

# ¿Por qué las estimaciones no se pueden alcanzar en nuestros proyectos?

**Maikel Muñoz Roja \***

Centro Nacional de Calidad de Software, Cuba, Habana, mroja@uci.cu

**Anisbert Suárez Batista**

Centro Nacional de Calidad de Software, Cuba, Habana, abatista@uci.cu

**Yurixay Trujillo Armas**

Centro Nacional de Calidad de Software, Cuba, Habana, yurixay@uci.cu

## **ABSTRACT**

In the beginning of the project every need, expectation or illusion exists only in the customers minds or max embodied in a document in the form of requirement or business rule, but there is always some uncertainty between what you really want and what is obtained at the end, so we don't know how much it will cost and how to obtain that result. Once a requirement is developed through the analysis and design and the various stages of product development this uncertainty fades and objectives and the way to achieve clearer appear on the horizon of the project. However, when a project is advanced in its development, has been asked an important question that all managers and project leaders are facing the beginning of each project and they have to respond to the available information (mostly poor) in that time: What is the date of delivery of this project? It seems very simple question but one of the most important projects success factors depends of it: delivery on time and also much of client satisfaction. So the response must take into account various factors that may affect the duration of a project. The analysis of these factors-often subjective-that influence the duration of a project is what we intend to do in this paper.

**Keywords:** Estimation, Project Planning, Software Project Management.

## **RESUMEN**

En el inicio del proyecto cada necesidad, expectativa o ilusión se encuentra en la mente de los clientes o cuando más plasmada en algún documento en forma de requerimiento o regla del negocio, pero siempre existe cierta incertidumbre entre lo que realmente se desea y lo que se obtendrá al final así que mucho menos se sabe cómo se hará y cuánto costará obtener dicho resultado. Una vez que un requisito se desarrolla a través del análisis y diseño y atraviesa las diferentes fases del desarrollo del producto esta incertidumbre se va desvaneciendo y los objetivos y el camino para alcanzarlos aparecen más claros en el horizonte del proyecto. Sin embargo, cuando ya un proyecto está avanzado en su desarrollo ya han pasado algunas preguntas impostergables al que todos los gerentes y líderes de proyectos se enfrentan al inicio de cada proyecto y que tienen que responder con la información disponible (casi siempre deficiente) en ese momento: ¿Cuánto tiempo costará este proyecto? ¿Cuánto esfuerzo requerirá? ¿Cuántos recursos serán necesarios? Aunque parezcan preguntas muy simples de ellas dependerá uno de los factores de éxito más importante del proyecto: la entrega en tiempo y junto a ello la satisfacción del cliente, que espera se cumplan los plazos pactados dentro del presupuesto y tiempo esperado; así que para su respuesta hay que tener en cuenta diversos factores que pueden afectar la duración de un proyecto. Precisamente el análisis de estos factores –muchas veces subjetivos- que influyen en la duración de un proyecto es lo que pretendemos realizar en el presente trabajo.

**Palabras claves:** gestión de proyectos, estimación.

## 1. INTRODUCCIÓN

Cuando se inicia un proyecto informático se conoce muy poco sobre las funcionalidades que se deben implementar puesto que lo que se tiene no son más que necesidades, expectativas o ilusiones que se encuentran en las mentes de los clientes por lo que siempre existe cierta incertidumbre entre lo que realmente se desea y lo que se obtendrá al final así que mucho menos se sabe cómo se hará y cuánto costará obtener dicho resultado. A pesar de ello, por lo general es necesario durante el inicio realizar una estimación del tiempo, esfuerzo, costo y otros parámetros que son imprescindibles definir para establecer acuerdos y contratos entre las partes, de modo que con la información disponible en ese momento (casi siempre deficiente) deben obtenerse las estimaciones más precisas posibles porque de ello dependerán varios factores de éxito del proyecto.

