



**UADY**  
FACULTAD DE  
MATEMÁTICAS

**Universidad Autónoma de Yucatán**

Facultad de Matemáticas

Licenciatura en Ingeniería de Software

**Mantenimiento de Software**

M.I.T. Carlos Benito Mojica Ruiz

**Requisitos de Software**

Versión 1.0

**Integrantes de Equipo:**

Canul Ordoñez, Josué Israel

Garcilazo Cuevas, Mónica

Leo Fernández, José Carlos

Pool Flores, Endrick Alfredo

Rodríguez Coral, Samuel David

### CONTROL DE DOCUMENTACIÓN

Título:	Requisitos de Software para el Proyecto Contador Azul
Referencia:	Canul, J. I., Garcilazo, M., Leo, J. C., Pool, E. A., & Rodríguez, S. D. (2025, febrero 19). <i>Requisitos de software para el proyecto Contador Azul</i>
Autor:	Canul, J. I., Garcilazo, M., Leo, J. C., Pool, E. A., & Rodríguez, S. D.
Fecha:	19 de febrero del 2025

### HISTÓRICO DE VERSIONES

Versión	Fecha	Estado	Responsable	Nombre de archivo
1.0	19/10/2024	A	Endrick Alfredo Pool Flores	Requisitos de Software (V1.0).docx

### HISTÓRICO DE CAMBIOS

Versión	Fecha	Cambios
1.0	19/02/2024	Definición de los requisitos de software.

# Índice

<b>1. Introducción.....</b>	<b>3</b>
1.1 Propósito.....	3
1.2 Ámbito del Sistema.....	3
1.3 Definiciones, Acrónimos y Abreviaturas.....	3
1.3.1 Definiciones.....	3
1.3.2 Acrónimos.....	3
1.3.3 Abreviaturas.....	3
1.4 Referencias.....	3
1.5 Visión General del Documento.....	4
<b>2. Descripción General.....</b>	<b>5</b>
2.1 Perspectiva del Producto.....	5
2.2 Funciones del Producto.....	5
2.3 Características de los Usuarios.....	5
2.4 Restricciones.....	6
2.5 Suposiciones y Dependencias.....	6
2.5.1 Suposiciones.....	6
2.5.2 Dependencias.....	6
<b>3. Requisitos Específicos.....</b>	<b>7</b>
3.1 Interfaces Externas.....	7
3.1.1 Interfaces de usuario.....	7
3.1.2 Interfaces de hardware.....	7
3.1.3 Interfaces de software.....	7
3.1.4 Interfaces de comunicación.....	7
3.2 Funciones.....	7
3.2.1 Leer un archivo de código fuente.....	7
3.2.2 Conteo del número total de líneas físicas.....	8
3.2.3 Conteo del número total de líneas lógicas.....	8
3.2.3 Generar el reporte de los resultados.....	8
3.3 Requisitos de Rendimiento.....	8
3.4 Restricciones de Diseño.....	9
3.5 Atributos del Sistema.....	9
3.5.1 Usabilidad.....	9
3.5.2 Seguridad.....	9
3.5.3 Mantenibilidad.....	10
3.5.4 Portabilidad.....	10
3.6 Otros Requisitos.....	10
<b>4. Apéndices.....</b>	<b>11</b>

# 1. Introducción

## 1.1 Propósito

El presente documento define los requisitos para el desarrollo del programa *Contador Azul*, que analiza código fuente para contar la cantidad total de líneas lógicas y físicas presentes en el mismo; de acuerdo con lo establecido en los documentos [Estándar de Conteo](#) y [Estándar de Codificación](#).

## 1.2 Ámbito del Sistema

El software permitirá analizar los archivos de código fuente escritos en el lenguaje de programación Java especificados por el usuario y que hayan sido correctamente compilados.

## 1.3 Definiciones, Acrónimos y Abreviaturas

### 1.3.1 Definiciones

**Línea física:** Toda línea que no es una línea en blanco o que es un comentario.

**Línea lógica:** Cualquier sentencia que ejecuta una acción dentro del flujo del programa.

**SOLID:** Principios de Diseño en Programación Orientada a Objetos.

### 1.3.2 Acrónimos

**MB:** Megabyte

### 1.3.3 Abreviaturas

No se incluyen abreviaturas en el documento.

