

Aplicación para estimar costos en proyectos de software

Andrea Berenice Lencina, Yanina Medina, Gladys Dapozo

Departamento de Informática. Facultad de Ciencias Exactas y Naturales y Agrimensura
Universidad Nacional del Nordeste, Av. Libertad 5450, 3400, Corrientes, Argentina

gaspagu.3@gmail.com, {yanina, gndapozo}@exa.unne.edu.ar

Abstract. La estimación es un factor que influye directamente en la calidad del software porque afecta la planificación de tiempo, recursos y presupuestos económicos en la gestión de proyectos. Una encuesta sobre las prácticas de estimación en las empresas o áreas de sistemas en la ciudad de Corrientes determinó que las técnicas más utilizadas son las que están basadas en el juicio de expertos, sin embargo, surge también que se conocen pero se utilizan poco las técnicas paramétricas que garantizan mayor precisión. Para contribuir con soluciones a esta problemática, en este trabajo se propone una herramienta web para estimar costos, basada en el método UCP CEIS – UFRO, el cual requiere información de los casos de uso, del proyecto y del equipo de trabajo. La aplicación implementa las funcionalidades necesarias para la estimación de costos y almacena datos históricos de los proyectos finalizados, los cuales retroalimentarán el sistema permitiendo mejorar las futuras estimaciones, o permitirán aportar información para implementar técnicas predictivas del campo de la Inteligencia Artificial.

Keywords: Calidad de software, métodos paramétricos, estimación temprana.

1. Introducción

Según [1] “una estimación es una predicción de cuánto tiempo durará o costará un proyecto” y constituye la base para la planificación de los proyectos. El desarrollo del software requiere de la estimación para controlar y administrar los recursos que se necesita utilizar antes y durante el proyecto. Son numerosas las variables, relacionadas con los recursos humanos, el contexto y las políticas que intervienen en el proceso de desarrollo que pueden afectar los resultados finales [2]. La falta de precisión en la estimación puede ocasionar en las empresas de software incumplimiento de plazos, entrega de productos incompletos, aumento del precio final y pérdida de competitividad, especialmente teniendo en cuenta que en la actualidad el software se convirtió en el elemento más caro de la mayoría de los sistemas informáticos [3].

La estimación no es una tarea que se realiza sólo al inicio de un proyecto de software, dado que en ese momento la información que se dispone es muy precaria, y las funcionalidades requeridas seguramente sufrirán modificaciones. El proceso de esti-

mar debe continuar conforme avance el desarrollo para minimizar el nivel de incertidumbre entre lo que ocurre en la realidad y lo estimado. En el Cono de incertidumbre [4], que se muestra en la Fig. 1, se nota claramente que no hay certeza alguna respecto a costo, duración o esfuerzo necesario al inicio del proyecto. Por eso es esencial disponer de una estimación temprana confiable que permita a una empresa analizar si un nuevo proyecto es viable, así como para brindar al cliente un presupuesto inicial lo más preciso posible.

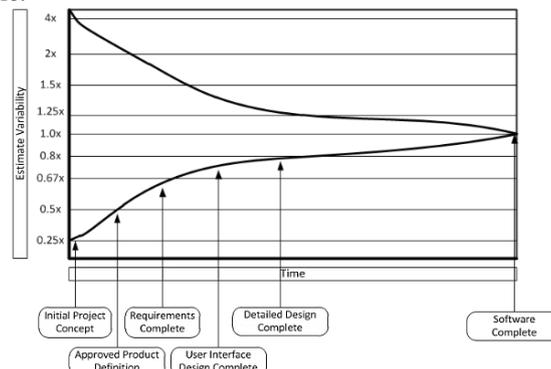


Fig.1.Cono de Incertidumbre de estimación en distintas etapas de un proyecto.

La estimación de costos en los proyectos de desarrollo de software es importante, dado que una estimación imprecisa provocará que un producto software se exceda del presupuesto y no se cumplan los plazos acordados.

Al hablar de costos de un proyecto hay que tener en cuenta [5]: Los costos hardware y software y los costos de esfuerzo. El costo de esfuerzo engloba las remuneraciones de empleados, el mantenimiento de la organización, las redes de comunicaciones, etc. Es por ello que se debe prestar especial atención a este tipo de costos al inicio de la estimación de un proyecto.

