

PSP 0.1



English

Entradas

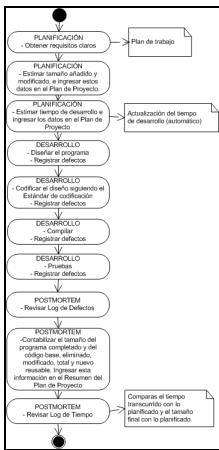
- Formulario del resumen del Plan del Proyecto PSPO.1
- Logs de registros de tiempo y defecto
- Estándar de tipos de defectos
- Descripción del problema
- Formulario PIP
- Cronómetro

Salidas

- Un programa probado cuidadosamente
- El formulario de Resumen del Plan de Proyecto completado con los datos estimados y actuales.
- Los Logs de Registro de Tiempo y Defecto completados
- El formulario PIP complementado

Solución

Proceso



Tiempo de Desarrollo

- ♦ Para adquirir el conocimiento necesario para desarrollar el producto software: 6 horas
- ♦ Para crear el Patrón de Producto: 2 días
- ♦ Para aplicar el Patrón de Producto: 3 horas

Video Explicación

Clase de Psp 0.1

Patrones Relacionados

- PSP 0

Controladores de Calidad

- Ninguno

Plantillas

 Logs de PSP 0.1  Guión para plantillas PSP 0.1

Ejemplos

 Ejemplo de conteo para PSP 0.1 .

Herramientas de Soporte

- Procesador de texto [[Microsoft Word](#)], [[OpenOffice Writer](#)]
- Lenguaje de programación [[Java](#)]
- Cronómetro

Contexto Inicial

Este producto se utiliza cuando deseemos tener una estimación precisa y exacta del tamaño de un programa, incorporando disciplina y medición a tu proceso. Se utilizan formularios para guiar en un proceso.

Contexto Resultante

El usuario del sistema obtendrá la declaración de requisitos documentados, el formulario de Resumen del Plan de Proyecto completado con los datos de tiempo de desarrollo estimado, el tamaño planificado y el Log de registro del Tiempo y formulario de propuesta de mejora del proceso (PIP) completados. El % completado a la fecha se va añadiendo.

Problema

Se desea utilizar una estructura apropiada para llevar a cabo tareas de pequeña escala y un marco de medición del tamaño que ocupan dichas tareas y del tiempo consumido para dichas tareas. A su vez sería propicio tener una base bien definida para la mejora y definición de procesos. Se pretende un control y corrección de defectos. Se determinan distintas piezas dentro del software.

Restricciones (*Forces*)

- **Tipo de Sistema:** Aplica a todos los tipos de sistemas.
- **Características de las organizaciones:** Este patrón puede utilizarse en los proyectos existentes en cualquier tipo de compañía.
- **Tipo de sistema a desarrollar:** Este producto puede utilizarse en proyectos en los que los requerimientos de usuario sean cambiantes.
- **Tipo de Cliente:** Debe existir, o debe conseguirse, que el área de negocio destinataria del desarrollo se implique en la consecución del mismo.
- **Paradigma de Programación:** Orientado a Objetos.

Roles

- Analista
- Cliente
- Jefe de Proyecto
- Usuarios del Sistema
- Desarrolladores

Lecciones Aprendidas

- Los grandes programas o aquellos que no son bien entendidos pueden requerir un enfoque iterativo. Con pequeños programas o bien entendidos, puedes ejecutar las fases en orden.
- Utilizar los Scripts (guiones). Esforzarse a utilizar este paradigma hasta que se convierta en un hábito.
- Un buen diseño hará que el programa pase la fase de pruebas con más facilidad.

Nivel de Madurez

- Nivel de Madurez 5 del CMMI.

Conocimientos y Habilidades Básicos

Conocimientos

- Definición de requisitos software (funcional y no funcional).
- Conocer cómo obtener datos de cómo trabajas realmente
- Habitarse a utilizar un paradigma concreto.
- Coger experiencia en registro de defectos a medida que son encontrados y corregidos.

Habilidades

- Capacidad de Abstracción.
- Capacidad de Análisis.

Recursos de Información

- Watts S. Humphrey. Introducción al proceso software personal. Addison Wesley. 2001.
 - Watts S. Humphrey. Introduction to the personal software process. Addison Wesley. 1997.
 - Watts S. Humphrey. PSP: a self-improvement process for software engineers. Addison Wesley. 2005.
-