## 1.4 Referencias

IEEE Computer Society. (1998). IEEE Std 830-1998: IEEE Recommended Practice for Software Requirements Specifications. IEEE.

## **1.5 Visión General del Documento**

El resto del documento de Especificación de Requisitos de Software (ERS) se estructurará en tres secciones principales, cada una abordando aspectos específicos para comprender el sistema a desarrollar. La primera sección, "Descripción General", contextualiza los factores que afectan en el producto y sus requisitos, estableciendo así una guía para comprender los requisitos detallados en la sección siguiente. La segunda sección, "Requisitos Específicos", define los requisitos con un alto nivel de detalle que permita a los diseñadores e ingenieros crear un sistema que cumpla las necesidades de las partes interesadas, así como a los equipos de prueba planificar y ejecutar pruebas que validen su cumplimiento. Finalmente, la sección de "Apéndices" albergará información relevante para la ERS, que, aunque no forme parte directa del documento, complementará la comprensión del sistema a desarrollar.

## 2. Descripción General

### 2.1 Perspectiva del Producto

El programa funciona como una herramienta de análisis estática de código que indicará el número de líneas lógicas y físicas que se encuentran en el código, basándose en lo establecido en los documentos [Estándar de Conteo](#) y [Estándar de Codificación](#).

### 2.2 Funciones del Producto

El sistema deberá dar soporte a las siguientes funcionalidades:

- **Leer un archivo de código fuente.**
- **Identificar y contar líneas físicas existentes en el código.**
- **Identificar y contar líneas lógicas existentes en el código.**
- **Imprimir el reporte con los resultados del análisis.**

Por otro lado, priorizar los requisitos asegura la claridad del proyecto, la eficiente gestión de recursos, la minimización de riesgos y una mejor adaptación a cambios, optimizando así el desarrollo y aumentando las posibilidades de éxito del proyecto. Por lo que se clasifican en tres categorías:

- **Clave:** Son aquellos que son absolutamente esenciales para el éxito del proyecto. Sin estos, el sistema no cumplirá con sus objetivos básicos y no será viable o funcional.
- **Obligatorio:** Son aquellos que deben ser incluidos para que el sistema sea considerado completo y aceptable. Aunque no son tan críticos como los requisitos clave, su ausencia afectaría negativamente la funcionalidad y la usabilidad del sistema.
- **Opcional:** Son aquellos que son deseables pero no esenciales. Su inclusión puede mejorar la experiencia del usuario o añadir valor, pero su ausencia no compromete la funcionalidad básica del sistema.

### 2.3 Características de los Usuarios

Para hacer uso del sistema se define un tipo de usuario, el cual tendrá características en general con respecto a características que los identifican.

**Desarrollador:**

- **Edad:** 20+.
- **Ubicación:** México.
- **Nivel educativo:** No se tienen restricciones con respecto al nivel educativo mínimo para hacer uso del sistema.
- **Experiencia con sistemas similares:** Cuentan con un nivel de experiencia medio con respecto a sistemas similares debido a que este tipo de usuario regularmente utilizan plataformas no óptimas.
- **Experiencia tecnológica:** Este tipo de usuarios cuentan con un nivel de conocimiento tecnológico medio.

## **2.4 Restricciones**

En términos de restricciones del software, el sistema deberá ser escrito en el lenguaje de programación Java y operará sin interfaz gráfica, utilizando solo la consola de comandos para realizar sus funciones. Además, todo el código que se quiera analizar por el programa, debe estar correctamente escrito y ser capaz de ser compilado.

## **2.5 Suposiciones y Dependencias**

### **2.5.1 Suposiciones**

Se parte del supuesto de que los archivos de código fuente no presentan errores de sintaxis y están escritos en el lenguaje de programación Java.

### **2.5.2 Dependencias**

El sistema no cuenta con dependencias.

## 3. Requisitos Específicos

### 3.1 Interfaces Externas

#### 3.1.1 Interfaces de usuario

El sistema funcionará en la consola de comandos, de tal forma que permita la interacción a través de líneas de comando.

#### 3.1.2 Interfaces de hardware

No aplican interfaces de hardware externas, ya que el proyecto se ha diseñado para funcionar de manera independiente, sin la necesidad de conectarse a dispositivos de hardware externos.