En particular, este trabajo se enfoca en la automatización del cálculo de la estimación temprana del costo de desarrollo a partir de la determinación del esfuerzo y productividad basada en el método UCP CEIS – UFRO, mediante una aplicación informática que permitirá obtener el valor estimado del costo, desde el inicio del proyecto, y además almacenar datos históricos como retroalimentación para estimaciones futuras que mejorarán la precisión por el aporte de datos históricos de proyectos concluidos. El artículo se organiza de la siguiente manera: la sección 2 presenta una síntesis de los métodos de estimación, en la sección 3 se presentan los resultados de una encuesta acerca del estado de la estimación en las empresas de software de la ciudad de Corrientes, en la sección 4 la metodología detalla el método UCP CEIS – UFRO y las especificaciones de la aplicación CostEs. En la sección 5 se describe la aplicación CostEs, mostrando los resultados obtenidos en el proyecto “OKE”, aplicación web que integra las funcionalidades que conforman el portal del Gobierno de la Provincia de Corrientes¹ que fue utilizado como caso de estudio para la validación de la herramienta. Finalmente, en la sección 6 se reseñan las conclusiones del trabajo.

¹ <http://www.corrientes.gov.ar/>

2. Métodos de estimación de proyectos de software

Si bien la estimación continúa siendo un problema abierto en el área de la Ingeniería del Software, dada la complejidad de los problemas que aborda, existen diferentes métodos que se han desarrollado para lograr estimar esfuerzo y costos de un proyecto de software.

En [2] se presenta una revisión de los métodos de estimación, en la cual se aprecia que al principio se intentaba que los métodos fueran independientes de las tecnologías y paradigmas de desarrollo, pero, en la actualidad se tiene en cuenta cada vez más, el marco de desarrollo específico, abarcando nuevos enfoques como la programación de aplicaciones móviles, el desarrollo web y la aplicación de técnicas de explotación de información. Los métodos se clasifican en dos categorías:

- Paramétricos o algorítmicos: esto es, una predicción producida por fórmulas matemáticas derivadas de información de proyectos anteriores que han finalizado;
- No paramétricos:
 - basados en juicio de expertos, es decir, según lo que piense un gestor de proyectos teniendo en cuenta su experiencia previa;
 - de aprendizaje automático o “machine learning”, fundados en técnicas de inteligencia artificial, como redes neuronales o árboles de decisión.

Los métodos basados en juicio de expertos presentan el inconveniente de no ser repetibles ni transferibles, lo que significa que si el gestor experto abandona la organización se lleva consigo todo el conocimiento adquirido durante su ejercicio. Sin embargo, es una técnica muy utilizada en las empresas de desarrollo [6].

Los métodos paramétricos o algorítmicos aparecieron en los años 70, siendo COCOMO y COCOMO II de Boehm, y SLIM de Putnam los más reconocidos. Estos requieren diferentes entradas, como cantidad de archivos de datos a mantener, interfaces, consultas, entre otras, las cuales combinadas en una o varias fórmulas permiten obtener estimaciones de distintos aspectos del producto a desarrollar. En principio, fueron desarrollados para estimar proyectos de escritorio, luego fueron evolucionando para adaptarse a otras plataformas. Este es el caso de WebMO que se presenta en [7], un modelo basado en COCOMO II orientado a aplicaciones web, que tiene en cuenta características propias de este tipo de desarrollo. Este método resulta muy adecuado a la hora de estimar, como se muestra en [8], pero como contrapartida no es de estimación temprana, dado que para otorgar estimaciones con mayor precisión requiere de mucha información. En [9] se presenta el método de estimación temprana de costos de productos de software transaccional web UCP CEIS-UFRO², una adaptación del método de UCP de Karner [10]. En [11] se propone un método para la estimación y monitorización de presupuestos en proyectos desarrollados bajo metodologías ágiles, basándose en el tamaño entregado por puntos de historia.

Como una alternativa a las técnicas anteriores, surgieron los métodos basados en inteligencia artificial. Estos, generalmente, utilizan información de proyectos anteriores para el entrenamiento y testeo de los algoritmos, y de esta forma producen el conocimiento necesario para realizar futuras estimaciones. La construcción de métodos

²Centro de Estudios de Ingeniería de Software de la Universidad de la Frontera, Chile.

de estimación de proyectos de software que logren resultados predictivos sobre los recursos a emplear que se ajusten de la mejor manera posible a la realidad obtenible es un problema abierto en el campo de los sistemas de información [12].

