**INSTITUTO TECNOLÓGICO SUPERIOR YAVIRAC**



**NOMBRE**: López Christian

**ASIGNATURA**: [Procesos](http://eva.yavirac.edu.ec/mod/forum/view.php?id=18480) de Desarrollo del Software

**TEMA**: Personal Software Process(psp) y Scrum

**FECHA**:05/08/2020

INTRODUCCIÓN

PSP como metodología de trabajo para el desarrollo en el cual se busca tener un proceso disciplinado para el desarrollar además permite resultados de calidad. Enfocada principalmente en el programador, enfocándolo en la planeación, diseño, estándares y revisiones al detalle de lo que va realizando, registrando todo aquello en formas o plantillas que han sido diseñadas por Humphrey. La metodología maneja un conjunto de scripts que especifican los requerimientos de entrada, el proceso que debe de seguirse y los resultados esperados. enfocado a cada fase de la metodología

**DESARROLLO:**

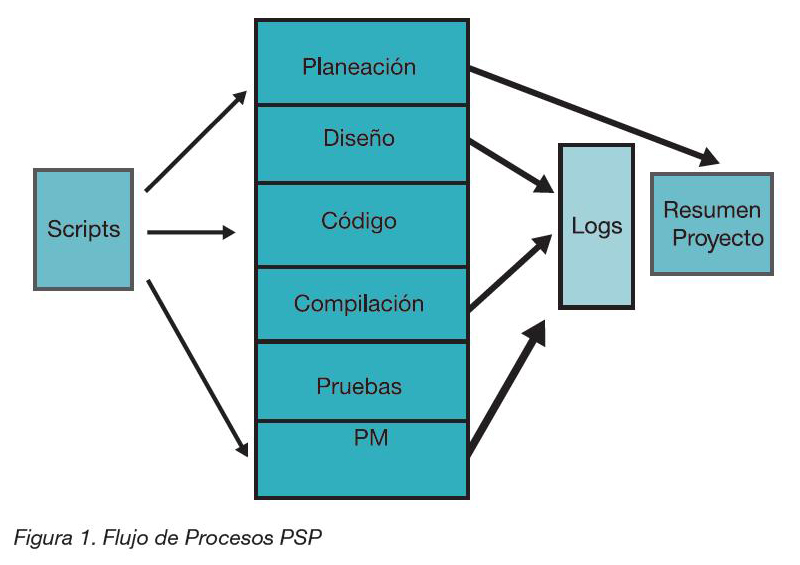
**PSP**

El proceso personal de software, PSP, es un conjunto de prácticas disciplinadas para la gestión del tiempo y mejora de la productividad personal de los programadores o ingenieros de software, en tareas de desarrollo y mantenimiento de sistemas, mediante el seguimiento del desempeño predicho frente al desempeño real. Está alineado y diseñado para emplearse en organizaciones con modelos de procesos [CMMI](https://es.wikipedia.org/wiki/CMMI) o [ISO 15504](https://es.wikipedia.org/wiki/ISO_15504). Fue propuesto por [Watts Humphrey](https://es.wikipedia.org/wiki/Watts_Humphrey) en [1995](https://es.wikipedia.org/wiki/1995) y estaba dirigido a estudiantes. A partir de [1997](https://es.wikipedia.org/wiki/1997) con el lanzamiento del libro "An introduction to the Personal Software Process" se dirige ahora a ingenieros juniors.

Con PSP los ingenieros de software pueden adquirir las habilidades necesarias para trabajar en un proceso de software en equipo [TSP](https://es.wikipedia.org/wiki/Team_Software_Process).

Se puede considerar como la guía de trabajo personal para ingenieros de software en organizaciones que emplean un modelo CMMI con nivel de madurez o de capacidad de procesos que implica la medición cualitativa y mejora de procesos.

Uno de los mayores problemas que tiene es la gran cantidad de datos que hay que tomar. El PSP tiene obsesión por la toma de datos y elaboración de tablas. El PSP se orienta el conjunto de áreas clave del proceso que debe manejar un desarrollador cuando trabaja de forma individual.



Esta metodología está apoyada por una herramienta computacional desarrollada por el Instituto de Carnegie Mellon, que genera una serie de registros con información valiosa para llevar a cabo la siguiente planeación  
y para actualizar el plan al terminar cada programa de software.  
  
