

Campus Monterrey
Escuela de Ingeniería y Tecnologías de Información
Departamento de Ciencias Computacionales

Scripts del PSPSM

Clase TC-3045 "Calidad y Pruebas de Software"

Índice:

Script del Proceso PSP0:	.3
Script del Proceso PSP1:	
Script del Proceso PSP2:	.8
Script del Proceso PSP2.1:1	12

Script del Proceso PSP01:

Propósito	Guiar el proceso de desarrollo de un programa computacional, utilizando el Process
	Dashboard como herramienta de captura de métricas personales.
Criterios de Entrada	- Descripción del Problema
	- Process Dashboard con las fases del PSPO asignadas al programa a desarrollar
	-Forma de Reporte de Casos de Prueba
	-Estándar de Tipo de Defectos

		Estandar de ripo de Derectos
#	Paso	Descripción
1		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
	_	-Obtener la descripción de los requerimientos
	<u>Planeación</u> ²	- Estudiar y entender los requerimientos
-	(Planning) ³	- Estimar el tiempo que tomará el desarrollo del programa y capturarlo en la forma Resumen
		del Plan (Project Plan Summary)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		- Utilizando como base los requerimientos:
2	Diseño	■ Diseñar en forma completa el programa
-	(Design)	■ Diseñar todos los casos de pruebas unitarias
		-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
3	Codificación	- Escribir el código de acuerdo al diseño realizado
	(Code)	-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
4	Compilación	-Compilar el código escrito (repetir hasta que el compilador no marque errores)
-	(Compile)	-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
	Pruebas Unitarias (Test)	- Ejecutar todos los casos de prueba diseñados (repetir hasta que ya no se encuentren
5		errores)
	(1555)	-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
6	<u>Postmortem</u>	- Revisar que se hayan registrado en forma completa y correcta todos los tiempos y los
	(Postmortem)	defectos
		- Detener el reloj y marcar la fase y el programa como terminados
Criterios de Salida		- Programa pasa exitosamente todas las pruebas unitarias y que cumple con los
		requerimientos
		- Resumen del Plan (<i>Project Plan Summary</i>) lleno con los datos estimados y reales
		-Bitácora de tiempo (<i>Time Log</i>) y de defectos (<i>Defect Log</i>) completas
		- Reporte de pruebas completo

¹ Traducido y adaptado del Proceso PSPO que se encuentra descrito en la página 18 del libro PSP A Self-Improvement Process for Software Engineers del autor Watts S. Humphrey de la editorial Addison Wesley.

 $^{^{2}\,\}text{Los}$ pasos que se encuentran $\underline{\text{subrayados}}$ tienen un sub-script que los detalla.

³ Entre paréntesis y en *itálica* se muestra el nombre en inglés utilizado por Process Dashboard.

Script PSP0 para la Planeación:

Pro	pósito	Guiar el proceso de planeación en PSP
Crite	erios de Entrada	- Descripción del Problema - Process Dashboard con las fases del PSPO asignadas al programa a desarrollar
		_ · · · ·
#	Paso	Descripción

#	Paso	Descripción
1	Requerimientos del Programa	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Obtener o producir el documento de los requerimientos del programa a desarrollar Asegurar que los requerimientos son claros y no ambiguos Obtener respuesta a todas las dudas
2	Estimado de Tiempo	 - Hacer el mejor estimado del tiempo que tomará desarrollar el programa - Capturar el tiempo estimado en la forma Resumen del Plan (Project Plan Summary) - Detener el reloj y marcar la fase de planeación como terminada
Critarias da Salida Decumento de requerimientos completo y entendido		

Criterios de Salida	-Documento de requerimientos completo y entendido
	-Forma Resumen del Plan (Project Plan Summary) con los datos del tiempo estimado
	-Bitácora de tiempo (Time Log) completa para esta fase

Script PSP0 para el Postmortem:

Propósito	Guiar el proceso de postmortem en PSP
Criterios de Entrada	-Documento de requerimientos completo y entendido
	- Programa pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos
	- Process Dashboard con todos los datos recolectados al estar realizando el programa

