**Importancia de Medir:**

\*Estimación

\*Evaluación de productividad

\*Control de proyectos

\*Calidad del Software

\*Tomar decisiones tácticas

**Métrica**: medida cuantitativa del grado que posee un atributo de un sistema, componente o proceso. Se recopilan medidas y desarrollan métricas para obtener indicadores.

Indicador: métrica o combinación de métricas que proporcionan una visión profunda del proceso del software, del proyecto de software, o el producto en sí.

**Utilidad de la métrica:**

• Evaluar el estado del proyecto en curso

• Seguir la pista de los riesgos potenciales

• Detectar las áreas de problema antes de que se conviertan en "críticas"

• Ajustar el flujo y las tares del trabajo

• Evaluar la habilidad del equipo del proyecto en controlar la calidad de los productos de trabajo de la ingeniería del software.

**Métricas de Software:**

**Directas**:

– Líneas de código (LDC) producidas

– Velocidad de ejecución

– Tamaño de memoria utilizado

– Defectos informados durante un período de tiempo

**Indirectas**:

– Funcionalidad

– Calidad

– Complejidad

– Facilidad de mantenimiento

**Clasificación de Métricas:**

**Orientadas al tamaño:** Errores por KLDC, $ por KLDC, LDC por persona-mes. No están aceptadas como las mejores para medir el proceso de desarrollo.

**Orientadas a la función**: miden la funcionalidad que ofrece una aplicación. La funcionalidad no puede medirse de forma directa, por lo que debe utilizarse otras métricas directas.

Técnica de puntos de función:

Esta técnica se basa en el **diseño lógico del sistema** y en características fundamentalmente “Externas” de la aplicación a desarrollar.

Los objetivos de los puntos de función son:

• Medir independientemente de la tecnología utilizada en la implantación del sistema.

• Proporcionar una métrica de tamaño que dé soporte al análisis de la calidad y la productividad.

• Proporcionar un medio para la estimación del software.

• Proporcionar un factor de normalización para la comparación de distintos softwares.

Elementos de la técnica:

1. Número de Entradas (EI, External Input).

Transacciones (datos introducidos para mantener ficheros lógicos internos), pantallas de entradas (ABM representa 3 entradas).

No son, datos utilizados por la aplicación, pero no mantenido en ficheros, la entrada en una consulta, pantallas de conexión que no mantengan un fichero lógico interno.

2. Número de Salidas (E0, External Output).

Transferencia de archivos, informes, gráficos, generadores de informes.

No son: ayudas, distintas formas de invocar la misma salida lógica, mensajes de error/confirmación, totalizaciones (informes de totales)

3. Número de Consultas (peticiones de usuario) (EQ, External Query).

4. Número de Archivos (Grupos de datos lógicos) internos (ILF, Internal Logic File).

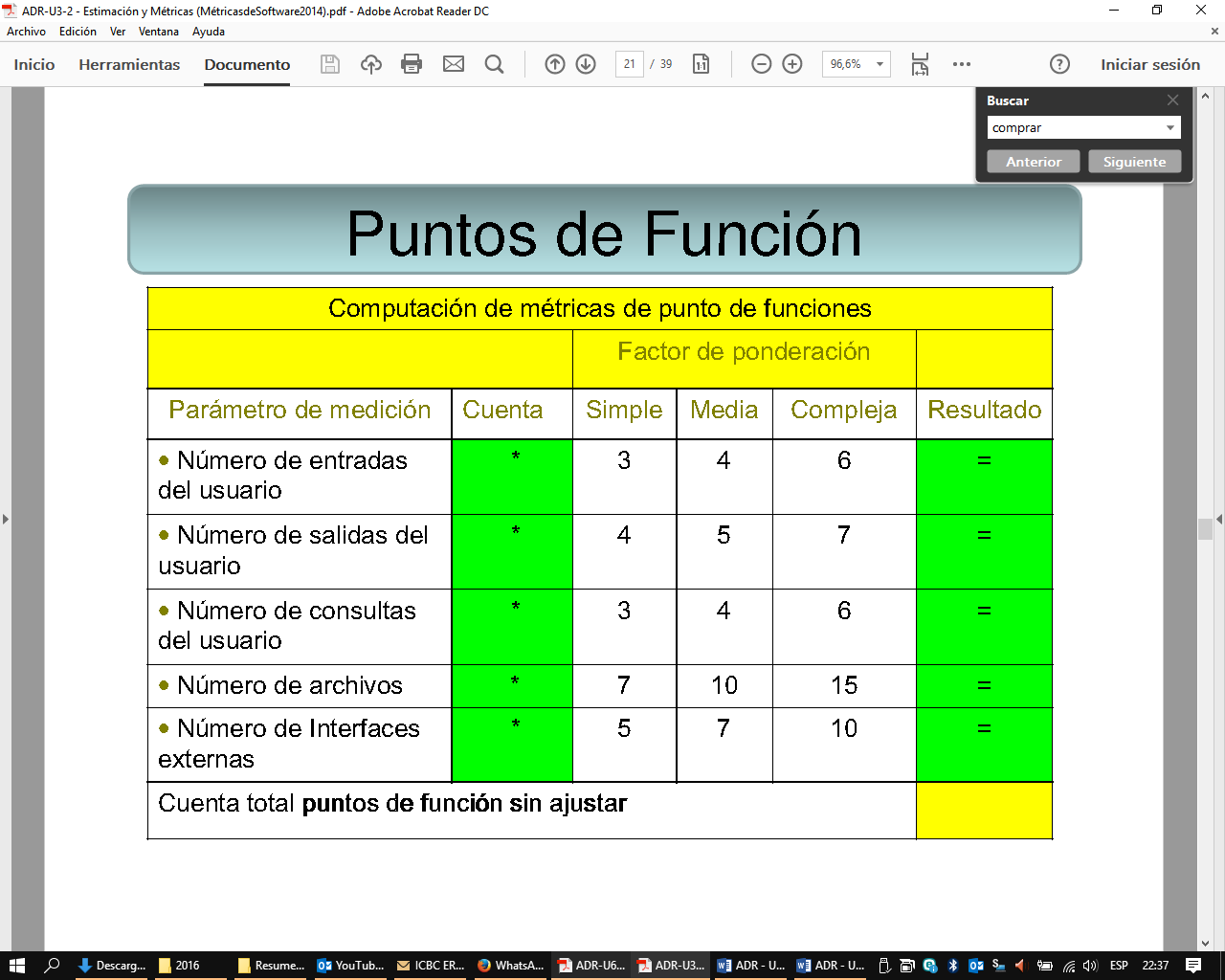
5. Número de interfaces externas (Grupos de datos lógicos externos) (EIF, External Interface File).