Cuando se termina el curso de PSP y se empieza a realizar su primer proyecto siguiendo la nueva metodología, uno mismo se da cuenta del tiempo que a veces invierte incorrectamente cuando quiere dirigirse directamente a codificar el programa, en lugar de haber realizado un análisis y un diseño general de la solución. Al ir avanzando en el desarrollo de los diferentes proyectos uno se acostumbra a definir cuántas líneas de código se programan, esto basado en un diseño conceptual y en aquel código que uno ha desarrollado, cuántas líneas pueden servir como base, que tanto código podemos reutilizar e incluso cuantas líneas modificarás o borrarás del código base que se usará. Lo importante es que toda esta información se registra y uno puede planear con más exactitud tamaños y tiempo relacionados al desarrollo de su programa.  
  
Cuando codificamos directamente utilizando un IDE (Medio ambiente de desarrollo integrado) estamos de una u otra forma tratando de quitar errores de sintaxis a través de la compilación, sin considerar que la mayor cantidad de errores serán por lógica y que muchas de las veces estamos depurándolos hasta la fase de prueba, cuando estos errores deberían de ser descubiertos si revisáramos a consciencia el diseño realizado. Lo que realmente se espera es que el tiempo en hacer las pruebas sea solo el tiempo que le tome en capturar los datos para cada caso de prueba y que los resultados esperados sean realmente igual a los resultados obtenidos. Quizás en muy pocas ocasiones esto no suceda.  
  
La revisión incorporada en la metodología implica la realización de un documento completo del diseño definiendo la parte operacional, lógica, de estados y funcional; asimismo será importante checar el  
cumplimiento de estándares tanto en el diseño como en la construcción y revisar este código generado no usando el compilador, sino de forma manual para encontrar la mayor cantidad de errores antes de la primera compilación, siendo los errores de lógica verificados con tablas de rastreo o tablas de ejecución, entre otras.  
  
Tu productividad mejorará, los defectos antes cometidos serán minimizados o eliminados esto debido a su registro constante, tus estimaciones serán más certeras pudiendo cumplir con las fechas prometidas al líder de proyecto.  
  
No está por demás mencionar que es importante que quien esté desarrollando y vaya a aplicar la metodología debe dominar el lenguaje de programación. Creo que esta es un área de oportunidad para.

**Scrum**

Scrum es un proceso en el que se aplican de manera regular [un conjunto de buenas prácticas](https://proyectosagiles.org/fundamentos-de-scrum) para **trabajar colaborativamente, en equipo**, y obtener [el mejor resultado posible](https://proyectosagiles.org/beneficios-de-scrum) de un proyecto. Estas prácticas se apoyan unas a otras y su selección tiene origen en un [estudio de la manera de trabajar de equipos altamente productivos](https://proyectosagiles.org/historia-de-scrum).

En Scrum se realizan entregas parciales y regulares del producto final, priorizadas por el beneficio que aportan al receptor del proyecto. Por ello, Scrum está especialmente indicado para proyectos en **entornos complejos**, donde se necesita **obtener resultados pronto**, donde los **requisitos son cambiantes o poco definidos**, donde la **innovación**, la **competitividad**, la **flexibilidad** y la **productividad**son fundamentales.

Scrum también se utiliza para resolver situaciones en que **no se está entregando al cliente lo que necesita**, cuando **las entregas se alargan demasiado**, **los costes se disparan**o **la calidad no es aceptable**, cuando se necesita **capacidad de reacción ante la competencia**, cuando **la moral de los equipos es baja y la rotación alta**, cuando es necesario **identificar y solucionar ineficiencias sistemáticamente** o cuando se quiere trabajar utilizando un **proceso especializado en el desarrollo de producto**.

El proceso

En Scrum un proyecto se ejecuta en ciclos temporales cortos y de duración fija ([iteraciones](https://proyectosagiles.org/desarrollo-iterativo-incremental) que normalmente son de 2 semanas, aunque en algunos equipos son de 3 y hasta 4 semanas, límite máximo de feedback de producto real y reflexión). Cada iteración tiene que proporcionar un resultado completo, un incremento de producto final que sea susceptible de ser entregado con el mínimo esfuerzo al cliente cuando lo solicite.



El proceso parte de la [lista de objetivos/requisitos priorizada](https://proyectosagiles.org/lista-requisitos-priorizada-product-backlog) del producto, que actúa como plan del proyecto. En esta lista **el**[**cliente (Product Owner)**](https://proyectosagiles.org/cliente-product-owner)**prioriza los objetivos balanceando el valor que le aportan respecto a su coste** (que el [equipo](https://proyectosagiles.org/equipo-team/) estima considerando la [Definición de Hecho](https://proyectosagiles.org/definicion-de-hecho-definition-of-done/)) y quedan repartidos en iteraciones y entregas.