#	Paso	Descripción
1	Registro de Defectos	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisar el Resumen del Plan (Project Plan Summary) para verificar que todos los defectos encontrados en cada fase están registrados Utilizando el mejor recuerdo, registrar en la bitácora de defectos (Defect Log) aquellos defectos que se hayan olvidado registrar
2	Consistencia de los datos de los defectos	 -Verificar que en la bitácora de defectos (Defect Log) los datos de cada defecto estén correctos y completos -Verificar que en el Resumen del Plan (Project Plan Summary) la cantidad de defectos inyectados y removidos por fase sean razonables y completos - Utilizando el mejor recuerdo, corregir cualquier dato que falte o esté incorrecto
3	Registro de tiempo	-Verificar la bitácora de tiempo (<i>Time Log</i>) buscando errores u omisiones - Utilizando el mejor recuerdo, corregir cualquier dato de tiempo que falte o esté incorrecto - Detener el reloj y marcar la fase de postmortem como terminada - Marcar el programa como terminado
Criterios de Salida		- Programa pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos

Criterios de Salida	-Programa pasa exitosamente todas las pruebas unitarias y que cumple con los
	requerimientos
	-Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales
	-Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas
	-Reporte de pruebas completo

Script del Proceso PSP14:

Propósito	Guiar el proceso de desarrollo de un programa computacional, utilizando el Process
	Dashboard como herramienta de captura de métricas personales.
Criterios de Entrada	- Descripción del Problema
	-Process Dashboard con las fases del PSP1 asignadas al programa a desarrollar
	-Forma de Reporte de Casos de Prueba
	- Estándar de Tipo de Defectos
	- Estándares de Codificación y Contabilización
	- Datos históricos de tiempo y tamaño

		Dutos historicos de tiempo y tumano
#	Paso	Descripción
	<u>Planeación</u> (<i>Planning</i>)	-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		-Obtener la descripción de los requerimientos
		-Estudiar y entender los requerimientos
		- Utilizar el método PROBE para estimar las LDC agregadas y modificadas
1		- Utilizar el método PROBE para estimar el tiempo que tomará el desarrollo del programa
		- Capturar los estimados en la Plantilla de Estimación de Tamaño (Size Estimating Template)
		- Verificar en la forma Resumen del Plan (Project Plan Summary) que la información
		planeada es correcta
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		- Utilizando como base los requerimientos:
2	Diseño	■ Diseñar en forma completa el programa
_	(Design)	■ Diseñar todos los casos de pruebas unitarias
		-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		-Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
	Codificación	-Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y
3	(Code)	contabilización
		-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
	0 11 17	-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
4	Compilación	-Compilar el código escrito (repetir hasta que el compilador no marque errores)
	(Compile)	- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
	Donale e d'Unite de la	-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
5	Pruebas Unitarias (Test)	- Ejecutar todos los casos de prueba diseñados (repetir hasta que ya no se encuentren errores)
		- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (<i>Defect Log</i>)
		- Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		- Capturar en la Plantilla de Estimación de Tamaño (Size Estimating Template) y en el
	Postmortem (Postmortem)	Resumen del Plan (<i>Project Plan Summary</i>) el tamaño real del programa
6		- Revisar que se hayan registrado en forma completa y correcta todos los tiempos y defectos
		- Elaborar una Propuesta de Mejora de Proceso (<i>Process Improvement Proposal</i>)
		- Detener el reloj y marcar la fase y el programa como terminados
Crit	erios de Salida	-Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los
		requerimientos y con los estándares de codificación y contabilización
		-Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales

Criterios de Salida	- Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los
	requerimientos y con los estándares de codificación y contabilización
	-Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales
	- Plantilla de Estimación de Tamaño (Size Estimating Template) completa
	- Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas
	- Reporte de pruebas completo
	- Formato de Propuesta de Mejora de Proceso (Process Improvement Proposal) completo

 $^{^{4}}$ Traducido y adaptado del Proceso PSP 1.0 del material académico proporcionado por el SEI.

Script PSP1 para la Planeación:

Propósito	Guiar el proceso de planeación en PSP
Criterios de Entrada	- Descripción del Problema
	-Process Dashboard con las fases del PSP1 asignadas al programa a desarrollar
	- Datos históricos de tiempo y tamaño