En general, los métodos de estimación requieren como información el esfuerzo necesario para desarrollar el producto, que depende del tamaño del software a entregar. Para determinar el tamaño se utilizan medidas directas, como Líneas de Código Fuente (LOC) en el caso de COCOMO, o medidas indirectas, como Puntos de Función (FP), que se determinan a partir de las funcionalidades del software [13]. En [14] se presenta un análisis comparativo entre distintas técnicas paramétricas de cálculo de esfuerzo en proyectos web, en el cual surge que, a mayor especificidad del método, y a mayor disponibilidad y pertinencia de datos históricos de proyectos, mejor es el ajuste a la duración real.

3. Estimación en las empresas

En [6] y [15] se presentan los resultados de una encuesta realizada por el GICS³ a empresas de desarrollo de software y áreas de sistemas de entidades públicas y privadas de la Ciudad de Corrientes, realizada con el objetivo de obtener un panorama actualizado sobre distintos aspectos del desarrollo que afectan la calidad del software, entre estos, formación de los equipos de desarrollo, plataformas y metodologías de desarrollo, técnicas de estimación, ingeniería de requerimientos, gestión de riesgos, gestión cuantitativa y medición, y testing. Sobre los aspectos vinculados con la estimación del software, se obtuvo lo siguiente:

- El 52% de las empresas/áreas realizan alguna estimación para el 100% de los proyectos desarrollados, utilizando en la mayoría de los casos (78%) técnicas convencionales de juicio de experto, por analogía y por consenso.
- Las técnicas paramétricas, como COCOMO, es conocida por el 50% de las organizaciones y utilizadas por el 22%, en tanto que las técnicas basadas en Puntos de Función son conocidas por el 28% pero no son utilizadas en ningún caso.
- Los aspectos de mayor interés para la estimación son la Duración (86%) y el Costo (43%).
- Los datos históricos como fecha de inicio, fin, recursos, costo, son almacenados por aproximadamente el 50% de las organizaciones. Del total de áreas/empresas, solamente el 33% utiliza esta información al momento de realizar una estimación para un nuevo proyecto.

La conclusión general a la que se llega es que las áreas o empresas, en gran medida, desconocen métodos y herramientas que pueden contribuir a mejorar la calidad y la gestión del software que desarrollan. En cuanto al aspecto específico de la estimación, la conclusión que se pudo obtener es que las técnicas paramétricas son poco utilizadas. También es bajo el porcentaje de las áreas/empresas que utilizan datos históricos para la estimación, siendo que estos pueden aportar mayor precisión en la estimación.

³ Grupo de Investigación sobre Calidad del Software de la UNNE.

4. Metodología

4.1. El método UCP CEIS-UFRO

Esta metodología nace de la experiencia del grupo de desarrollo del Centro de Estudios de Ingeniería de Software de la Universidad de la Frontera de Chile y la adaptación de técnicas de estimación temprana. Destacan la ventaja de estimar a partir de los casos de usos porque permiten realizar una estimación temprana sobre un conocimiento cierto de los requerimientos a desarrollar [9]. Fue probado con éxito en proyectos web que superaron las 1000 HH (Horas-Hombre), con equipos de desarrollo de 3-5 personas, tamaño tipo de los proyectos desarrollados en CEIS UFRO, que responde a las características particulares de las pequeñas y medianas empresas de Temuco (Chile).

Este método toma de la técnica UCP la determinación de los actores, de los factores técnicos y de entorno, y las fórmulas para calcular tamaño y esfuerzo. La forma de determinar los puntos de casos de uso sin ajustar y la productividad fueron redefinidas. A continuación se presenta brevemente su metodología:

1. Definir los flujos de trabajo del ciclo de vida del producto y el porcentaje de esfuerzo que le corresponde a cada uno respecto de la totalidad. Uno de esos flujos (en general, la etapa de construcción – implementación) se toma como base para cuantificar la productividad asociada a la implementación de los principales artefactos de software.
2. Determinar la productividad (PROD), esto es, el esfuerzo necesario para completar un UCP (el cociente entre esfuerzo y UCP). Este valor puede ser información disponible en el grupo de trabajo, y se lo toma como tal, o bien se calcula como el promedio de esfuerzo en HH de cada artefacto web que propone cada integrante del grupo de desarrollo.

