

TEAM SOFTWARE PROCESS (TSP): MEJORAS EN LA ESTIMACIÓN, CALIDAD Y PRODUCTIVIDAD DE LOS EQUIPOS EN LA GESTIÓN DEL SOFTWARE

Bayona, S., Calvo Manzano, J., Cuevas, G., San Feliu, T.,

¹ Dpto. Lenguajes y Sistemas Informáticos e Ingeniería del Software. Facultad de Informática

Universidad Politécnica de Madrid Campus de Montegancedo, 28660 Boadilla del Monte, Madrid, España.E-

Mail : [{jcalvo.gcuevas.tsanfe}@fi.upm.es](mailto:E-Mail: sbayona@zipi.fi.upm.es)

Resumen Este artículo presenta los resultados de una experiencia basada en Team Software Process (TSP), los cuales muestran que a medida que los equipos incrementan sus conocimientos y adquieren mayor experiencia en su uso, éstos van mejorando en estimaciones, disminuyen la densidad de los defectos e incrementan la productividad. Los resultados están basados en un análisis de los datos suministrados por 22 equipos de trabajo formados por 5 a 6 estudiantes del cuarto curso de la carrera Informática, en los formularios ajustados a formato TSP. Se recogieron datos de estimaciones de tamaño, esfuerzo, así como de sus valores reales y de los defectos. A partir de estos datos primarios, se ha realizado el análisis de las desviaciones obtenidas al estimar el tamaño a lo largo de los dos ciclos. Se ha analizado también la calidad en términos de defectos/KLOC, así como la planificación en términos de productividad y costes. Finalmente, se presentan las lecciones aprendidas de la utilización de la metodología TSP haciendo hincapié en la calidad del proceso de medida y en el entrenamiento.

Palabras Clave: Estimación de tamaño, defectos, productividad, costes, TSP, trabajo en equipo

1. INTRODUCCIÓN

En los últimos años hemos asistido a un crecimiento exponencial de la demanda de software, que se ha venido aplicando en la resolución de tareas cada vez más complejas y proporcionando cada vez mayor valor añadido [8]. Los productos software siguen entregándose fuera de tiempo, exceden en coste y no cumplen con la calidad esperada por el cliente [9]. En estas circunstancias, se debe hacer la siguiente pregunta: ¿la industria del software está preparada para entregar el software que se necesita en los próximos años, con los niveles de productividad y calidad que se requieren? Según Broadman [2] y Carreira [3] la respuesta es no, razón por la que en los últimos años se están desarrollando una serie de modelos y metodologías orientadas a minimizar la problemática a la que enfrenta la gestión de proyectos.

Para satisfacer los requerimientos que hoy en día requieren las organizaciones, muchos programas de Ciencias de la Computación ofrecen cursos de Ingeniería del software. La educación de los futuros ingenieros de software, que ejercerán su actividad profesional en el ámbito público y/o privado son fundamentales para garantizar el crecimiento de la productividad de la organización.

La experimentación de los estudiantes al formar parte de un proyecto software, en un dominio de problema real, siguiendo un proceso definido, permite que los futuros ingenieros de software puedan realmente estar preparados para generar productos software de calidad.

Un reciente informe realizado por la comisión para la renovación de las metodologías docentes [10], concluye que las universidades deben impulsar las tutorías personalizadas y el trabajo en equipo. Este informe, indica como deberá ser la enseñanza al adaptarse al Espacio Europeo

de Educación Superior (EEES) y mencionan tres aspectos importantes que son (1) la amplia utilización de las nuevas tecnologías en el campo de la enseñanza y (2) practicas con grupos reducidos en torno a casos reales y trabajo en equipo (3) clases de apoyo a grupos reducidos en función de las necesidades del alumno.

Las personas llevan a cabo cada una de las actividades y son quienes pueden llevar al éxito o al fracaso un proyecto. Un objetivo del Team Software Process (TSP) [5] [6] es construir y sostener un entorno de equipos que trabajen de manera cohesionada. Estos equipos deben atender las necesidades del negocio, mejorar la calidad, y reducir los costes y tiempos. El trabajo en equipo es un grupo de personas, cada uno con un rol determinado, trabajando de manera coordinada en la ejecución de un proyecto, que comparten un objetivo común y responden a un resultado final.