		- Datos históricos de tiempo y tamaño
#	Paso	Descripción
1	Requerimientos del Programa	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Obtener o producir el documento de los requerimientos del programa a desarrollar Asegurar que los requerimientos son claros y no ambiguos Obtener respuesta a todas las dudas
2	Estimado de Tamaño	- Realizar un diseño conceptual del programa a desarrollar - Capturar los estimados de LDC de todas las partes en la Plantilla de Estimación de Tamaño (Size Estimating Template) - Utilizar el método PROBE para estimar las LDC agregadas y modificadas para este programa (en Process Dashboard utilizar el "PROBE Wizard")
3	Estimado de Tiempo	- Utilizar el método PROBE para estimar el tiempo que tomará desarrollar el programa - Capturar el tiempo estimado en la Plantilla de Estimación de Tamaño (Size Estimating Template) - Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detener el reloj y marcar la fase de planeación como terminada
Crit	erios de Salida	- Documento de requerimientos completo y entendido - Diseño conceptual del programa - Plantilla de Estimación de Tamaño (Size Estimating Template) completa - Forma Resumen del Plan (Project Plan Summary) con estimados de tiempo y tamaño completos - Bitácora de tiempo (Time Log) completa para esta fase

Script PSP1 para el Postmortem:

Propósito	Guiar el proceso de postmortem en PSP
Criterios de Entrada	-Documento de requerimientos completo y entendido
	-Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los
	requerimientos y con los estándares de codificación y contabilización
	- Process Dashboard con todos los datos recolectados al estar realizando el programa

ш	D	Description of the
#	Paso	Descripción
1	Registro de Defectos	-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		-Revisar el Resumen del Plan (<i>Project Plan Summary</i>) para verificar que todos los defectos
		encontrados en cada fase están registrados
		- Utilizando el mejor recuerdo, registrar en la bitácora de defectos (<i>Defect Log</i>) aquellos
		defectos que se hayan olvidado registrar
	Consistencia de los datos de los	-Verificar que en la bitácora de defectos (<i>Defect Log</i>) los datos de cada defecto estén correctos y completos
2		-Verificar que en el Resumen del Plan <i>(Project Plan Summary)</i> la cantidad de defectos
_	defectos	inyectados y removidos por fase sean razonables y completos
	defectos	-Utilizando el mejor recuerdo, corregir cualquier dato que falte o esté incorrecto
		- Medir el tamaño real del programa
		- Determinar el tamaño de las partes base (LDC base, borradas, modificadas, agregadas),
		partes nuevas (cantidad de ítems y tamaño en LDC) y partes reusadas (LDC).
3	Tamaño	-Capturar estos tamaños en la Plantilla de Estimación de Tamaño (Size Estimating
		Template)
		-Calcular el Tamaño TOTAL del programa y capturarlo en el Resumen del Plan (Project Plan
		Summary)
4	Dogistro do tiomas	-Verificar la bitácora de tiempo (Time Log) buscando errores u omisiones
4	Registro de tiempo	-Utilizando el mejor recuerdo, corregir cualquier dato de tiempo que falte o esté incorrecto
		- Analizar las métricas de la forma Resumen del Plan para encontrar una o más áreas de
		mejora en el proceso personal
		-Documentar la(s) propuesta(s) de mejora en la forma Propuesta de Mejora de Proceso
5	Mejora de Proceso	(Process Improvement Proposal) describiendo el (los) problema(s), la(s) sugerencia(s) de
		mejora y/o la(s) lección(es) aprendida(s)
		- Detener el reloj y marcar la fase de postmortem como terminada
		-Marcar el programa como terminado
Crit	erios de Salida	-Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los
		requerimientos y con los estándares de codificación y contabilización
		-Resumen del Plan (<i>Project Plan Summary</i>) lleno con los datos estimados y reales
		- Plantilla de Estimación de Tamaño (Size Estimating Template) completa
		-Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas
		-Reporte de pruebas completo
		-Formato de Propuesta de Mejora de Proceso (<i>Process Improvement Proposal</i>) completo

Script del Proceso PSP2⁵:

Propósito	Guiar el proceso de desarrollo de un programa computacional, utilizando el Process Dashboard como herramienta de captura de métricas personales.
Criterios de Entrada	- Descripción del Problema - Process Dashboard con las fases del PSP2 asignadas al programa a desarrollar - Forma de Reporte de Casos de Prueba - Estándar de Tipo de Defectos - Estándares de Codificación y Contabilización - Checklists de revisión de diseño y de revisión de código - Datos históricos de tiempo y tamaño