Esta estimación de esfuerzo por artefacto cubre la etapa asociada al flujo base, por lo que se debe deducir el esfuerzo total teniendo en cuenta el porcentaje de este flujo, mediante regla de tres simple. Luego se toma el mayor de ellos y se hace correspondencia a un UCP (valor de la productividad), y se traducen después los demás valores de esfuerzo en UCP. Los artefactos web son definidos según la arquitectura estructural y características de software que se produce, del tipo de herramientas (framework), plataformas tecnológicas y bibliotecas y componentes reutilizables.

3. Calcular los puntos de casos de uso sin ajustar (UCPcu). Para ello es necesario primero señalar la cantidad de artefactos diferentes que intervienen en cada caso de uso. Posteriormente, se emplea la ecuación 1 para hallar los UCPcu:

$$UCPcu = \sum_{i=1}^n \sum_{j=1}^m CA_{ij} * PA_{ij} \quad (1)$$

Donde CA_{ij} es la cantidad de artefactos del tipo j y PA_{ij} es la correspondencia en UCP para ese artefacto obtenida en el paso anterior, para el caso de uso i .

4. Identificar los actores. Se cuenta la cantidad de actores de cada tipo para calcular los UCPa (puntos de actores), siguiendo la ecuación 2.

$$UCPa = \sum_{i=1}^4 Ac_i * P_i \quad (2)$$

Donde Ac_i es la cantidad de actores del tipo i , y P_i es el peso asociado al tipo.

5. Fijar los Factores de Complejidad Técnica. Están definidos por las influencias técnicas que puedan afectar el proceso de desarrollo del sistema a construir. Cada factor tiene un peso asociado, y el estimador le asigna una ponderación según cómo afecte al producto entre 0 (importancia nula) y 5 (vital importancia). Se deduce, mediante la ecuación 3, el Factor Técnico Total (FTT):

$$FTT = 0,6 + 0,01 \sum_{i=1}^{13} FPi * IFi \quad (3)$$

Donde FPi es el peso que le corresponde al factor técnico i , e IFi es el valor de influencia asignado por el estimador.

6. Precisar los Factores de Entorno. Indican la influencia del grupo humano involucrado en el proyecto sobre el sistema a desarrollar. También tienen un peso asociado y un valor de ponderación entre 0 y 5. El Factor de Entorno Total (FET) se obtiene con la ecuación 4.

$$FET = 1,4 - 0,03 \sum_{i=1}^{10} EFi * IEi \quad (4)$$

Donde EFi es el peso del factor de entorno i , e IEi es el valor de influencia asignado por el estimador.

7. Calcular el tamaño y esfuerzo. Con los valores obtenidos en los pasos anteriores aplicados a las ecuaciones 5 y 6, es posible realizar las estimaciones de tamaño (en UCP) y esfuerzo (en HH).

$$UCP = (UCPcu + UCPa) * FTT * FET \quad (5)$$

$$Esfuerzo (HH) = UCP * PROD \quad (6)$$

8. Estimar costo. El último paso consiste en utilizar la fórmula 7 para conseguir el costo total estimado del producto.

$$CP = (Esfuerzo * CPH) + GA + ME + CF \quad (7)$$

Donde:

CPH: El costo promedio por hora de trabajo de los profesionales del equipo de desarrollo; GA: Gastos adicionales de operación del proyecto; ME: Margen de error, porcentaje promedio estimado a partir de la percepción de completitud del levantamiento de casos de uso, que posibilita la incorporación y gestión de imprevistos del proyecto; CF: Costos fijos, todos los proyectos tienen costos fijos asociados a la infraestructura física, staff de soporte administrativo y de gerencia.

4.2. Aplicación web CostEs

La aplicación CostEs implementa la técnica de estimación UCP CEIS-UFRO y permite administrar proyectos, usuarios y el cálculo del método en sí. En la fig. 2 se presenta el diagrama de casos de uso del sistema.

Este aplicativo está desarrollado en PHP con Symfony® Framework en el backend, y HTML5, MaterializeCSS y JQuery en el frontend.

Los datos post mortem del proyecto “OKE”, aplicación web que integra las funcionalidades que conforman el portal del Gobierno de la Provincia de Corrientes⁴, fueron utilizados para validar la técnica. La empresa encargada del desarrollo brindó toda la información posible y se concertaron reuniones con el líder del proyecto para explicar el funcionamiento del método.