Las actividades que se llevan a cabo en Scrum son las siguientes (***los tiempos indicados son para iteraciones de 2 semanas***):

[Planificación de la iteración](https://proyectosagiles.org/planificacion-iteracion-sprint-planning)

El primer día de la iteración se realiza la reunión de planificación de la iteración. Tiene dos partes:

**Selección de requisitos**(2 horas). El cliente presenta al equipo la lista de requisitos priorizada del producto o proyecto. El equipo pregunta al cliente las dudas que surgen y selecciona los requisitos más prioritarios que prevé que podrá completar en la iteración, de manera que puedan ser entregados si el cliente lo solicita.

**Planificación de la iteración**(2 horas). El equipo elabora la [lista de tareas de la iteración](https://proyectosagiles.org/lista-tareas-iteracion-sprint-backlog) necesarias para desarrollar los requisitos seleccionados. La estimación de esfuerzo se hace de manera conjunta y los miembros del equipo se autoasignan las tareas, se [autoorganizan](https://proyectosagiles.org/2018/10/09/auto-organizacion-fundamentos-y-relacion-con-la-motivacion-intrinseca/) para trabajar incluso en parejas (o grupos mayores) con el fin de compartir conocimiento (creando un equipo más resiliente) o para resolver juntos objetivos especialmente complejos.

[Ejecución de la iteración](https://proyectosagiles.org/ejecucion-iteracion-sprint)

Cada día el equipo realiza una [reunión de sincronización](https://proyectosagiles.org/reunion-diaria-de-sincronizacion-scrum-daily-meeting)(15 minutos), normalmente delante de un [tablero físico o pizarra (Scrum Taskboard)](https://proyectosagiles.org/2010/09/26/ejemplo-tablero-pizarra-tareas-scrum-taskboard/). El equipo inspecciona el trabajo que el resto está realizando (dependencias entre tareas, progreso hacia el objetivo de la iteración, obstáculos que pueden impedir este objetivo) para poder hacer las adaptaciones necesarias que permitan cumplir con la previsión de objetivos a mostrar al final de la iteración. En la reunión cada miembro del equipo responde a tres preguntas:

¿Qué he hecho desde la última reunión de sincronización para ayudar al equipo a cumplir su objetivo?

¿Qué voy a hacer a partir de este momento para ayudar al equipo a cumplir su objetivo?

¿Qué impedimentos tengo o voy a tener que nos impidan conseguir nuestro objetivo?

Durante la iteración el [Facilitador (Scrum Master)](https://proyectosagiles.org/facilitador-scrum-master) se encarga de que el equipo pueda mantener el foco para cumplir con sus objetivos.

Elimina los obstáculos que el equipo no puede resolver por sí mismo.

Protege al equipo de interrupciones externas que puedan afectar el objetivo de la iteración o su productividad.

Durante la iteración, el cliente junto con el equipo [refinan la lista de requisitos (para prepararlos para las siguientes iteraciones) y, si es necesario, cambian o replanifican los objetivos del proyecto (10%-15% del tiempo de la iteración)](https://proyectosagiles.org/replanificacion-proyecto) con el objetivo de [maximizar la utilidad de lo que se desarrolla](https://proyectosagiles.org/beneficios-de-scrum#flexibilidad-adaptacion) y el [retorno de inversión](https://proyectosagiles.org/beneficios-de-scrum#gestion-roi).

Inspección y adaptación

El último día de la iteración se realiza la reunión de revisión de la iteración. Tiene dos partes:

[Revisión (demostración)](https://proyectosagiles.org/demostracion-requisitos-sprint-review)(1,5 horas). El equipo presenta al cliente los requisitos completados en la iteración, en forma de incremento de producto preparado para ser entregado con el mínimo esfuerzo. En función de los resultados mostrados y de los cambios que haya habido en el contexto del proyecto, el cliente realiza las adaptaciones necesarias de manera objetiva, ya desde la primera iteración, replanificando el proyecto.

[Retrospectiva](https://proyectosagiles.org/retrospectiva-sprint-retrospective)(1,5 horas). El equipo analiza cómo ha sido su manera de trabajar y cuáles son los problemas que podrían impedirle progresar adecuadamente, mejorando de manera continua su productividad. El Facilitador se encargará de eliminar o escalar los obstáculos identificados que estén más allá del ámbito de acción del equipo.

**CONCLUSIONES:**

Es absolutamente necesario aplicar una metodología aun proyecto estos nos permiten cumplir con tiempos y nos dan una mejor constancia y rendimiento de trabajo, se puede hacer un proyecto de software sin metodología, pero esto nos daría problemas a futuro pues la metodología con sus parámetros y reglas ayudan mucho a que se realice de una manera más profesional, presentándole al cliente una calidad en nuestro software.

WEBGRAFIA:

<https://sg.com.mx/content/view/797>

<https://es.wikipedia.org/wiki/Personal_Software_Process>

<https://proyectosagiles.org/que-es-scrum/>