#### 3.1.3 Interfaces de software

El sistema se ejecutará en un entorno de sistema operativo compatible con la terminal de comandos.

#### 3.1.4 Interfaces de comunicación

No aplican interfaces de comunicación, ya que el proyecto se ha diseñado para funcionar de manera local mediante los archivos de texto y la consola de comandos..

### 3.2 Funciones

#### 3.2.1 Leer un archivo de código fuente

**RF-001.** El sistema debe permitir la selección de un archivo de código fuente.

**Prioridad:** Obligatorio.

**RF-002.** El sistema debe ser capaz de detectar si el archivo seleccionado está escrito en el lenguaje apropiado.

**Prioridad:** Obligatorio.

**RF-003.** El sistema debe ser capaz de procesar el archivo seleccionado

**Prioridad:** Obligatorio.



**RF-004.** El sistema debe excluir un archivo con errores de sintaxis.

**Prioridad:** Obligatorio.

### **3.2.2 Conteo del número total de líneas físicas**

**RF-006.** El sistema debe reconocer una línea física.

**Prioridad:** Obligatorio.

**RF-007.** El sistema debe llevar el conteo de las líneas físicas.

**Prioridad:** Obligatorio.

**RF-008.** El sistema debe respetar el estándar de conteo para las líneas físicas.

**Prioridad:** Obligatorio.

### **3.2.3 Conteo del número total de líneas lógicas**

**RF-006.** El sistema debe reconocer una línea lógica.

**Prioridad:** Obligatorio.

**RF-007.** El sistema debe llevar el conteo de las líneas lógicas.

**Prioridad:** Obligatorio.

**RF-008.** El sistema debe respetar el estándar de conteo para las líneas lógicas.

**Prioridad:** Obligatorio.

### **3.2.3 Generar el reporte de los resultados**

**RF-006.** El sistema debe generar un reporte con los resultados finales del análisis.

**Prioridad:** Obligatorio.

## **3.3 Requisitos de Rendimiento**

**RR-001.** El sistema debe procesar archivos de hasta 5MB en menos de 5 segundos.

**Prioridad:** Obligatorio.

**RR-002.** Texto

**Prioridad:** Obligatorio.

### 3.4 Restricciones de Diseño

**RRD-001.** El software debe desarrollarse en el lenguaje de programación Java.

**Prioridad:** Obligatorio.

**RRD-002.** El software debe ejecutarse en plataformas Windows, Linux y MacOS.

**Prioridad:** Obligatorio.

**RRD-003.** El software debe ser modular para facilitar futuras mejoras.

**Prioridad:** Clave.

**RRD-004.** La implementación debe seguir los principios SOLID.

**Prioridad:** Clave.

### 3.5 Atributos del Sistema

#### 3.5.1 Usabilidad

**RAC(U)-001.** El software debe contar con mensajes de error descriptivos.

**Requisitos relacionados:** RF-001, RF-002, RF-003, RF-004.

**Prioridad:** Obligatorio.

**RAC(U)-002.** El software debe contar con mensajes de ayuda claros desde la línea de comando para guiar al usuario.

**Requisitos relacionados:** RF-001, RF-002, RF-003, RF-004, RF-006.

**Prioridad:** Obligatorio.

#### 3.5.2 Seguridad

**RAC(S)-001.** Texto

**Requisitos relacionados:** RF-002.

**Prioridad:** Clave.

**RAC(S)-002.** Texto

**Requisitos relacionados:** RF-008.

**Prioridad:** Clave.

### **3.5.3 Mantenibilidad**

**RAC(M)-001.** El código debe estar correctamente documentado siguiendo un estándar para ello.

**Prioridad:** Obligatorio.

**RAC(M)-002.** El código debe ser modular.

**Prioridad:** Obligatorio.

### **3.5.4 Portabilidad**

**RAC(P)-001.** Texto

**Prioridad:** Clave.

## **3.6 Otros Requisitos**

No se cuentan con otros requisitos, ya que, todos los aspectos necesarios para el proyecto ya han sido abordados y detallados en las secciones anteriores del documento.

## 4. Apéndices

No se incluyen apéndices en el documento.