

Apostamos a la calidad

Uruguay es un fuerte exportador de software. Para mantenernos competitivos es necesario apostar a la calidad. En breve, marcará la diferencia.

En los dos primeros artículos de esta serie presentamos la importancia de la calidad de software y resultados empíricos obtenidos con el uso del Personal Software Process (PSP) y al Team Software Process (TSP). Este tercer artículo presenta el PSP basándose en el reporte técnico "The Personal Software Process" de Humphrey [1].

La creación del PSP surge de una inquietud de Humphrey. En uno de sus libros comenta lo siguiente: "Aunque el Capability Maturity Model (CMM) [luego CMMI] fue y continúa siendo altamente exitoso, pronto vi problemas. [...] Me di cuenta de que hasta que no cambiáramos las prácticas de los propios profesionales, nunca se podría alcanzar una verdadera capacidad de proceso en la ingeniería de software ." [2] Entonces, Humphrey crea el PSP teniendo como objetivo fundamental mejorar la calidad del trabajo individual de cada ingeniero; objetivo que logró con creces.

Principios del PSP

El PSP está basado en los siguientes principios:

- Cada ingeniero es diferente; para ser más efectivos los ingenieros deben planificar su trabajo y deben basar sus planes en sus propios datos personales.
- Los ingenieros deben usar procesos bien definidos y medidos, de forma de mejorar consistentemente su rendimiento.
- Para lograr desarrollar productos de calidad los ingenieros deben sentirse personalmente responsables de la calidad de sus productos. Los productos superiores no son producidos por casualidad, los ingenieros deben esforzarse para realizar trabajos de calidad.
- Cuesta menos encontrar y corregir defectos tempranamente que tardíamente.
- Es más eficiente prevenir los defectos que encontrarlos y corregirlos,
- La forma correcta termina siendo siempre la forma más rápida y barata de realizar un trabajo.

La estructura del PSP

El PSP es un proceso de software para el individuo, y es, como dice su nombre, personal. El proceso comienza en la fase de planificación, que tiene como entrada los requerimientos del módulo de software a desarrollar, y culmina con la fase de post-mortem (luego de finalizadas las pruebas unitarias). En la Figura 1 se presenta la estructura del proceso.

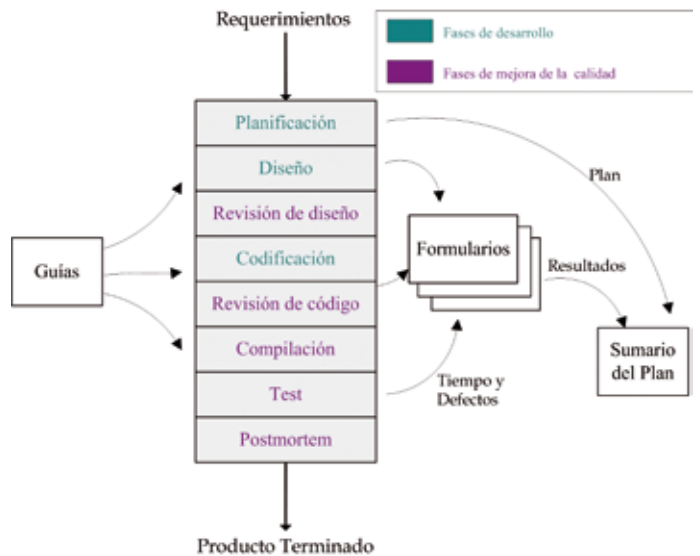


Figura 1 – Proceso PSP

El PSP se divide en fases, cada una de ellas tiene un propósito específico y una guía (script) que indica los pasos que se deben realizar al ejecutarla. Las fases son: planificación, diseño, revisión de diseño, codificación, revisión de código, compilación, test (pruebas unitarias) y postmortem.

El PSP también provee formularios donde se ingresan (o registran) tanto los datos de la planificación como los datos de lo que realmente sucede durante el desarrollo; como por ejemplo el tiempo planificado y el tiempo realmente utilizado en cada fase.

Durante la fase de postmortem se registran datos finales (como por ejemplo el tamaño del módulo que se ha construido) y se analizan los datos recolectados durante todo el proceso.

Estimación de tamaño y esfuerzo

En la fase de planificación se realiza un diseño conceptual basado en los requerimientos del módulo a desarrollar. El diseño conceptual se utiliza para estimar el tamaño del producto, y a partir de este se estiman los recursos (esfuerzo) y el cronograma del desarrollo. El proceso de planificación utilizado en el PSP se presenta en la Figura 2.



Figura 2 – Estimación con PROBE

Para planificar, el PSP utiliza el método PROxy Based Estimating (PROBE). Este método utiliza datos históricos de tamaño de módulos para obtener el tamaño estimado del módulo que se va a desarrollar. Para obtener el tamaño estimado se utiliza el modelo conceptual y el método de regresión lineal.

Una vez que se tiene el tamaño estimado del módulo, se estima el tiempo necesario para desarrollarlo (estimación de esfuerzo a nivel personal).

Para esto se utilizan datos históricos de la productividad personal (tamaño/tiempo), y nuevamente se usa el método de regresión lineal sobre estos datos.

Como última etapa en la planificación se arma el cronograma de desarrollo. Para esto basta con considerar el tiempo disponible en cada semana y el tiempo estimado de desarrollo. Toda la planificación es soportada por herramientas de software que hacen liviana la tarea.