*“La gestión de proyectos de software comienza con un grupo de actividades que en conjunto se denominan planificación de proyecto. Antes de que el proyecto comience el gestor del proyecto y el equipo de software deben estimar el trabajo que habrá de realizarse, los recursos que se requerirán y el tiempo que transcurrirá desde el principio hasta el final”. (R. Pressman, 2009a)*

Una vez que un requisito del cliente se desarrolla a través del análisis y diseño y atraviesa las diferentes fases del desarrollo del producto esta incertidumbre se va desvaneciendo y los objetivos y el camino para alcanzarlos aparecen más claros en el horizonte del proyecto. De hecho, un software se especifica a medida que se construye puesto que la propia especificación y el análisis forman parte del desarrollo del producto por tanto cabe esperar que aquella primera estimación tenga sólo unas mínimas posibilidades de convertirse en realidad.

En todo caso, para una estimación aceptable será necesario disponer de diferentes métricas que sean útiles para caracterizar y medir las funcionalidades del producto que se quiere desarrollar y asimismo se necesitan unidades de medida que permitan asociar estas funcionalidades con el esfuerzo que cuesta implementarlas; es decir, se deben tener referencias sobre la productividad en la construcción de software.

*“La estimación de recursos, costo y programa de trabajo para una tarea de ingeniería de software requiere experiencia, acceso a buena información histórica (métricas) y el valor para comprometerse con predicciones cuantitativas cuando la información cualitativa es todo lo que existe. La estimación implica riesgo inherente y este conduce a la incertidumbre”. (R. Pressman, 2009b)*

Sin embargo, disponer de buenas métricas no es suficiente, tampoco lo es contar con modelos bien calibrados ni tener amplias bases de datos históricos; aun así habrán otros factores que afectarán la precisión de las estimaciones.

Precisamente el análisis de algunos de esos factores –muchas veces subjetivos- que afectan las estimaciones es lo que pretendemos realizar en el presente trabajo.

## 2. LA ESTIMACIÓN EN LA GESTIÓN DE PROYECTOS

¿Por qué planteamos que las estimaciones no se pueden alcanzar en nuestros proyectos? Esto puede parecer retórica pero analicemos algunos hechos. A nuestro entender existen un conjunto de factores que interfieren con la precisión de las estimaciones que muy difícilmente se puedan eliminar independientemente de los modelos seleccionados, la cantidad y calidad de los datos históricos y el nivel de confianza alcanzado.

La actividad de la primera estimación de costes de un proyecto informático es y será siempre problemática, sobre todo por el hecho de tener que estimar el esfuerzo necesario para implementar una aplicación cuando todavía no se conocen con detalle las funcionalidades que se han de cubrir.

*“La estimación implica riesgo y este conduce a la incertidumbre”.* (R. Pressman, 2009b)

Estamos de acuerdo que para lograr una estimación aceptable es necesario conocer los diferentes modelos de estimación que se encuentran definidos y seleccionar los más adecuados en cada caso. Sin embargo, durante las últimas dos décadas, los cambios en la ingeniería de software han sido tantos, las metodologías son tan diversas y complejas que a menudo los modelos de estimación disponibles no parecen los más adecuados para la compleja y difícil tarea de estimar.

Veamos entonces algunos de los factores que a nuestro entender afectan de una manera significativa las estimaciones.

## **2.1 MODELOS VS PRÁCTICA REAL DE LA ESTIMACIÓN**

Tal vez el método más utilizado todavía para realizar la estimación es el que se basa en la experiencia personal de quien la lleva a cabo. Como es de suponer se trata de un método marcadamente subjetivo y muy propenso a errores. Puede ocurrir que quien realice la estimación no tenga en cuenta de manera adecuada todas las variables que pueden intervenir en el proyecto y también puede suceder que la experiencia disponible no se corresponda en absoluto con el tipo de proyecto o de sistema que se debe construir.

Por lo general se suelen considerar más serios y seguros los modelos que incorporan fórmulas, pero no debemos olvidar que, muy a menudo, eso no resuelve en absoluto el problema real de la estimación. Los modelos algorítmicos nos permiten, por ejemplo, saber cuál será el esfuerzo necesario para obtener un determinado número de líneas de código (LOC). También nos muestran los meses empleados en la construcción del proyecto, el número de profesionales que deberían estar involucrados, el número de errores que se estima obtener, el número de páginas de documentación que se deben llevar a cabo, etc. Sin embargo, todo parte de un determinado número global de líneas de código o de la estimación del total de puntos de función (PF) que se deben implementar. Si bien, lo cierto es que, aun así, los métodos algorítmicos no resuelven en absoluto el problema de la estimación porque nadie nos dice de dónde sale el número de LOC que utilizaremos en las fórmulas, ni la persona que lo determina.