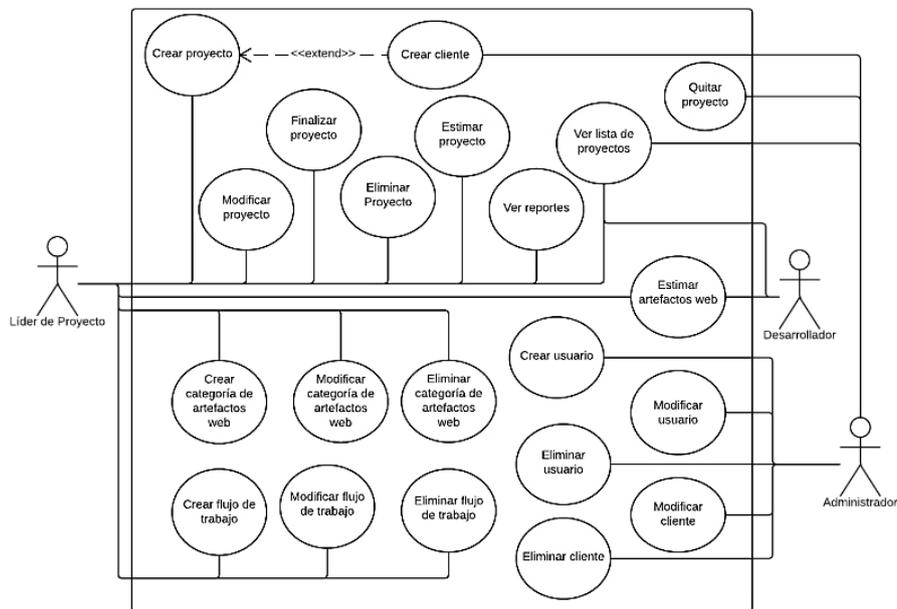


Fig. 2. Diagrama de casos de uso de CostEs

5. Descripción y validación de la herramienta

La herramienta web está disponible en: www.costespfclencina.esy.es. El acceso está restringido a los usuarios registrados, por lo que se deben completar los campos nombre de usuario y contraseña para ingresar. Si las credenciales son correctas, se muestra la pantalla principal del sistema (fig. 3).

⁴ <http://www.corrientes.gov.ar/>

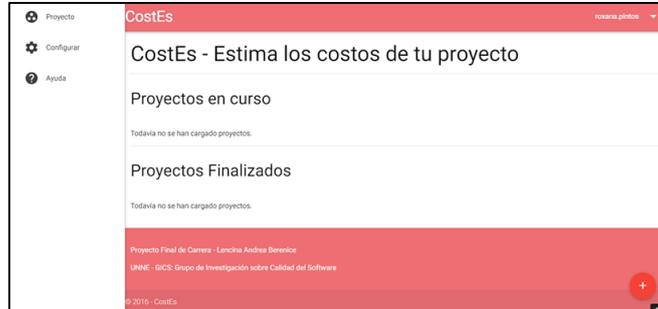


Fig. 3. Pantalla principal de la aplicación

El usuario Líder de Proyectos controla todo el proceso de estimación y utilización de los datos históricos para mejorar estimaciones de proyectos futuros. El paso previo a la estimación de proyectos es configurar los flujos de trabajo. Para ello, hay que seleccionar en el menú lateral la opción Flujos de trabajo. Para el proyecto OKE se han considerado los flujos de trabajo definidos por el método, los cuales son relevamiento inicial (5%), especificación (25%), análisis (15%), arquitectura y diseño técnico (10%), construcción – implementación (30%, flujo base) y testing (15%).

Categorías de artefactos web

La definición de las categorías de artefactos web se debe hacer previa a la estimación de un proyecto, accediendo desde el menú lateral. Las categorías del proyecto OKE fueron pantallas de entrada, pantallas de salida y reportes, gestión de archivos, transacción de datos, excepciones, operación y web services.

Estimación de artefactos web

La sección de estimación de artefactos muestra un listado de todas las estimaciones realizadas por los usuarios, y un botón para agregar una nueva, que posibilita un formulario para la carga de horas estimadas para cada categoría definida.

Creación y estimación de proyectos

En la parte inferior derecha de la página principal se ubica el botón para agregar un proyecto (de igual forma se encuentra esta opción en el menú lateral), lo que abrirá un formulario para completar con los datos generales del proyecto, como ser nombre, descripción, cliente, fecha de inicio, equipo, entre otros.