A pesar de que existen argumentos frecuentes de que la implementación de métodos rigurosos es muy costosa y conlleva mucho tiempo, se mostrará en este artículo como esto no sucede con los equipos que reciben entrenamiento TSP. En especial se tratará en este artículo, sobre el impacto positivo de este entrenamiento para alcanzar menores costes, una mejor productividad y calidad del producto y reducir la densidad de los defectos.

Así mismo, con el dominio del TSP los equipos se encuentran más preparados para gestionar las necesidades de cambio del cliente, tienen la capacidad para evaluar inmediatamente como los cambios pueden afectar al plan detallado, basándose en la experiencia y las lecciones aprendidas.

Esta experimentación está orientada hacia aspectos de calidad y productividad en la gestión de software con equipos que están trabajando alrededor de un objetivo común y donde cada persona asume un rol específico. Cada equipo está conformado por 5 ó 6 integrantes (Responsable de Desarrollo,

Responsable de Equipo, Responsable de Planificación, Responsable de Proceso, Responsable de Calidad, Responsable de Soporte).

TSP está basado en los principios siguientes:

- Los ingenieros conocen más acerca de su trabajo y pueden hacer mejores planes.
- Cuando los ingenieros planifican su propio trabajo, ellos se comprometen con el plan.
- Para minimizar la duración del proyecto los ingenieros pueden balancear su carga de trabajo.
- Sólo la gente que hace el trabajo, puede recoger datos precisos.
- Para maximizar la productividad, hay que centrarse primero en la calidad.

Este artículo presenta una experiencia sobre la base del entrenamiento, uso del proceso TSP [4] [3] y análisis de los resultados que fueron obtenidos durante el desarrollo de un proyecto. Los datos históricos fueron proporcionados por 22 equipos del cuarto curso de la carrera de Ingeniería Informática de la Universidad Politécnica de Madrid a lo largo de la realización de un proyecto. Todos los equipos recibieron los mismos requisitos funcionales (todos debían de obtener el mismo producto).

El proceso seguido es el siguiente:

- La conformación de los equipos fue realizada de manera libre (se les permitió organizarse libremente entre ellos) y se les entrenó en técnicas de decisión, de factores humanos y definición de roles.
- En el Ciclo I, recibieron entrenamiento en TSP, se les entrega las especificaciones del cliente, se asignan los roles a cada miembro del equipo, se dan las instrucciones del proceso a seguir, los formularios a utilizar y se inicia el proyecto.
- En el Ciclo II, se refuerza el entrenamiento en TSP y se da

entrenamiento en técnicas de revisiones e inspecciones.

Se parte de la hipótesis que los equipos al inicio carecían del conocimiento de TSP y que durante el segundo ciclo ya habían adquirido conocimiento de TSP y experiencia en el desarrollo del proyecto.

Los equipos tuvieron que registrar los datos en los formularios estándar TSP. Se registraron los valores estimados del tamaño y esfuerzo que tendría el producto en el primer y segundo ciclo de desarrollo.

A lo largo del proyecto se recogieron los datos reales de tamaño y esfuerzo permitiendo así la comparación con los valores estimados. En términos de calidad se recogieron los datos de defectos eliminados.

El presente artículo está organizado de la siguiente manera: La sección 2 ilustra el análisis de la desviación de estimación de tamaño del producto, en la sección 3 se hace un análisis de la densidad de defectos, en la sección 4 se ilustra el análisis de la productividad y en la sección 5 se analiza los costes. Para cada una de estas secciones se analiza la tendencia de los equipos y las mejoras obtenidas. Finalmente en la sección 6 se presentan las conclusiones junto con las lecciones aprendidas y trabajos futuros.

2. ERROR DE ESTIMACION DE TAMAÑO

La estimación es una de las actividades prioritarias en la gestión de proyectos, por lo que es importante estimar el tamaño basado en datos históricos, en este caso, solamente fue posible en el Ciclo II cuando ya contaban con los datos históricos del Ciclo I.

La hipótesis a probar en esta sección es: “a través de los diferentes ciclos de desarrollo del proyecto la estimación de tamaño se ajusta gradualmente al tamaño real del producto a realizar”.

