

Facultad de matemáticas

Estimación de Tamaño mediante Puntos de Función

Versión 2.0

Autores:

Aaron Isaac Graniel Arzat

Fernando Joachín Prieto

David Peña Muñoz

Jose Luis Pooc Moo

Andrea Isabel Torres Perez

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:** | Estimación de Tamaño mediante Puntos de Función |
| --- | --- |
| **Referencia:** | REP\_EstimacionTamano\_v2.0\_2025-04-02 |
| **Autor(es):** | Endrick Alfredo Pool Flores |
| **Fecha:** | 02/04/2025 |

**Histórico de versiones**

| **Versión** | **Fecha** | **Estado** | **Responsable** | **Nombre de archivo** |
| --- | --- | --- | --- | --- |
| 1.0 | 26/02/2025 | A | Aaron Isaac Graniel Arzat | REP\_EstimacionTamano\_v1.0\_2025-02-17 |
| 2.0 | 02/04/2025 | A | Endrick Alfredo Pool Flores | REP\_EstimacionTamano\_v2.0\_2025-04-02 |

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

**Histórico de cambios**

| **Versión** | **Fecha** | **Cambios** |
| --- | --- | --- |
| 1.0 | 26/02/2025 | Primera versión. |
| 2.0 | 02/04/2025 | Ajustes de responsable y redacción para ajustarse a los nuevos requisitos. |

**Índice**

[**1. Introducción 4**](#_heading=h.ls2cbcg8pxjk)

[**2. Alcance 4**](#_heading=h.t5qjox9vp91h)

[**3. Identificación y Definición de Funciones según IFPUG 4**](#_heading=h.5h0iuuwupv2i)

[**4. Asignación de Complejidad y Puntos Función 6**](#_heading=h.4lusv6svs205)

[**5. Cálculo de los Puntos Función No Ajustados (UFP) 7**](#_heading=h.2viu77l3v1i3)

[**6. Clasificación del Tamaño del Proyecto según UFP 7**](#_heading=h.nueojb2yee5x)

[**7. Conclusiones 8**](#_heading=h.56ucwysynujr)

[**8. Referencias 8**](#_heading=h.rpa1tytmpgz2)

# **1. Introducción**

Este documento tiene como objetivo estimar el tamaño funcional del software desarrollado, el cual consiste en un programa de línea de comandos que cuenta las líneas físicas y métodos de los archivos ubicados en una ruta especificada. Se utiliza la metodología Function Point Analysis (FPA) siguiendo el enfoque IFPUG, basándonos exclusivamente en los Puntos Función No Ajustados (UFP) y considerando que se trata de un nuevo desarrollo. Esto implica que se medirá únicamente la funcionalidad entregada al usuario sin la necesidad de aplicar ajustes adicionales. Además, la práctica de conteo utilizada corresponde a la versión IFPUG 4.2, la cual está alineada con la norma ISO/IEC 20926. Esto garantiza que los principios básicos del conteo de Puntos de Función están reconocidos internacionalmente y se centran principalmente en la medición del tamaño funcional (UFP). El nombre del proyecto, la fecha de conteo, y los nombres de los contadores y participantes se documentaron conforme a los lineamientos mínimos de la metodología.

# **2. Alcance**

La medición se centra en la funcionalidad principal del sistema, que comprende:

* **Entrada:** Recepción de la ruta (directorio o archivo) desde la cual se obtendrán los archivos a analizar.
* **Proceso:** Análisis y conteo de las líneas de código y métodos presentes en dichos archivos.
* **Salida:** Presentación en la terminal del resultado del conteo (número de líneas físicas y métodos).

Dado que la aplicación es simple y de uso puntual, no se contempla almacenamiento persistente ni integración con otros sistemas.

# **3. Identificación y Definición de Funciones según IFPUG**

La metodología IFPUG clasifica la funcionalidad en cinco categorías principales. A continuación, se explica cada una de ellas con detalle para facilitar la comprensión:

* **Entradas Externas (EI):**Se refieren a las funciones que capturan datos o información provenientes del exterior para ser procesados por el sistema. Estas entradas pueden ser, por ejemplo, formularios, comandos o archivos que el usuario introduce.
* **Salidas Externas (EO):**Corresponden a las funciones que generan información procesada por el sistema y la presentan al usuario. Esta información puede mostrarse en forma de reportes, mensajes o resultados finales.
* **Consultas Externas (EQ):**Son funciones que permiten al usuario consultar datos sin que ello modifique el estado interno del sistema. Se diferencian de las salidas en que no implican procesos de transformación o actualización de la información, sino únicamente su recuperación.
* **Archivos Lógicos Internos (ILF):**Representan conjuntos de datos o archivos que el sistema mantiene y gestiona de forma interna. Estos archivos suelen ser bases de datos o estructuras de almacenamiento que el sistema utiliza para guardar información de manera persistente.
* **Archivos de Interfaz Externa (EIF):**Se refieren a los datos o archivos que utiliza el sistema pero que son mantenidos por otro sistema externo. Es decir, el sistema accede a estos archivos sin tener la responsabilidad de actualizarlos o gestionarlos internamente.