#	Paso	Descripción
		- Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		-Obtener la descripción de los requerimientos
		-Estudiar y entender los requerimientos
	Dlangasián	- Utilizar el método PROBE para estimar las LDC agregadas y modificadas
1	<u>Planeación</u> (Planning)	- Utilizar el método PROBE para estimar el tiempo que tomará el desarrollo del programa
	(Planning)	- Capturar los estimados en la Plantilla de Estimación de Tamaño (Size Estimating Template)
		- Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada
		es correcta
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		- Utilizando como base los requerimientos:
2	Diseño	■ Diseñar en forma completa el programa
-	(Design)	■ Diseñar todos los casos de pruebas unitarias
		-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
	Davidska dal Diagra	-Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de
3	Revisión del Diseño	diseño
	(Design Review)	- Corregir todos los defectos que se encuentren
		- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		-Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y
4	Codificación	contabilización
-	(Code)	-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		-Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
		-Revisar el código siguiendo el script de revisión de código y el checklist de revisión de
_	Revisión del Código	código
5	(Code Review)	-Corregir todos los defectos que se encuentren
		-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
6	Compilación	-Compilar el código escrito (repetir hasta que el compilador no marque errores)
0	(Compile)	- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada
		- Iniciar el reloj (pausarlo cada vez que haya una interrupción)
7	Pruebas Unitarias	-Ejecutar todos los casos de prueba diseñados (repetir hasta que ya no se encuentren errores)
′	(Test)	-Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)
		- Detener el reloj y marcar la fase como terminada

 $^{^{5}}$ Traducido y adaptado del Proceso PSP 2.0 del material académico proporcionado por el SEI.

Script del Proceso PSP2: (continuación)

8	Postmortem (Postmortem)	 -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Capturar en la Plantilla de Estimación de Tamaño (Size Estimating Template) y en el Resumen del Plan (Project Plan Summary) el tamaño real del programa -Revisar que se hayan registrado en forma completa y correcta todos los tiempos y defectos -Elaborar una Propuesta de Mejora de Proceso (Process Improvement Proposal) -Detener el reloj y marcar la fase y el programa como terminados
Crit	erios de Salida	 - Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos y con los estándares de codificación y contabilización - Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales - Plantilla de Estimación de Tamaño (Size Estimating Template) completa - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas - Reporte de pruebas completo - Checklists de revisión de diseño y de revisión de código completos - Formato de Propuesta de Mejora de Proceso (Process Improvement Proposal) completo

Script PSP2 para la Planeación:

Propósito		Guiar el proceso de planeación en PSP
Criterios de Entrada		- Descripción del Problema - Process Dashboard con las fases del PSP2 asignadas al programa a desarrollar
		-Datos históricos de tiempo y tamaño
#	Paso	Descripción
1	Requerimientos del Programa	-Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Obtener o producir el documento de los requerimientos del programa a desarrollar -Asegurar que los requerimientos son claros y no ambiguos -Obtener respuesta a todas las dudas

1	Requerimientos del	- Obtener o producir el documento de los requerimientos del programa a desarrollar
1	Programa	-Asegurar que los requerimientos son claros y no ambiguos
		-Obtener respuesta a todas las dudas
		- Realizar un diseño conceptual del programa a desarrollar
	Estimado de Tamaño	-Capturar los estimados de LDC de todas las partes en la Plantilla de Estimación de Tamaño
2		(Size Estimating Template)
		- Utilizar el método PROBE para estimar las LDC agregadas y modificadas para este programa (en Process Dashboard utilizar el "PROBE Wizard")
		- Utilizar el método PROBE para estimar el tiempo que tomará desarrollar el programa - Capturar el tiempo estimado en la Plantilla de Estimación de Tamaño (Size Estimating
3	Estimado de Tiempo	Template)
3		-Verificar en la forma Resumen del Plan (<i>Project Plan Summary</i>) que la información planeada
		es correcta
		-Basado en los datos históricos de defectos totales y los inyectados y removidos en cada
		fase, Process Dashboard calcula el cantidad de defectos a inyectar y remover en cada fase
	Estimado de	de este programa
4	defectos	- Ajustar los estimados anteriores para incluir los defectos a remover en las revisiones del
	derectos	diseño y del código
		- Detener el reloj y marcar la fase de planeación como terminada
		Beterier erreioj y marcar la rase de planeación como terminada
Criterios de Salida		-Documento de requerimientos completo y entendido
		- Diseño conceptual del programa

Criterios de Salida	- Documento de requerimientos completo y entendido
	- Diseño conceptual del programa
	-Plantilla de Estimación de Tamaño (Size Estimating Template) completa
	-Forma Resumen del Plan (Project Plan Summary) con estimados de tiempo y tamaño
	completos
	-Bitácora de tiempo <i>(Time Log)</i> completa para esta fase