La Figura 1 muestra la distribución del Error de Estimación de Tamaño para el Ciclo I y el

Ciclo II del proyecto, cuyos datos se han obtenido mediante la siguiente formula:

$$\text{Error Estimación Tamaño} = 100 * (\text{Tamaño Real} - \text{Tamaño Estimado}) / \text{Tamaño Estimado}$$

2.1 Tendencia de los equipos

Como se observa en la Figura 1, los errores de estimación del Ciclo I son muy pronunciados porque algunos equipos sobreestimaron mientras que otros subestimaron el tamaño a realizar. Mientras que en el Ciclo II se observa que la curva es más estacionaria y cercana a cero es decir una mayor precisión en la estimación. Se observa como los equipos que sobreestimaron adquieren experiencia y mejoran considerablemente en la estimación de tamaño en el Ciclo II.

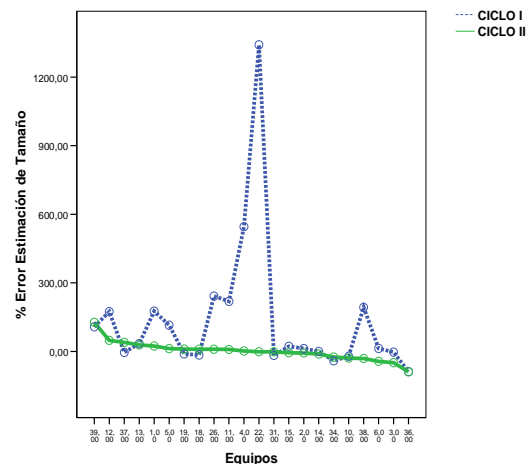


Figura 1. Error de estimación de tamaño de Ciclo I y Ciclo II

En la gráfica se observa los valores extremos que representan el comportamiento anormal: en el Ciclo I dos equipos tuvieron un error de estimación superior al 500% (equipos 4 y 22). En el Ciclo II el valor extremo tuvo un error inferior al 200% (equipo 29).

2.2 Análisis de mejoras en la estimación de tamaño

Los datos mostrados en la Tabla 1 fue generado con el paquete estadístico Statistical Product for Service Solutions (SPSS)[1] usando desviación de estimación de tamaño, proporcionados por cada uno de los equipos.

	Mín	p(25)	Media	p(75)	Max
Error Estimación CICLO I	-88,1	-12,0	136,1	180,5	1341,6
Error Estimación CICLO II	-89,3	-25,61	1,3	15,4	127,7

Tabla 1. Estadísticos descriptivos de Estimación

Se utiliza el análisis de percentiles como una herramienta estadística para determinar la dispersión de los datos. El rango de percentiles del 25% al 75% del Ciclo I es del 192,56% de error en la estimación de tamaño, y este rango se reduce a 41% para el Ciclo II. El valor de la media en el Ciclo I es 136,1% y en el Ciclo II se reduce hasta 1,3%.

Esto es debido a que en el Ciclo II:

- Los equipos ya cuentan con experiencia y conocimiento de técnicas de estimación,
- Cuentan con experiencia en el producto,
- Cuentan con los datos históricos del Ciclo I.

De estos datos podemos deducir que a medida que los equipos van adquiriendo mayor experiencia mejoran la estimación a través de los ciclos.

Esta experiencia y los conocimientos adquiridos influirán en los demás aspectos como densidad de defectos, productividad y costes que se detallan a continuación.

3. DENSIDAD DE DEFECTOS

Para producir programas libres de defectos es importante medir la calidad de los programas, y con esta información dar los

pasos para mejorar. Introducir defectos ocasionalmente en los programas es algo innato en los seres humanos. El reto es llegar a gestionarlos para reducirlos cada vez más alcanzando el valor lo más próximo a cero.

Un defecto es cualquier cosa que reduce la capacidad del software para cumplir de manera completa y eficiente la necesidad de los usuarios. El defecto es algo objetivo que se puede identificar, describir y contabilizar.

La construcción de software es un proceso que se puede evaluar a lo largo de las etapas de los productos intermedios. Una alta densidad de defectos en los productos intermedios se traduce en un esfuerzo añadido a causa del trabajo que hay que volver a realizar o el tiempo incurrido en solucionar defectos. Sin embargo, esta situación puede mejorar si se lleva un registro de los defectos que se están produciendo y si se identifican los factores que los ocasionan.