La sección de estimación de proyectos está dividida en 4 secciones: los actores, los casos de uso, los factores de complejidad y los totales. Las figs. 4 a 6 muestran cada sección con la información correspondiente al proyecto OKE.

Estimar el proyecto Oke

ACTORES
CASOS DE USO

Ingrese la cantidad de actores por cada tipo.

Actores	Cantidad
Simple	<input type="text" value="8"/>
Medios	<input type="text" value="0"/>
Complejos	<input type="text" value="1"/>
Muy complejos	<input type="text" value="2"/>

Fig. 4. Actores de OKE

	ACTORES	CASOS DE USO			FACTORES	TOTALES		
		Pantallas de Entrada	Pantallas de Salida y Reportes	Gestión de Archivos	Transacción de Datos	Excepciones	Operación	WebService, REST
Alta de usuarios	1	1	0	1	3	0	1	
Modificación de u	1	1	0	1	3	0	1	

Fig. 5. Casos de uso (parcial) de OKE

ACTORES	CASOS DE USO	FACTORES	TOTALES
Factores Técnicos		Factores de Entorno	
Seleccione un valor de incidencia de cada factor para el proyecto.		Seleccione un valor de incidencia de cada factor para el equipo.	
Factor	Ponderación	Factor	Ponderación
Sistema Distribuido	<input type="range" value="50"/>	Familiaridad con el Modelo de Proyecto Utilizado	<input type="range" value="50"/>
Objetivos de rendimiento	<input type="range" value="75"/>	Experiencia en la Aplicación	<input type="range" value="75"/>
Eficiencia del usuario final	<input type="range" value="75"/>	Experiencia en la Orientación a Objetos	<input type="range" value="75"/>

Fig. 6. Factores de complejidad (parcial) de OKE

En la fig. 7 se muestra el paso final de la estimación. Allí se deben ingresar los parámetros para el cálculo de costo. Una vez completados los campos se presiona el botón Calcular (celeste) para controlar los valores, y luego el botón Guardar (verde) para almacenar la estimación y ser dirigido a la página de información del proyecto (fig. 8).

ACTORES	CASOS DE USO	FACTORES	TOTALES
Estimación de Costos			
Costo Promedio por Hora (\$)	Gastos Adicionales (\$)	Margen de Error (%)	Costos Fijos (\$)
200	20660	15	40537
<input type="button" value="✓"/> <input type="button" value="🔒"/>			
Totales			

Fig. 7. Totales de OKE

La página de información del proyecto incluye todas las estimaciones realizadas para ese proyecto, tal como se ve en la fig. 8. Los botones inferiores son las acciones que se puede realizar, e incluyen: estimar artefactos, estimar el proyecto, editar el proyecto, finalizar el proyecto, exportar a PDF, y eliminar el proyecto.

Tamaño (UCP)	Esfuerzo (HH)	Costo (\$)
137.77	918.93	272550.9
104.07	1040.66	300548
145.71	1942.73	508024.9

Mostrando 1 a 3 de 3 registros
 < 1 >

Fig. 8. Información de proyecto con estimaciones

Cerrar un proyecto

Una parte muy importante consiste en guardar los datos reales de un proyecto una vez que este haya finalizado, para corroborar el funcionamiento del método. Para dar por terminado un proyecto se debe seleccionar el botón “Finalizar”, ya sea en la pantalla principal o la de información del proyecto.

Reportes y Estadísticas

La sección de reportes se divide en dos partes: generales y por proyecto. Los resultados se presentan gráficos que representan los valores de las duraciones, costos y facturaciones estimadas y reales, por un lado para los proyectos finalizados y por otro para los proyectos en curso. En la fig. 9 se muestra una comparación entre duraciones estimadas y reales.



Fig. 9. Gráfico de duraciones, pestaña “General”

La estimación de esfuerzo obtenida con CostEs con los datos del proyecto OKE tuvo un error del -16% respecto del esfuerzo real, lo cual se mantiene dentro del rango aceptable de $\pm 30\%$ definido por el método. La estimación del costo estimado no pudo ser contrastada con el costo real por ser información reservada de la empresa, pero el líder del proyecto manifestó que era aceptable en el rango mencionado de $\pm 30\%$.