Script PSP2 para la revisión del diseño:

Propósito	Guiar el proceso de revisión del diseño detallado
Criterios de Entrada	- Diseño del programa completo
	- Diseño de los casos de prueba completos
	-Checklist de revisión de diseño
	-Documento de requerimientos completo y entendido
	- Estándar de Tipo de Defectos

#	Paso	Descripción
1	Preparación	- Descansar un momento antes de revisar el diseño - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Analizar la complejidad del diseño y el checklist de revisión, para decidir la mejor estrategia de revisión
2	Revisión	-Seguir el checklist de revisión de diseño -Revisar parte por parte -Dentro de cada parte, revisar categoría por categoría -Cada vez que se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos (Defect Log) -Marcar cada categoría al terminar de revisarla
3	Revisión de Correcciones	 Revisar que se hayan corregido todos los defectos encontrados Revisar todos los cambios realizados Si se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos y, si procede, llenar el campo "Fix Defect" del defecto que se estaba arreglando cuando se inyectó este nuevo defecto Detener el reloj y marcar la fase de revisión de diseño como terminada
Crit	erios de Salida	- Diseño del programa completamente revisado - Diseño de las pruebas completamente revisado - Checklist de revisión marcado en forma completa de acuerdo a las partes que se revisaron - Todos los defectos encontrados fueron corregidos - Todos los cambios fueron revisados - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas

Script PSP2 para la revisión del código:

Propósito	Guiar el proceso de revisión del código
Criterios de Entrada	-Código del programa completo
	-Checklist de revisión de código
	- Documento de requerimientos completo y entendido
	-Estándar de Tipo de Defectos
	- Estándares de Codificación y Contabilización

#	Paso	Descripción
1	Preparación	 - Descansar un momento antes de revisar el código - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Analizar la complejidad del código y el checklist de revisión, para decidir la mejor estrategia de revisión
2	Revisión	- Seguir el checklist de revisión de código - Revisar parte por parte - Dentro de cada parte, revisar categoría por categoría - Cada vez que se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos (Defect Log) - Marcar cada categoría al terminar de revisarla

Script PSP2 para la revisión del código: (continuación)

3	Revisión de Correcciones	 Revisar que se hayan corregido todos los defectos encontrados Revisar todos los cambios realizados Si se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos y, si procede, llenar el campo "Fix Defect" del defecto que se estaba arreglando cuando se inyectó este nuevo defecto Detener el reloj y marcar la fase de revisión de código como terminada
Criterios de Salida		- Código del programa completamente revisado - Checklist de revisión marcado en forma completa de acuerdo a las partes que se revisaron - Todos los defectos encontrados fueron corregidos - Todos los cambios fueron revisados - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas

Script PSP2 para el Postmortem:

Propósito	Guiar el proceso de postmortem en PSP
Criterios de Entrada	-Documento de requerimientos completo y entendido
	- Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos y con los estándares de codificación y contabilización
	-Process Dashboard con todos los datos recolectados al estar realizando el programa
	-Checklists de revisión de diseño y de revisión de código completos

#	Paso	Descripción
1	Registro de Defectos	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisar el Resumen del Plan (Project Plan Summary) para verificar que todos los defectos encontrados en cada fase están registrados Utilizando el mejor recuerdo, registrar en la bitácora de defectos (Defect Log) aquellos defectos que se hayan olvidado registrar
2	Consistencia de los datos de los defectos	 -Verificar que en la bitácora de defectos (Defect Log) los datos de cada defecto estén correctos y completos -Verificar que en el Resumen del Plan (Project Plan Summary) la cantidad de defectos inyectados y removidos por fase sean razonables y completos -Calcula el yield del proceso y verifica que el valor es razonable y correcto -Utilizando el mejor recuerdo, corregir cualquier dato que falte o esté incorrecto
3	Tamaño	 - Medir el tamaño real del programa - Determinar el tamaño de las partes base (LDC base, borradas, modificadas, agregadas), partes nuevas (cantidad de ítems y tamaño en LDC) y partes reusadas (LDC). - Capturar estos tamaños en la Plantilla de Estimación de Tamaño (Size Estimating Template) - Calcular el Tamaño TOTAL del programa y capturarlo en el Resumen del Plan (Project Plan Summary)
4	Registro de tiempo	- Verificar la bitácora de tiempo (<i>Time Log</i>) buscando errores u omisiones - Utilizando el mejor recuerdo, corregir cualquier dato de tiempo que falte o esté incorrecto
5	Mejora de Proceso	 - Analizar las métricas de la forma Resumen del Plan para encontrar una o más áreas de mejora en el proceso personal - Documentar la(s) propuesta(s) de mejora en la forma Propuesta de Mejora de Proceso (Process Improvement Proposal) describiendo el (los) problema(s), la(s) sugerencia(s) de mejora y/o la(s) lección(es) aprendida(s) - Detener el reloj y marcar la fase de postmortem como terminada - Marcar el programa como terminado
Cuitanias da Calida		Ducarent and property and prope