En esta sección se estudia la densidad de defectos producida por los equipos (número de defectos eliminados por cada 1000 líneas de código).

La hipótesis investigada en esta sección es: “a medida que los equipos avanzan a través de los ciclos, el número de defectos inyectados, y por lo tanto eliminados, por cada 1000 líneas de código (KLOC) disminuye”.

La densidad de defectos es un indicador temprano de la calidad final y se define mediante la siguiente fórmula:

$$\text{Densidad defectos} = \text{Defectos eliminados} / \text{KLOC}$$

3.1 Tendencia de los equipos

La Figura 2 muestra la distribución de densidad de defectos para el Ciclo I y para el Ciclo II del proyecto con los datos de defectos inyectados y eliminados por cada 1000 líneas de código generados por los 22 equipos.

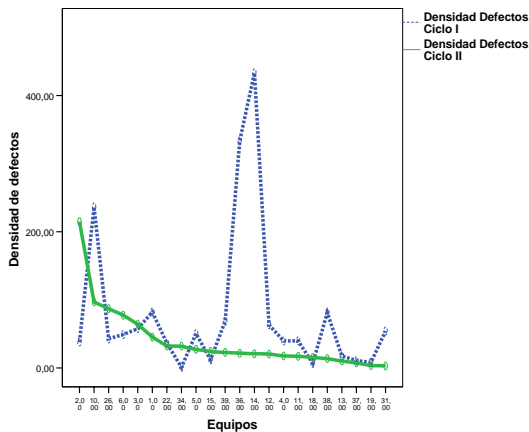


Figura 2. Densidad de defectos para Ciclo I y Ciclo II.

En muchos ambientes de desarrollo, el registrar los defectos tiene sólo un objetivo de medir la calidad del código. Sin embargo se debe tener en consideración que los procesos de revisión e inspección, y las pruebas pueden también verse afectados por la experiencia y calidad de los revisores.

En la gráfica se observa que en la mayoría de los equipos la densidad de defectos se reduce en el Ciclo II, con una tendencia cercana al cero. Además, se observa los valores extremos que representan el comportamiento anormal: en el Ciclo I tres equipos tuvieron 236 defectos/KLOC (equipo 10), 333 defectos/KLOC (equipo 36) y 433 defectos/KLOC (equipo 14) respectivamente. En el Ciclo II el valor extremo se reduce a 214 defectos/KLOC (equipo 2).

3.2 Análisis de mejoras en la densidad de defectos

Los datos mostrados en la Tabla 2 son los percentiles que fueron generados con SPSS, usando la variable cantidad de defectos eliminados y tamaño de producto real, proporcionados por cada uno de los equipos.

	Mín	p(25)	Media	p(75)	Max
Densidad de defectos CICLO I	1,0	16,2	80,0	71,6	433,3
Densidad de Defectos CICLO II	2,9	14,9	40,0	50,0	214,7

Tabla 2. Estadísticos descriptivos de la Densidad de Defectos

Del análisis de percentiles, se deduce que el rango del percentil 25% al 75% del Ciclo I es 55 defectos/KLOC y el rango obtenido en el Ciclo II es de 35 defectos/KLOC. La media de la densidad de defectos en el Ciclo I fue de 80 defectos/KLOC y en el Ciclo II esta densidad se reduce a 40 defectos/KLOC. Así mismo analizando la desviación típica como una medida de dispersión, se observa que en el Ciclo I el valor es muy alto (110,3) mientras que en el Ciclo II baja a un valor cercano a la mitad (47,2).

La variación del Ciclo I al Ciclo II fue debido a la introducción de técnicas de revisión e inspección.

De los datos anteriores podemos deducir que a medida que los equipos adquirieron mayor experiencia, así como la introducción de revisiones de diseño y código en el Ciclo II se produjo una disminución de los defectos inyectados, con lo que se obtuvo un producto de mayor calidad.