**Identificación en el Proyecto**

1. **Entrada Externa (EI):**
   * El usuario proporciona la ruta o directorio de archivos que se analizarán.
   * **Cantidad:** 1
2. **Salida Externa (EO):**
   * Se muestra en la terminal el reporte con el conteo de líneas físicas y métodos.
   * **Cantidad:** 1
3. **Consultas Externas (EQ):**
   * Nuestro sistema no implementa ninguna funcionalidad de consulta interactiva; es decir, no hay un mecanismo para solicitar información adicional sin modificar el estado, por lo que no se identifica ninguna EQ.
   * **Cantidad:** 0
4. **Archivos Lógicos Internos (ILF):**
   * El sistema no almacena datos de forma interna, ya que simplemente procesa la información de los archivos existentes en el directorio proporcionado y no crea ni mantiene ningún archivo o base de datos propia.
   * **Cantidad:** 0
5. **Archivos de Interfaz Externa (EIF):**
   * El sistema no interactúa con datos que provengan de otros sistemas ni requiere acceder a archivos que se mantengan externamente, por lo que no se identifican EIF.
   * **Cantidad:** 0

# **4. Asignación de Complejidad y Puntos Función**

A continuación, se presenta la **tabla de referencia** utilizada para determinar la cantidad de Puntos Función, de acuerdo con la complejidad de cada función:

*Tabla 1 - Resumen de un recuento de puntos de función*

| **Componente** | **Complejidad Baja** | **Complejidad Media** | **Complejidad Alta** |
| --- | --- | --- | --- |
| **EI** | 3 PF | 4 PF | 6 PF |
| **EO** | 4 PF | 5 PF | 7 PF |
| **EQ** | 3 PF | 4 PF | 6 PF |
| **ILF** | 7 PF | 10 PF | 15 PF |
| **EIF** | 5 PF | 7 PF | 10 PF |

**Asignación en el Proyecto**

* **Entrada Externa (EI):**
  + La única entrada, que consiste en el ingreso de la ruta, es simple y no requiere múltiples validaciones, por lo que se clasifica como de baja complejidad.
  + **Puntos asignados:** 3 PF
* **Salida Externa (EO):**
  + La salida es un reporte sencillo en consola que muestra el conteo de líneas, sin procesamiento complejo adicional.
  + **Puntos asignados:** 4 PF
* **EQ, ILF y EIF:**
  + Al no existir funciones de consulta (EQ), almacenamiento interno (ILF) ni acceso a archivos externos (EIF) en este sistema, se asigna 0 PF para cada uno.

# **5. Cálculo de los Puntos Función No Ajustados (UFP)**

La suma de los puntos asignados es la siguiente:

* **EI:** 1 × 3 = 3 PF
* **EO:** 1 × 4 = 4 PF
* **Total UFP =** 3 + 4 = 7 PF

# **6. Clasificación del Tamaño del Proyecto según UFP**

A continuación, se muestra una tabla de clasificación de proyectos en función de su tamaño medido en UFP, desarrollada a partir de la “Rule’s Relative Size Scale” como referencia y adaptada para facilitar la interpretación en nuestro contexto. Esta tabla es una guía orientativa que permite clasificar un proyecto como “Muy pequeño”, “Pequeño”, “Mediano”, etc., basándose en rangos de UFP:

*Tabla 2 - Clasificación del tamaño según la cantidad de puntos de función*

| **Categoría de Tamaño** | **Rango (uFP)** |
| --- | --- |
| **Muy pequeño** | 0 – 30 |
| **Pequeño** | 30 – 150 |
| **Medio** | 150 – 1000 |
| **Grande** | 1000 – 5000 |
| **Muy grande** | 5000 – 15000 |
| **Extragrande** | > 15000 |

Dado que nuestro proyecto tiene un total de 7 uFP, se clasificaría dentro de la categoría Muy pequeño. Esto indica que la aplicación posee una funcionalidad mínima y que, en consecuencia, el esfuerzo inicial de desarrollo y las labores de mantenimiento serán moderados.

# **7. Conclusiones**

Basado en los resultados del análisis y la medición utilizando el método de Function Point Analysis, el sistema analizado, programa para contar las líneas físicas y los métodos del código en archivos dentro de un directorio especificado, obtuvo un total de 7 Puntos de Función. Este resultado clasifica al sistema como extremadamente pequeño, lo que indica que se trata de una herramienta altamente específica con una complejidad funcional mínima.

Dada esta clasificación, el sistema es fácil de desarrollar y mantener, ya que consta únicamente de una entrada y una salida sin necesidad de almacenamiento interno de datos ni procesamiento complejo. Su simplicidad lo hace altamente accesible y fácil de usar, ya que no requiere capacitación extensa ni conocimientos avanzados para su operación. Además, debido a su tamaño reducido, cualquier modificación, actualización o expansión puede implementarse rápidamente y con un esfuerzo mínimo.

# **8. Referencias**

Bundschuh, M., & Dekkers, C. (2008). The IFPUG Function Point Counting Method. En The IT Measurement Compendium (pp. 324–356). doi:10.1007/978-3-540-68188-5

Software Measurement Services Ltd. (2005). *Rule’s Relative Size Scale (Version 1.b)*. Software Measurement Services Ltd. Recuperado de:<https://totalmetrics.com/function-points-downloads/Function-Point-Scale-Project-Size.pdf>