Por otra parte, en la práctica, disponer de una cifra de esfuerzo única no es en absoluto el camino más fácil para las etapas posteriores en la gestión de un proyecto informático: la planificación y el seguimiento del proyecto. Saber sólo que un proyecto debe durar seis meses no nos permite conocer si, por ejemplo, a tres meses del comienzo se van cumpliendo o no las previsiones realizadas (faltan detalles).

Si tenemos en cuenta que, posteriormente a la estimación de esfuerzo, es necesario planificar en el tiempo y efectuar el seguimiento de todo el conjunto de actividades o tareas que forman el proyecto, el hecho de tener una estimación global única es poco útil. Es mucho más interesante un planteamiento ascendente que nos permita, antes de efectuar la estimación, descomponer el proyecto en diferentes actividades. Este paso previo lo convierte en mucho más planificable y controlable.

La mayoría de métodos de estimación utilizados hasta ahora se centran en dos grandes unidades de medida: el número de líneas de código para medir el tamaño y los puntos de función para medir las funcionalidades. Se da una relación empírica entre LOC y PF, y la tendencia actual parece que es el predominio de los puntos de función e, incluso, la creación de una nueva unidad para la orientación a objetos: los puntos objeto. Los diferentes modelos de estimación de costes se encuentran hoy un poco obsoletos y todavía no se han desarrollado completamente otros modelos que recojan los cambios en las herramientas y entornos de desarrollo y la productividad que ha alcanzado la informática sobre todo en la última década. De momento se llevan a cabo algunas iniciativas para actualizar modelos clásicos como COCOMO (Center for Software Engineering, 2000) y otros.

## **2.2 DEFINICIÓN DEL ÁMBITO DEL PROYECTO**

Si usted no sabe a dónde va ningún camino lo conducirá allí. Entonces, ¿cómo se pueden realizar estimaciones en un proyecto cuando no se sabe qué es el proyecto en sí o hasta dónde llega? El sentido común indica que no se puede, pero por otra parte es muy raro encontrar a un cliente que sepa al inicio de un proyecto exactamente cómo debe funcionar su sistema. De hecho casi todos los clientes desean cierta flexibilidad, o lo que en otras palabras significaría que quieren que el sistema haga cualquier cosa que puedan desear en el futuro aunque no tienen ni idea de lo que podría ser. La flexibilidad no es un requisito, pero para el cliente esto no es una opción.

Uno de los problemas más irritantes para los desarrolladores es la adición o cambios de nuevos requisitos por el cliente, o en el peor de los casos por la propia gerencia porque a “alguien” se le ha ocurrido que sería algo “bueno” para el producto tenerlo incorporado.

## **2.3 DEJE QUE LA PERSONA INDICADA REALICE LAS ESTIMACIONES**

Si usted no va a desarrollar el producto no se arriesgue en adivinar el tiempo de desarrollo. Un proyecto está prácticamente condenado cuando la alta gerencia o el departamento de marketing (o cualquier otro) definen las estimaciones de sus sueños, con irresistibles y fabulosos números para que sus clientes se queden asombrados de lo genial que es usted. En el mejor de los casos la consecuencia podría ser que la estimación resulte completamente equivocada, en el peor escenario el equipo de desarrollo se encargará de demostrar que estaban equivocados. La lección es sólo una: deje que el que va a construir el producto realice la estimación. Es un asunto de compromiso y responsabilidad. Deje que esa persona se comprometa y asuma sus responsabilidades.

## **2.4 SER PESIMISTAS PUEDE SER UN BUEN CONSEJO**

El tiempo pasa rápido cuando se está desarrollando y es muy difícil para uno evaluar su propio ritmo y aún más el de las personas que se encuentran a nuestro alrededor. Muchos programadores somos demasiados optimistas y tendemos a olvidar otras partes del proceso de desarrollo como el análisis, el diseño, las pruebas y otras. Estas actividades propias del desarrollo de software y de la gestión de proyectos agregan esfuerzo de trabajo y complejidad al desarrollo y por lo tanto afectan directamente la estimación. Debemos ser muy cuidadosos a la hora de realizar las estimaciones, dejando siempre espacio para estas actividades que también forman parte del desarrollo del producto. Tener un buen colchón de tiempo y ser un poco pesimistas puede prevenir no pocos inconvenientes cuando los plazos se van acortando y el final se va acercando.

