterTest.java | | |
| **Método donde se realiza la prueba** | testCount\_MultipleMethods | | |
| **Objetivo** | Verificar que el método detecta correctamente múltiples métodos en un archivo Java. | | |
| **Descripción** | Se proporcionará un archivo con dos métodos y se validará que el contador retorna el número de métodos correcto. | | |
| **Criterios de éxito** | Retorna el numero de métodos correcto | | |
| **Criterios de falla** | El método retorna un número incorrecto o genera una excepción. | | |
| **Precondiciones** | Debe existir un archivo válido en el directorio especificado. | | |
| **Necesidades para el caso de prueba** | Archivo a leer:  public class TestClass {  public void methodOne() { }  private int methodTwo(int x) { return x; }  } | | |
| **Autor** | Samuel David Rodriguez Coral | | |
| **Fecha de creación** | 01/04/2025 | | |
| **Flujo del caso de prueba** | **No paso** | **Usuario del sistema** | **Sistema** |
| **1** | **Crear un archivo Java con multiples metodos** |  |
| **2** | **Ejecutar el método count()..** |  |
| **3** |  | **Verifica que se retorna el numero correcto de métodos** |
| **Post condiciones** | Se obtiene el resultado esperado sin errores. | | |

*Tabla 4 - Resultado del caso de prueba 2.3*

| **Número de ejecución** | 1 | |
| --- | --- | --- |
| **Estatus (Pasó / No Pasó)** | Pasó | |
| **Responsable de la ejecución** | Samuel David Rodriguez Coral | |
| **Fecha de ejecución** | 03/04/2025 | |
| **Salida** | Retorno el resultado correcto de métodos que eran 2 | |
| **Faltas detectadas** | | |
| **Descripción falta** | | **Severidad (Alta, Media, Baja)** |
| N/A | | |

## 

## **2.4. Contar Métodos en Archivo con Método Comentado**

*Tabla 3 - Caso de prueba 2.1*

| **Caso de prueba** | Leer Archivo Existente | | |
| --- | --- | --- | --- |
| **Identificador caso de prueba** | CP04\_ArchivoMetodoComentado | | |
| **Función probar** | MethodLineCounter.count | | |
| **Ubicación de la prueba** | com\example\MethodLineCounterTest.java | | |
| **Metodo donde se realiza la prueba** | testCount\_CommentedMethod | | |
| **Objetivo** | Verificar que el método ignora correctamente los métodos comentados y solo cuenta los activos. | | |
| **Descripción** | Se proporcionará un archivo con un método comentado y otro activo, verificando que solo se cuente el activo. | | |
| **Criterios de éxito** | El método retorna el valor correcto de metodos. | | |
| **Criterios de falla** | El método cuenta el método comentado o genera una excepción. | | |
| **Precondiciones** | El archivo de prueba debe existir y contener un método comentado y otro activo. | | |
| **Necesidades para el caso de prueba** | public class TestClass {  // public void commentedMethod() { }  public void actualMethod() { }  } | | |
| **Autor** | Samuel David Rodriguez Coral | | |
| **Fecha de creación** | 01/04/2025 | | |
| **Flujo del caso de prueba** | **No paso** | **Usuario del sistema** | **Sistema** |
| **1** | **Crear un archivo Java con un método comentado y otro activo.** |  |
| **2** | **Ejecutar el método count().** |  |
| **3** |  | **Verificar que el método retorna 1.** |
| **4** |  | **Retorna una lista con las líneas exactas del archivo.** |
| **Post condiciones** | Se obtiene el resultado esperado sin errores. | | |

*Tabla 4 - Resultado del caso de prueba 2.2*

| **Número de ejecución** | 1 | |
| --- | --- | --- |
| **Estatus (Pasó / No Pasó)** | Pasó | |
| **Responsable de la ejecución** | Samuel David Rodriguez Coral | |
| **Fecha de ejecución** | 03/04/2025 | |
| **Salida** | Retorna 1, por lo que ignoro el método comentado | |
| **Faltas detectadas** | | |
| **Descripción falta** | | **Severidad (Alta, Media, Baja)** |
| N/A | | |

## 