La reducción de la densidad de defectos total se traduce directamente en una reducción en la cantidad de trabajo que hay que rehacer y minimización de posibles retrasos y rechazo de parte del cliente. Un factor importante es que en el Ciclo II los miembros del equipo tienen la experiencia del Ciclo I y han aprendido de sus errores. 1000 líneas de código generados por los 22

4. PRODUCTIVIDAD

Una métrica importante en toda organización es el índice de productividad, es decir, la generación de producto por unidad de tiempo.

La hipótesis a probar en esta sección es: “a medida que se avanza en los ciclos de desarrollo del proyecto, la productividad se incrementa”.

La productividad está definida por la relación que existe entre la cantidad de líneas de código generadas por unidad de tiempo o esfuerzo realizado.

$$\text{Productividad} = \text{LOC}/\text{Horas}$$

4.1 Tendencia de los equipos

Para constatar la hipótesis planteada se tomaron los datos de 22 equipos.

La Figura 3 muestra la distribución de la productividad de los equipos, observándose una mejora en términos medios.

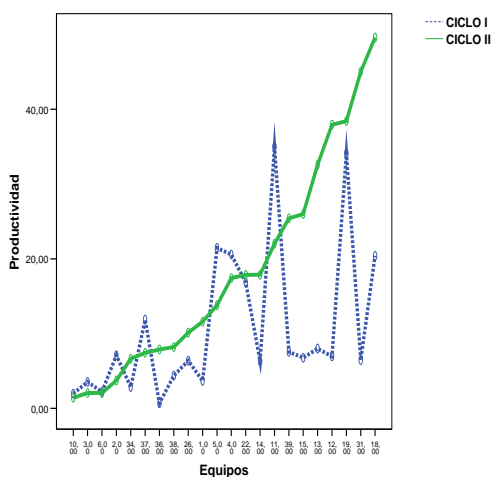


Figura 3. Productividad de los equipos por ciclo.

En la gráfica se observa en el Ciclo II, la tendencia de crecimiento de la productividad. En el Ciclo I se alcanzó un máximo de 35 LOC/hora (equipo 11) mientras que en el Ciclo II se incrementa a 49 LOC/hora (equipo 18) de productividad. El valor mínimo se incrementa de 0,6 LOC/hora a 1,4 LOC/hora.

4.2 Análisis de mejora en la productividad

Los datos mostrados en la Tabla 3 son los percentiles que fueron generados con SPSS, usando la variable tamaño real del producto y esfuerzo real, proporcionados por cada uno de los equipos.

	Mín	p(25)	Media	p(75)	Max
Productividad CICLO I	0,6	3,6	10,6	17,7	35,0
Productividad CICLO II	1,4	7,2	18,4	27,6	49,0

Tabla 3. Estadísticos descriptivos de Productividad

El análisis de percentiles muestra que en el rango del 25%, en el Ciclo I se ha pasado de 3,6 LOC/hora a 7,2 LOC/hora en el Ciclo II. En el rango del 75% se ha pasado de 17,7 LOC/hora en el Ciclo I a 27,6 LOC/hora en el Ciclo II.

El valor de la media en el Ciclo I es 10,6 LOC/hora y en el Ciclo II aumenta a 18,4 LOC/hora.

Este incremento se explica porque:

- En el Ciclo I el equipo de trabajo dedicará más horas de trabajo a la organización y planificación del proyecto, así como al conocimiento del dominio del problema a resolver.
- En el Ciclo I la productividad está condicionada al entrenamiento, los lenguajes de programación usados y la plataforma tecnológica.
- En el Ciclo II el equipo se va consolidando con una visión compartida para el logro de las metas y objetivos.

Con TSP se enfatiza en las fases tempranas del proyecto, dedicando más tiempo a los requerimientos, diseño de alto nivel, y revisando e inspeccionando el diseño. Así mismo el equipo se va consolidando con una visión compartida para el logro de las metas y objetivos.

5. COSTES

La relación coste y productividad es otra inquietud en los proyectos software y se debe responder a la pregunta si están teniendo alta productividad pero a costes elevados.

Entonces, es importante medir si realmente el indicador de productividad alto permite reducir costes.

La hipótesis a probar es: “*a medida que los proyectos van mejorando a través de los ciclos, no sólo la productividad se incrementa, sino que se reducen significativamente los costes*”.