Valoración de la Complejidad:

Para cada uno de los parámetros externos se ha de indicar su complejidad como baja, media o alta.

Para las entradas, salidas y consultas, se puede evaluar su complejidad en función del número de campos que contengan y del número de ficheros a los que hagan referencia.

Para los ficheros, por el contrario, su complejidad vendrá dada en función del número de registros y de campos que tengan.



Cálculo del TDI: Una vez calculados el total de puntos de función sin ajustar, debe realizarse un ajuste teniendo en cuenta las características generales del sistema. A cada una de estas características se las pondera con un valor de 0 a 5 (0 No influencia,1 Incidental, 2 Moderado, 3 Medio, 4 Significativo, 5 Esencial)

Luego se calcula el valor del factor de ajuste. AF = (TDI x 0,01) + 0,65

El valor de los puntos de función ajustados será: FPA = FP x AF

Esfuerzo(HS) = PFA \* Promedio\_Organización(Lenguaje)

Esfuerzo(HS) = PFA \* DatosEstadisticos

Medida de Calidad de Software:

• Corrección: La corrección es el grado en el que el software lleva a cabo su función, la medida más común de corrección son los defectos por KLDC.

• Facilidad de mantenimiento: es la facilidad con la que se puede corregir un programa si se encuentra un error o un cambio solicitado por el usuario, para ello se deben utilizar medidas indirectas, como el tiempo medio de cambio (TMC).

• Integridad: Este atributo mide la habilidad de un sistema para resistir ataques (tanto accidentales como intencionados). La integridad del sistema se puede definir como: integridad = Sumatoria [1- amenaza \* (1- seguridad)].

• Facilidad de uso: es un intento de cuantificar «lo amigable que puede ser con el usuario» y se puede medir en función de:

– Habilidad intelectual y/o física requerida para aprender el sistema

– El tiempo requerido para llegar a ser moderadamente eficiente en el uso del sistema

– Aumento neto en productividad.

– Valoración subjetiva de la disposición de los usuarios hacia el sistema.

Métrica de Calidad: proporciona beneficios tanto a nivel del proyecto como del proceso, es la eficacia de la eliminación de defectos (EED).

Se define como: EED = E / (E + D)

E = número de errores encontrados antes de la entrega del software al usuario final

D = número de defectos encontrados después de la entrega.

El valor ideal de EED ES 1.

|  |  |  |
| --- | --- | --- |
| Herramientas de Medición | Ventajas | Desventajas |
| • Juicio Experto |  |  |
| • Estimación Análoga (Top Down): basada en datos históricos de proyectos anteriores similares. | Rápida y de Menor Costo  Las actividades no necesitan estar desagregadas.  Proveen idea a nivel gerencial. | Poco precisa.  Se preparan con conocimiento limitado del proyecto.  Se necesita personal con experiencia.  No toma en cuenta las diferencias entre proyectos. |
| • Estimación Detallada (Button Up): trasladan hacía arriba los estimados individuales para obtener el total del proyecto. | Más preciso que Ascendente  Análisis detallado. | Tiempo y costo.  Tendencia a generar “colchones”.  El proyecto debe estar bien definido antes de comenzar a estimar. |
| • Estimación Paramétrica: definidas por una fórmula en función del esfuerzo por unidad de trabajo. | Muy utilizando cuando se están implementando proyectos similares.  Consiste en determinar variables claves del proyecto que son los principales determinantes del tamaño del proyecto. | |
| • Estimación 3 valores: Se utiliza el promedio ponderado de estimaciones para calcular la duración de una actividad.  (PERT) | Se basa en 3 valores:  Tm: valor más probable.  To: valor optimista.  Tp: valor pesimista.  Tiempo Esperado. Te = (To + 4Tm + Tp) / 6 | |

Validación de la estimación:

• Experiencia: en base a experiencia personal.

• Analogía: comparar con proyectos similares.

• Puntos de Función.

• Registros históricos.

Cronograma

• Precedencia.

• Recursos.

• Hitos previstos.