Criterios de Salida	- Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los
	requerimientos y con los estándares de codificación y contabilización
	-Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales
	-Plantilla de Estimación de Tamaño (Size Estimating Template) completa
	-Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas
	-Reporte de pruebas completo
	-Checklists de revisión de diseño y de revisión de código completos
	-Formato de Propuesta de Mejora de Proceso (<i>Process Improvement Proposal</i>) completo
	-1 offiliato de Propuesta de Mejora de Proceso (Proceso Improvement Proposur) completo

Script del Proceso PSP2.16:

Propósito	Guiar el proceso de desarrollo de un programa computacional, utilizando el Process
	Dashboard como herramienta de captura de métricas personales.
Criterios de Entrada	- Descripción del Problema
	-Process Dashboard con las fases del PSP2.1 asignadas al programa a desarrollar
	- Forma de Reporte de Casos de Prueba
	- Estándar de Tipo de Defectos
	- Estándares de Codificación y Contabilización
	-Checklists de revisión de diseño y de revisión de código
	- Datos históricos de tiempo y tamaño

- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Obtener la descripción de los requerimientos - Estudiar y entender los requerimientos - Utilizar el método PROBE para estimar las LDC agregadas y modificadas y, en el caso de los métodos A y B, el intervalo de predicción del tamaño - Utilizar el método PROBE para estimar las LDC agregadas y modificadas y, en el caso de los métodos A y B, el intervalo de predicción del tiempo - Capturar los estimados en la Plantilla de Estimación del tiempo - Capturar los estimados en la Plantilla de Estimación del Tamaño (Size Estimating Template) - Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño para cumplir con esta especificación de estados y lógica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisian el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Codificación (Code) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el	#	Paso	Descripción
- Estudiar y entender los requerimientos - Utilizar el método PROBE para estimar las LDC agregadas y modificadas y, en el caso de los métodos A y B, el intervalo de predicción del tamaño - Utilizar el método PROBE para estimar el tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo que tomará el desarrollo del programa y, en el caso de los métodos de la metodo programa y, en el caso de los métodos del programa y, en el caso de los métodos del programa y, en el caso de los métodos del programa y, en el caso de los métodos de los mátodos de la metodo programa y, en el caso de los estimatos de project programa y, en el caso de ver que haya una interrupción) Revisión del Diseño (Design Review) 1			
- Utilizar el método PROBE para estimar las LDC agregadas y modificadas y, en el caso de los métodos A y B, el intervalo de predicción del tamaño - Utilizar el método PROBE para estimar el tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo - Capturar los estimados en la Plantilla de Estimación de Tamaño (Size Estimating Template) - Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación de estados y lógica - Basado en la especificación enterna en las plantillas de especificación de estados y lógica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño (Design Review) - Codificación (Code) - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un d			l
Planeación (Planning) ###odos A y B, el intervalo de predicción del tamaño - Utilizar el método PROBE para estimar el tiempo que tomará el desarrollo del programa y, en el caso de los métodos A y B, el intervalo de predicción del tiempo - Capturar los estimados en la Plantilla de Estimación de Tamaño (Size Estimating Template) - Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño en las plantillas de especificación externa - Registrar este diseño en las plantillas de especificación externa - Registrar este diseño en las plantillas de especificación externa - Revisión del Diseño (Design Review) - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el soript de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código siguiendo el script de revisión de código y el checklist de revisión de codigo y el checklist de revisión de código y el checklist de revisión de codigo y el checklist de revisión de codigo y el			
Planteación (Planning) Planteación (Planning) Planteación (Planning) Planteación (Planning) Planteación			
Codificación (Code) Code Review		Planeación	
Capturar los estimados en la Plantilla de Estimación del Tamaño (Size Estimating Template) - Capturar los estimados en la Plantilla de Estimación del Tamaño (Size Estimating Template) - Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detenner el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Producir un diseño para cumplir con esta especificación externa - Registrar esta diseño en las plantillas de especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisión del Diseño (Design Review) - Codificación (Code) - Codificación (Code) - Codificación (Code) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisión del Código (Code Review) - Revisión d	1		
-Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada es correcta - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño en las plantillas de especificación de estados y lógica - Basado en la especificación peracional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -		(ranning)	