En este caso se utiliza la relación que existe entre los costes del proyecto por líneas de código generadas y se define mediante la siguiente formula:

$$\text{Productividad (Euros/LOC)} = \text{Coste proyecto} / \text{LOC}$$

$$\text{Coste proyecto} = X \text{ Euros} * \text{hora de trabajo}$$

5.1 Tendencia de los equipos

Para constatar la hipótesis planteada se tomaron los datos de 22 equipos.

La Figura 4 muestra como comparativa de costes entre el Ciclo I y el Ciclo II,

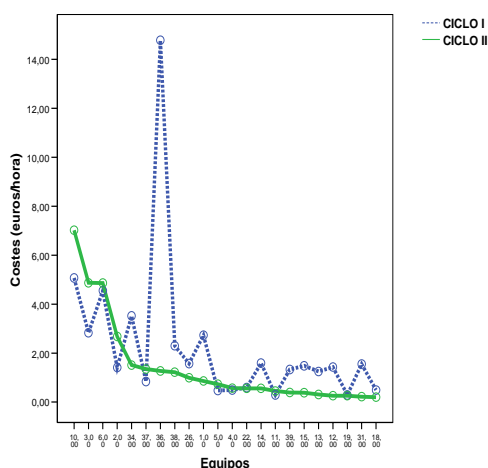


Figura 4. Comparativa de Coste por línea de código

En la gráfica se observa en el Ciclo II, la tendencia de reducción de costes. Se han disminuido los máximos de 14,7 (equipo 36) a 7 euros/LOC (equipo 10).

5.2 Análisis de mejora en los costes

Los datos mostrados en la Tabla 4 son los percentiles que fueron generados con SPSS, con los datos proporcionados por cada uno de los equipos.

	Mín	p(25)	Media	p(75)	Max
Costes CICLO I	0,2	0,5	2,3	2,7	14,7
Costes CICLO II	0,2	0,3	1,4	1,3	7,0

Tabla 4. Estadísticos descriptivos de Costes

En el rango entre percentil del 25% al 75% del Ciclo I, el coste es de 2,2 Euros/LOC y, en este mismo rango, en el Ciclo II se reduce a 1 euro/LOC. El percentil 25% del Ciclo I es de 0,5 Euros/LOC y se reduce en el Ciclo II a 0,3 Euros/LOC.

El percentil 75% del Ciclo I es de 2,7 Euros/LOC y se reduce en el Ciclo II a 1,3 Euros/LOC. La media en el Ciclo I es 2,3 Euros/LOC y se reduce en el Ciclo II a 1,4 euros/LOC.

Esto se explica por que la experiencia del Ciclo I junto con el entrenamiento en TSP de los equipos ha hecho de que en el Ciclo II:

- Hayan incrementado su productividad.
- Se haya logrado reducir costes.

c) Las técnicas de pruebas mejoran y por lo tanto reducen tiempos sin descuidar la calidad.

Es decir a más entrenamiento se incrementa la productividad y se reducen costes.

6. CONCLUSIONES Y FUTUROS TRABAJOS

Las hipótesis planteadas en este experimento han sido comprobadas y en la Tabla 5 se presentan los resultados. Se observa que el

error de estimación de tamaño disminuye, la densidad de defectos disminuye, la productividad se incrementa y los costes disminuyen con el uso y experiencia en TSP.

METRICA	Res ult ado	CICLO I			CICLO II		
		Med	Mín	Máx	Med	Mín	Máx
Error Estimación Tamaño (%)	-	136, 1	-88,1	1341,6	1,3	-89,3	127,7
Densidad defectos (defectos/KLOC)	-	80,0	1,0	433,3	40,0	2,9	214,7
Productividad (LOC/hora)	+	10,6	0,6	35,0	18,4	1,4	49,0
Costes (Euros/LOC)	-	2,3	,2	14,7	1,4	,2	7,0

Tabla 5. Cuadro comparativo entre Ciclo I yII

El rango de error de estimación de tamaño disminuye con el avance del proyecto. En el Ciclo I el rango se sitúa entre -88,0% y 1341,6% con una media de 136,1%. En el Ciclo II el rango se sitúa entre -89,3% y 127,7% con una media de 1,3%.