## **2.5 SER PESIMISTAS, PERO NO TANTO**

Dale a un programador 5 días para completar una tarea y se tomará los 5 días en completarla. Si el desarrollador se toma 3 días para resolver la tarea empleará el resto del tiempo en hacer otras actividades. El saldo final sólo afectará el plan de proyecto.

Desgraciadamente esta situación es muy frecuente en nuestros proyectos, donde la estimación se convierte en el número mínimo de tiempo de desarrollo y el tiempo de entrega sólo podría ir de mal en peor y nunca al contrario.

Ejercer un poco de presión sobre las personas no les hace daño. Cuando no contamos con registros de productividad que sean fiables es recomendable ejercer un poco de presión sobre el calendario hasta tanto no se aprecie en toda su magnitud lo que nuestro equipo puede dar, de esta forma los llevaremos hasta la frontera de su productividad, que desde luego siempre estará muy por debajo de sus potencialidades reales.

## **2.6 MÁS PERSONAS ES DISTINTO DE UN DESARROLLO MÁS VELOZ**

Un proyecto de 100 días no podría ser completado en 1 día por 100 programadores. Mayor cantidad de personas sólo incrementan la complejidad de un proyecto.

Este clásico dilema quizás proviene del uso de una unidad de medida para el esfuerzo que es el producto de dos, persona-(hora, mes, año) en cualquiera de sus variantes. Digamos que para explicarlo mejor tomaremos la unidad persona-mes. Esta unidad sugiere que debe ser posible intercambiar personas y meses. Esto no es cierto, y no saberlo es uno de los errores en los que puede incurrir cualquier administrador de proyectos. Un proyecto de 15

meses para una persona no puede ser realizado en 1 mes por 15 personas. Sin embargo la matemática así lo sugiere. La matemática también indica otras distribuciones similares como 5 personas y 3 meses y así sucesivamente una lista infinita.

Éste posible error de interpretación es abordado ampliamente por Brooks en su obra, ya histórica, *The Mythical Man-Month*. (Brooks. F, 1975)

## 2.7 ADAPTAR LOS MODELOS AL PROYECTO NO EL PROYECTO A LOS MODELOS

Un modelo de estimación es una abstracción de la realidad, y como tal utiliza fórmulas obtenidas mayormente de forma empírica para predecir el esfuerzo como una función. Los datos que soportan la mayoría de los modelos de estimación proceden de una muestra reducida de proyectos, que en última instancia podrían no tener nada en común con nuestro proyecto. Por tal razón ningún modelo de estimación es apropiado para todos los proyectos y en consecuencia los resultados obtenidos deben ser empleados con mucho juicio.

Los modelos de estimación deben ajustarse a las condiciones de cada proyecto y para ello debemos contar con nuestros propios estándares de productividad, que además sean buenos y dignos de ser utilizados para no depender de lo que dicen los libros y artículos. Estos datos podrían ser obtenidos en proyectos anteriores del mismo tipo y con equipos de desarrollo de características conocidas. Por este motivo es tan importante disponer de la documentación de gestión y de los datos de proyectos informáticos anteriores.

Cuando sea posible los modelos también deben ser evaluados mediante su aplicación a proyectos similares ya completados de forma que se puedan comparar los resultados obtenidos por el modelo con lo real del proyecto.

## 2.8 AJUSTAR LA ESTIMACIÓN

La mayoría de las metodologías de desarrollo de software modernas basan sus ciclos de vida en iteraciones e incrementos. En estos enfoques es posible realizar nuevas estimaciones cuando existe mayor información y modificarlas cuando cambian los requisitos; esto asegura una mayor precisión de la estimación a medida que avanza el desarrollo. Desde luego en muchos casos esto no es aceptable porque los plazos se deben establecer al inicio del desarrollo, aunque una vez que el proyecto incumple sus compromisos de tiempo es casi seguro que deberá volver a estimar nuevamente el tiempo restante para finalizar.