6. Conclusiones y futuros trabajos

Se realizó una aplicación web para estimar costos de desarrollo que implementa la técnica de estimación temprana basada en casos de uso CEIS-UFRO y proporciona valores de tamaño, esfuerzo y costo. La herramienta fue validada con los datos de un proyecto web finalizado, el portal del Gobierno de Corrientes, obteniendo resultados satisfactorios.

Como punto fuerte, CostEs almacena toda la información referida al proyecto y al proceso de estimación, y de esta forma permite ir mejorando las estimaciones futuras. Esta disponibilidad de información histórica permitiría a futuro la aplicación de técnicas de minería de datos.

La aplicación se destaca por poseer un diseño limpio, intuitivo y responsivo, para que sea accesible desde cualquier dispositivo, ventajas que permitirá su incorporación en la gestión de proyectos de desarrollo de software en las pymes de la región NEA.

A futuro se propone validar la herramienta en contextos reales de producción de software, desde el inicio de un proyecto.

Referencias

1. S. McConnell, Software Estimation. Demystifying the Black Art, Redmond, Washington: Microsoft Press, 2006.
2. G. Dapozo, C. Greiner, Y. Medina, M. Ferraro, G. Pedrozo Petrazzini y B. Lencina, «Métodos de estimación de software. Un análisis desde un enfoque evolutivo,» de III Jornadas de Investigación en Ingeniería del NEA y Países Limítrofes, Resistencia, 2014.

3. R. S. Pressman, Ingeniería del Software. Un enfoque práctico, Madrid: McGraw-Hill Interamericana de España, S. A. U., 2002.
4. M. Lacey, «MSDN - Estimación» Microsoft Developer Network, Enero 2012. [En línea]. Available: <https://msdn.microsoft.com/es-es/library/hh765979.aspx>. [Último acceso: 16 Junio 2015].
5. I. Somerville, Ingeniería del software, Madrid: Pearson, 2005.
6. G. Dapozo, Y. Medina y B. Lencina, «La práctica de la estimación en empresas y áreas de Sistemas,» de III Seminário Argentina-Brasil de Tecnologias da Informação e da Comunicação (SABTIC), Uruguaiana, RS. Brasil, 2015.
7. D. J. Riefer, «Web Development: Estimating Quick-to-Market Software» IEEE Software, vol. 17, nº 6, pp. 57-64, 2000.
8. G. N. Dapozo, Y. Medina, A. B. Lencina y O. G. Pedrozo Petrazzini, «Métodos de estimación de esfuerzo y duración en proyectos web pequeños,» Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação, vol. I, nº 2, 2014.
9. J. E. Villegas y G. Robiolo, «Method of Estimating Costs of Software Web Product » SADIO Electronic Journal of Informatics and Operations Research, vol. 14, nº 1, pp. 22-41, 2015.
10. G. Karner, Metrics for objectory, Sweden: University of Linköping, 1993.
11. H. A. Mitre-Hernández, E. Ortega-Martínez y C. Lemus-Olalde, «Estimación y control de costos en métodos ágiles para desarrollo de software: un caso de estudio» Ingeniería Investigación y Tecnología, vol. XV, nº 3, pp. 403-418, 2014.
12. D. Rodríguez, F. Pollo-Cattaneo, P. Britos y R. García-Martínez, «Estimación Empírica de Carga de Trabajo en Proyectos de Explotación de Información,» de Anales del XVI Congreso Argentino de Ciencias de la Computación, Morón, 2010.
13. A. J. Albrecht, «Measuring Application Development Productivity» Proceedings of the Joint SHARE/GUIDE/IBM Application Development Symposium, vol. 10, pp. 83-92, 1979.
14. G. N. Dapozo, Y. Medina, A. B. Lencina y O. G. Pedrozo Petrazzini, «Métodos de estimación de esfuerzo y duración en proyectos web pequeños,» *Revista Eletrônica Argentina-Brasil de Tecnologias da Informação e da Comunicação*, vol. I, nº 2, 2014.
15. G. N. Dapozo, C. L. Greiner, E. Irrazábal, Y. Medina, M. d. I. Á. Ferraro y B. Lencina, «Características del desarrollo de software en la ciudad de Corrientes,» de XII Workshop Ingeniería de Software (WIS). XXI Congreso Argentino de Ciencias de la Computación, Junín, 2015.