



***Técnico en***  
**< DESARROLLO DE SOFTWARE >**

***Metodología de Desarrollo de  
Software II***

(CC BY-NC-ND 4.0)  
International

Attribution-NonCommercial-NoDerivatives 4.0



## **Atribución**

Usted debe reconocer el crédito de una obra de manera adecuada, proporcionar un enlace a la licencia, e indicar si se han realizado cambios. Puede hacerlo en cualquier forma razonable, pero no de forma tal que sugiera que tiene el apoyo del licenciante o lo recibe por el uso que hace.



## **No Comercial**

Usted no puede hacer uso del material con fines comerciales.



## **Sin obra derivada**

Si usted mezcla, transforma o crea un nuevo material a partir de esta obra, no puede distribuir el material modificado.

No hay restricciones adicionales - Usted no puede aplicar términos legales ni medidas tecnológicas que restrinjan legalmente a otros hacer cualquier uso permitido por la licencia.

<http://creativecommons.org/licenses/by-nc-nd/4.0/>



# ***Metodología de Desarrollo de Software II***

## ***Unidad IV***

### ***Técnicas De Estimación y Control de Proyectos de Software***

#### **1. Técnicas De Estimación y Control de Proyectos de Software**

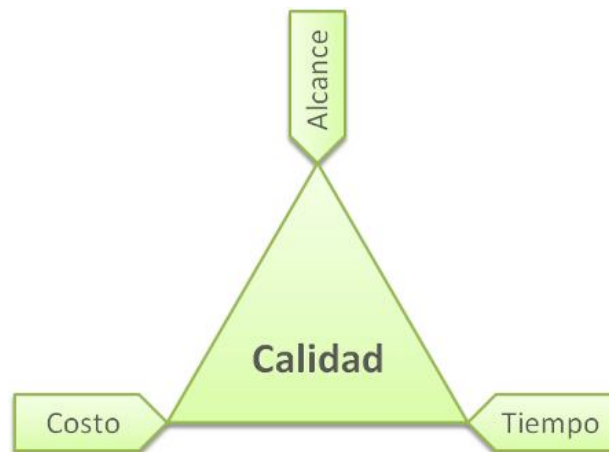
El desarrollo de software es una actividad compleja que depende principalmente del esfuerzo intelectual de las personas, así como de los procesos y la tecnología con la que se desarrolla.

Las metodologías para el desarrollo del software imponen un proceso disciplinado sobre el desarrollo de software con el fin de hacerlo más predecible y eficiente. Una metodología de desarrollo de software tiene como principal objetivo aumentar la calidad del software evaluando en cada etapa los requerimientos.

#### **Triángulo de Hierro**

En todo proyecto existen tres variables relacionadas, el llamado “triángulo de hierro” el alcance el cual refleja los requisitos o tareas para realizar, el tiempo o planificación muestra cuánto durará el proyecto, el costo o recursos que analiza cuánto dinero,

personas, recursos se dedicará al proyecto. Para mantener unos objetivos de calidad determinados, cualquier modificación en una de las tres variables implica la modificación de alguna(s) de las otras dos. (Kniberg, 2012).



## 2. Estimación de costos

Los costos suelen ser cuantificables y estimables en unidades económicas. En un análisis costo/beneficio se debe considerar aquellos aspectos tangibles, cuantificables en valores como dinero, tiempo, recursos, y los intangibles, no ponderables, ni palpables pero si indispensables para analizar los costos de un proyecto. Aunque los beneficios intangibles sean difíciles de cuantificar no hay razón para no tenerlos en cuenta, donde deben estar involucrados todos los interesados en el software (Sexton, 2009). Algunos indicadores económicos para el desarrollo software que se tendrán en cuenta en la investigación.

- **Análisis Costo-Beneficio:** La técnica del análisis costo/beneficio tiene como objetivo fundamental proporcionar una medida de los costos en que se incurre en la realización de un proyecto y comparar dicha previsión de costos con los beneficios esperados de la realización de dicho proyecto.
- **Productividad:** Existen estándares con dedicación exclusiva a esta medida (IEEE, 1992) es también una de las más polémicas e imprecisas. (Oded & Mounford, 2008).
- **Costo-eficiencia:** Este plantea que la conveniencia de la ejecución de un proyecto se determina por la observación conjunta de dos factores, el costo (hardware, software y operaciones) y la eficiencia (bienes y servicios finales).

## Cálculo de los Costos en el Desarrollo de Software

Cuando la necesidad de utilizar el software aumenta en cualquier actividad humana, mayor es también la complejidad y dificultad de implementación que este adquiere. Aunque cada vez existan más técnicas que facilitan el diseño y desarrollo de los sistemas, las nuevas exigencias de los usuarios y los nuevos dominios de aplicación generan nuevos problemas. Creando la necesidad de prestar cada vez más atención a los procesos de planificación, medición y estimación de diversos parámetros software, antes de comenzar con el desarrollo de la aplicación.

El proceso de estimación del costo de un producto software está formado por un conjunto de técnicas y procedimientos que se usan en la organización para poder llegar a una predicción fiable. Este es un proceso continuo, que debe ser usado y consultado a lo largo de todo el ciclo de vida del proyecto. Se divide en los siguientes pasos (Agarwal & Kumar, 2006):

Estimación del tamaño, estimación del costo y del esfuerzo, estimación de la programación temporal, estimación de la cantidad de recursos computacionales, ausencias de riesgos, inspección y aprobación, redacción de informes de estimación, medición y perfeccionamiento del proceso (Agarwal & Kumar, 2001).