No obstante aunque en su proyecto no sea posible realizar nuevas estimaciones de cara al cliente siempre es una buena idea ajustar su estimación a medida que avanza el desarrollo, de forma que al finalizar el proyecto su modelo de estimación se pueda calibrar con los datos obtenidos por su proyecto.

## 2.9 LAS ESTIMACIONES SE TOMAN DEMASIADO LITERALES

La precisión de la estimación también depende de la habilidad del planificador para traducir la estimación del tamaño en esfuerzo humano, en trabajo y dinero. Por tanto la estimación de un proyecto es tan buena como la estimación del tamaño del trabajo para realizarlo.

Desafortunadamente, es fácil culpar a los programadores en la parte inferior de la pirámide y esto no es un buen presagio para proyectos futuros donde los programadores se negarán a proporcionar nuevas estimaciones o las inflarán dramáticamente.

**Tabla 1: Sobre costos y sobretiempos de implementación de requisitos en proyectos. (The Standish Group, 2013)**

	2004	2006	2008	2010	2012
<b>Tiempo</b>	84%	72%	79%	71%	74%
<b>Costos</b>	56%	47%	54%	46%	59%

En la **Tabla 1** se muestra un indicador muy interesante del conocido (y controversial) Chaos Report del Standish Group (Reporte del Caos). En él se puede apreciar la problemática que aborda el presente trabajo en toda su magnitud a través de cifras preocupantes. Si bien es cierto que existen puntos que han sido criticados en estos informes, como la metodología, la muestra y la forma en la que es aplicada la investigación, no deja de ser trascendente y hay que reconocer que fue uno de los primeros estudios elaborados de forma seria exponiendo la gran crisis del software.

En la tabla se pueden observar las desviaciones en tiempo y costos de un conjunto de proyectos en diferentes años. La cifra más espectacular la ocupa el tiempo, el cual se mantiene sobre un sostenido 70 % en todos los años, esto significando nada más y nada menos que en promedio los proyectos para concluir debieron emplear un 70 % más de tiempo del planificado. Lo mismo ocurre con el costo donde este promedio oscila entre un 46 % en el 2010 a un 59 % en el 2012.

### 3. CONCLUSIONES

Cuando el jefe de proyecto se plantea por primera vez evaluar el trabajo necesario para construir un software determinado sólo dispone de unos pocos datos de lo que ha de ser la aplicación futura y la realidad es que no sabe en absoluto lo suficiente como para llegar a una estimación correcta. Este problema, simplemente, no tiene solución.

¿Por qué es tan importante la estimación? Pues porque no sólo es parte esencial de la gestión de todo proyecto sino que es imprescindible. ¿Acaso a algún arquitecto se le ocurriría comenzar una obra sin conocer el tiempo, el esfuerzo y el costo de su obra? Desde luego que no, y asimismo sucede en la construcción de los sistemas informáticos donde la estimación coloca los cimientos para la planificación y esta a su vez traza la ruta para una ingeniería de software exitosa.

La estimación requiere un conjunto de condiciones como la experiencia, información histórica y sobre todo mucho valor para comprometerse con predicciones cuantitativas cuando la información cualitativa es todo lo que se posee.

Crear estimaciones precisas en los proyectos es algo muy difícil, pero para proyectos de menor envergadura la tarea es mucho más fácil.

### REFERENCIAS

- Brooks, F. (1975). "The Mythical Man-Month". *Addison-Wesley*.
- Pressman, R. (2009) "Ingeniería de Software. Un enfoque práctico". *Editorial Mc Graw Hill*. 2009. p. 690.
- Pressman, R. (2009) "Ingeniería de Software. Un enfoque práctico". *Editorial Mc Graw Hill*. 2009. p. 692.
- The Standish Group. (2013). "Chaos Manifesto". p. 1.
- Center for Software Engineering, (2000). "COCOMO II – Model Definition Manual". Center for Software Engineering.

### ***Authorization and Disclaimer***

*Authors authorize LACCEI to publish the paper in the conference proceedings. Neither LACCEI nor the editors are responsible either for the content or for the implications of what is expressed in the paper.*