Al culminar la planificación en PSP se tiene una estimación del tamaño del producto, del tiempo total de desarrollo, del tiempo requerido en cada fase y del cronograma para realizar el proyecto. Además, utilizando datos históricos se estima, de forma automática, la cantidad de defectos que se van a inyectar y a remover durante el desarrollo.

Los errores de estimación en PSP, tanto en tamaño como en esfuerzo, fueron presentados en la edición anterior, disponible en www.1024.com.uy. En el mismo se mostró la increíble mejora que logran los ingenieros en sus estimaciones a medida que transcurren los ejercicios del curso de PSP.

Recolección de datos en PSP

Durante la ejecución del proceso el desarrollador recolecta diversos datos. Estos datos sirven para realizar el seguimiento personal del proceso y para realizar mejores estimaciones en futuros proyectos. Para esto, el PSP provee distintos formularios; los mismos se encuentran en la herramienta de soporte al proceso.

Los principales registros durante el desarrollo son los de tiempo y defectos. Para cada fase se registra el tiempo total dedicado a la fase. Esto se puede hacer simplemente iniciando y deteniendo un cronómetro provisto por la herramienta de soporte.

También se registra cada uno de los defectos que se detectan en las distintas fases. Para cada uno se ingresan pocos datos, entre ellos la descripción del defecto.

Estas descripciones sirven luego para mejorar las revisiones de diseño y de código de futuros proyectos.

Al finalizar el desarrollo del módulo se contará con el tamaño real del mismo. Este tamaño será registrado y servirá de base para tener mejores estimaciones en un futuro.

Gestión de la calidad

Uno de los principios del PSP, y fundamental en cuanto a la calidad, es que los ingenieros deben ser responsables de la calidad de los módulos que producen. Para lograr el compromiso del ingeniero el PSP propone que sea este quien planifique y gestione su propio trabajo. Para lograr productos de calidad el PSP se basa no solo en remover los defectos de forma temprana sino también en prevenir la inyección de los mismos.

Para remover tempranamente los defectos se proponen las fases de revisión de diseño y de revisión de código. Conociendo los defectos en los que incurre, el ingeniero crea checklists a partir de sus propios defectos, logrando con esto realizar revisiones de diseño y de código de forma eficiente y efectiva. De esta manera se llega a la fase de test con menos defectos y será mucho menos costosa la ejecución de dicha fase.

Para prevenir la inyección de defectos el PSP propone dos formas. Ocurre naturalmente que al estar registrando continuamente los defectos en los que uno incurre estos se comiencen a evitar. Como segunda forma, el PSP propone realizar el diseño del módulo de forma completa; para esto el PSP provee un método y una notación específica. Cuando se elabora un diseño completo, la fase de codificación del PSP requiere menos dedicación de tiempo. Empíricamente se ha probado que la inyección de defectos por hora es mucho mayor en la fase de codificación que en diseño. Entonces, trasladando tiempo de desarrollo desde la codificación hacia el diseño se previenen defectos y se obtienen productos de mejor calidad a menor costo.

PSP y TSP

“Uno de los problemas principales en cualquier campo de la ingeniería es lograr que los ingenieros usen consistentemente los métodos que les han sido enseñados. En el PSP estos métodos son: seguir un proceso definido, planificar el trabajo, recolectar datos y usar esos datos para analizar y mejorar el proceso. Si bien esto parece simple desde la teoría, no lo es en la práctica. Debido a este problema, el foco principal al momento de introducir el PSP en una organización es el proveer un ambiente de trabajo adecuado que soporte las prácticas del mismo. Esta es la razón que tuvo el SEI para crear el Team Software Process.” [1] En el próximo número presentaremos brevemente el TSP.

Conclusiones

La calidad es la prioridad más importante en el desarrollo de software. Mejorar los procesos permite mejorar la calidad del producto. Mejorar la calidad permite controlar mejor los procesos (ya que estos no dependen del *testing* de software y por lo tanto son más predecibles), reducir costos y acortar cronogramas. El software de calidad es desarrollado por personas disciplinadas y motivadas. El PSP y el TSP brindan el marco necesario para esto.

La estrategia principal que utiliza el PSP para construir software de calidad es tanto la prevención de defectos como la eliminación de defectos de forma temprana. Para esto propone: un proceso definido, realizar diseños completos, revisar el diseño, revisar el código y gestionar la calidad durante todo el proceso mediante medidas objetivas. Los resultados del uso del proceso están documentados y son extraordinarios.

Uruguay en este momento es un fuerte exportador de software. Para lograr mantener esta posición hay que ser competitivo. En este sentido, la apuesta a la calidad marcará la diferencia. Los países de la región se mueven rápido y sobre alguno de ellos en breve nos preguntaremos: ¿Cómo lo están haciendo? [3, 4]

Bibliografía

1. Humphrey, Watts S. The Personal Software Process. Software Engineering Institute, 2000. CMU/SEI-2000-TR-022.
2. Humphrey, Watts S. Winning with Software: An Executive Strategy. Addison-Wesley Professional, 2001.
3. Humphrey, Watts S. y Carleton, Anita. How Mexico is Doing it. News@SEI, <http://www.sei.cmu.edu/library/abstracts/news-at-sei/wattsnew.cfm>, August 2008.
4. Nichols, William y Salazar Rafael. Deploying TSP on a National Scale: An Experience Report from Pilot Projects in Mexico. Software Engineering Institute, 2009. CMU/SEI-2009-TR-011.