## Métricas de Desarrollo de Software

En el campo de la ingeniería de software, se suele hablar indistintamente de “métricas” y de “medidas”, pero sin embargo existen diferencias entre estos términos. Una medida indica cuantitativamente algún atributo de proceso o de producto (extensión, cantidad, dimensiones, capacidad, tamaño, etc.). Una métrica es definida por el glosario de estándares del IEEE (Institute of Electrical and Electronics Engineers) (IEEE Standard Glossary of Software Engineering Terminology, 1993) como una “medida cuantitativa del grado en que un sistema, componente o proceso posee un atributo determinado”. Si se recopila un solo tipo de datos, como por ejemplo el número de errores dentro de un componente, se ha establecido una medida.

Una métrica de software relaciona de alguna manera las medidas individuales, podría tratarse del número de errores encontrados en cada revisión o prueba (Pressman, 1998).

## 3. Técnicas de estimación de costos en el desarrollo de software

Se han desarrollado varias técnicas de estimación para el desarrollo de software estableciendo de antemano el ámbito del proyecto, usar las métricas del software sirven de base para la realización de estimaciones y desglosar el proyecto en partes más pequeñas que se estiman individualmente (Angelis L & Stamelos, 2000).

Debido al crecimiento en la industria del software y siendo una actividad compleja la estimación de costos, las compañías dedicadas a ofrecer diferentes herramientas de estimación de costos en el mercado han aumentado. A partir del 2005, algunas de esas herramientas de estimación son: COCOMO II, CoCots, CoStar, CostModeler, CostXpert, SoftCost.

La estimación de costos implica la realización de predicciones sobre la cantidad más probable de esfuerzo, tiempo y niveles de personal que se requieren para construir un sistema de software. Estas se realizan a lo largo de todo el ciclo de vida del software.

Existen formas de realizar la estimación de los costos éstas pueden ser mediante:

## La opinión de expertos

Un desarrollador o gestor describe los parámetros del proyecto y los expertos hacen estimaciones basadas en su experiencia.

## Técnica Delphi

Permite sistematizar y mejorar la opinión de los expertos consultados.

## Analogía

Enfoque más formal que el de la opinión de expertos. Los expertos comparan el proyecto propuesto con uno o más proyectos anteriores intentando encontrar similitudes y diferencias particulares.

Base Histórica de proyectos, se busca ver hacia el pasado para hacer comparaciones con proyectos que se están trabajando actualmente.

## COCOMO

El Modelo COCOMO creado por Barry Boehm en 1981. Su nombre significa Constructive cost Model (Modelo constructivo de costo) constituye una jerarquía de modelos de estimación para el software. La jerarquía está constituida por los siguientes modelos (COCOMO Model definition manual., 1990).

- **COCOMO básico.** Calcula el esfuerzo y el costo del desarrollo en función del tamaño del programa estimado en LOC.
- **COCOMO intermedio.** Calcula el esfuerzo del desarrollo en función del tamaño del programa y un conjunto de conductores de costo que incluyen la evaluación subjetiva del producto, del hardware, del personal y de los atributos del proyecto.
- **COCOMO detallado.** Incorpora las características de la versión intermedia y lleva a cabo una evaluación del impacto de los conductores de costo en cada fase (análisis, desarrollo, etc.) del proceso.

COCOMO es una herramienta basada en las líneas de código la cual le hace muy poderoso para la estimación de costos y no como otros que solamente miden el esfuerzo en base al tamaño. Es necesario para un administrador de proyectos una herramienta de estimación de costos; y esta herramienta puede estar relacionada con COCOMO ya que esta técnica representa uno de los más completos modelos empíricos para la estimación de software (Bumett, 1998).



## Líneas de Código y Puntos de Función

Las métricas para puntos de función están basadas en las guías proporcionadas por el International Function Point User Group (IEPUG, 1994). Los puntos de función procuran cuantificar la funcionalidad de un sistema de software. La meta es obtener un número que caracterice completamente al sistema. Son útiles estimadores ya que están basados en información que está disponible en las etapas tempranas del ciclo de vida del desarrollo de software.

Los datos de líneas de código (LDC) y los puntos de función (PF) se emplean de dos formas durante la estimación del proyecto de software:

- **Variables de estimación**, utilizadas para calibrar cada elemento del software.
- **Métricas de base**, recogidas de anteriores proyectos utilizadas junto con las variables de estimación para desarrollar proyecciones de costo y esfuerzo.

## Referencia:

Dianelys del Valle Roque, Eduardo Cueto Ible, Pavel Navarro Guerrero, “Desarrollo de Métricas para calcular el costo en el proceso de desarrollo de software”.

.

## ***Descargo de responsabilidad***

La información contenida en este documento descargable en formato PDF o PPT es un reflejo del material virtual presentado en la versión online del curso. Por lo tanto, su contenido, gráficos, links de consulta, acotaciones y comentarios son responsabilidad exclusiva de su(s) respectivo(s) autor(es) por lo que su contenido no compromete al área de e-Learning del Departamento GES o al programa académico al que pertenece.

El área de e-Learning no asume ninguna responsabilidad por la actualidad, exactitud, obligaciones de derechos de autor, integridad o calidad de los contenidos proporcionados y se aclara que la utilización de este descargable se encuentra limitada de manera expresa para los propósitos educacionales del curso.