La densidad de defectos disminuye con la introducción de técnicas de inspección y revisión. En el ciclo I, donde no se llevaron a cabo inspecciones ni revisiones, el rango se sitúa entre 1,0 defectos/KLOC y 433,3 defectos/KLOC. En el Ciclo II el rango se sitúa entre 2,9 defectos/KLOC y 214,7 defectos/KLOC con una media de 40 defectos/KLOC.

La productividad en el Ciclo II se incrementa porque los miembros del equipo han adquirido experiencia y conocimientos. En el ciclo I el rango se sitúa entre 0,6 LOC/hora y 35 LOC/hora. En el Ciclo II el rango se sitúa entre 1,4 LOC/hora y 49 LOC/hora con una media de 18,4 LOC/hora. En consecuencia los costes bajan de una media de 2,3 Euros/LOC a 1,4 Euros/LOC.

Podemos concluir como lecciones aprendidas que:

- El uso de un proceso definido y medido disciplinado hace que sean posibles mejoras en estimación, densidad de defectos, productividad y, en consecuencia, reducción de costes.

- Es importante definir, coleccionar y analizar de manera rigurosa las métricas porque permiten conocer cómo se evoluciona en el ciclo de producción, sí se está mejorando la calidad y productividad, y dónde conviene actuar para mejorar.

- El entrenamiento, las habilidades y los conocimientos necesarios para desarrollar las actividades encomendadas son factores positivos en términos de productividad y calidad. Por tanto, invertir en entrenamiento del personal es un elemento esencial para la mejora.

- Los equipos al usar técnicas TSP reducen los defectos, incrementan la productividad y disminuyen los errores de estimación, a través de los ciclos, elemento que refuerza la información de otras universidades.

- Los equipos en TSP evolucionan desde una visión orientada al producto en el Ciclo I, hacia una visión orientada a la mejora del proceso en el Ciclo II.

- El introducir mejoras en los procesos que ya están definidos y medidos puede ser rápido y poco costoso como ha demostrado la introducción de inspecciones y revisiones.

Como futuros trabajos dentro de esta área se propone seguir estudiando:

- Los factores que influyen en la productividad tales como lenguajes y entornos de desarrollo empleados.
- La estimación de tamaños por fases y su impacto en la productividad y calidad del producto.

6. AGRADECIMIENTOS

Este artículo esta patrocinado por las empresas ENDESA, DMR Consulting Foundation y SUN Microsystems a través de la “Cátedra de Mejora del Proceso Software en el Espacio Iberoamericano”

7. REFERENCIAS

- [1] Alvarez Santos, Carmen (1994), Manual de SPSS: Estadística descriptiva.
- [2] Brodman, J. & Johnson, D. "Project Planning: Disaster Insurance for Small Software Projects. LOGOS International, Inc. Proceedings from SEPG 2000: Ways to Make Better Software. 20-23 March 1999. Seattle, Washington.
- [3] Carreira, M. & Román, I. "Estimación del Coste de la Calidad del Software a través de la Simulación del Proceso de Desarrollo". Revista Colombiana de Computación, 2 (1) pp. 75-87. 2002.
- [4] Donald R, McAndrews, "The Team Software ProcessSM (TSP):An Overview and Preliminary Results of Using Disciplined Practices", November 2000, SEI Technical Report CMU/SEI-2000-TR-015, ESC-TR-2000-015.
- [5] Humphrey, W. "TSP: Leading a Development Team": Addison-Wesley Publishing Company, Inc., 2005.
- [6] Humphrey, W. "Introduction to the Team Software Process": Addison-Wesley Publishing Company, Inc., 1999.
- [7] Noopur Davis, Julia Mullaney, "The Team Software ProcessSM (TSP) in Practice: A Summary of Recent Results", September 2003, SEI Technical Report CMU/SEI-2003-TR-014.
- [8] Pressman, R.S. Software Engineering: A Practitioner's Approach. 5th Edition - European Adaptation. McGraw Hill, 2004.
- [9] Sommerville, I. and Sawyer, P. (1997), Requirements Engineering: A good practice guide. England: John Wiley & Sons.
- [10] http://www.mec.es/educa/ccuniv/html/metodologias/docu/PROPUESTA_RENOVACION.pdf, 2006.