es correcta - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño en las plantillas de especificación de estados y lógica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que			, , , , , , , , , , , , , , , , , , , ,
- Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa para cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño en las plantillas de especificación externa - Registrar este diseño en las plantillas de especificación de estados y lógica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos			
- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Estudiar los requerimientos y producir una especificación externa cumplirlos - Registrar esta especificación externa en las plantillas de especificación operacional y funcional - Producir un diseño para cumplir con esta especificación externa - Registrar este diseño en las plantillas de especificación externa - Registrar este diseño en las plantillas de especificación de estados y lógica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
-Estudiar los requerimientos y producir una especificación externa para cumplirlos -Registrar esta especificación externa en las plantillas de especificación operacional y funcional -Producir un diseño para cumplir con esta especificación externa -Registrar este diseño en las plantillas de especificación de estados y lógica -Basado en la especificación operacional diseñar todos los casos de pruebas unitarias -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj (pausarlo cada vez que haya una interrupción) -Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj (pausarlo cada vez que haya una interrupción) -Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una i			
Producir un diseño para cumplir con esta especificación externa en las plantillas de especificación operacional y funcional Producir un diseño para cumplir con esta especificación de estados y lógica Pasado en la especificación operacional diseñar todos los casos de pruebas unitarias Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisión del Diseño Corregir todos los defectos que se encuentren Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada Iniciar el reloj (pausarlo cada vez que haya una interrupción) Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisión del Código Code Review) Revisión del Código (Code Review) Revisión del Código siguiendo el script de revisión de código y el checklist de revisión de código Corregir todos los defectos que se encuentren Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada Iniciar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compil			, , , , , , , , , , , , , , , , , , , ,
Diseño (Design) Diseño (Design) Diseño (Design) Diseño (Design) Diseño (Design) Diseño (Design) Diseño (Design) Detener el reloj y marcar la fase como terminada Diseño (Design Review) Detener el reloj y marcar la fase como terminada Diseño (Design Review) Detener el reloj y marcar la fase como terminada Diseño (Design Review) Detener el reloj y marcar la fase como terminada Diseño (Design Review) Detener el reloj y marcar la fase como terminada Diniciar el reloj (pausarlo cada vez que haya una interrupción) Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño Corregir todos los defectos que se encuentren Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj (pausarlo cada vez que haya una interrupción) Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj (pausarlo cada vez que haya una interrupción) Revisión del Código (Code Review) Detener el reloj (pausarlo cada vez que haya una interrupción) Revisión del Código (Code Review) Detener el reloj (pausarlo cada vez que haya una interrupción) Compilación (Compile) Detener el reloj (pausarlo cada vez que haya una interrupción) Compilación (Compile) Detener el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo cada vez que haya una interrupción) Compilar el reloj (pausarlo ca			
Diseno (Design) -Producir un diseño para cumplir con esta especificación externa -Registrar este diseño en las plantillas de especificación de estados y lógica -Basado en la especificación operacional diseñar todos los casos de pruebas unitarias -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código -Corregir todos los defectos que se encuentren -Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) -Detener el reloj y marcar la fase como terminada -Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una interrupción) -Compilar el reloj (pausarlo cada vez que haya una i			
Registrar este diseño en las plantillas de específicación de estados y lógica		Diseño	
- Registrar este aiseno en las plantillas de especificación do estados y logica - Basado en la especificación operacional diseñar todos los casos de pruebas unitarias - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compele y marcar la fase como terminada - Iniciar el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compila el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Compila el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	2	(Design)	
- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)		, 3,	
- Detener el reloj y marcar la fase como terminada Revisión del Diseño (Design Review) - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño (Corregir todos los defectos que se encuentren) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código (Jeausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisión del Diseño (Design Review) - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Revisión del Diseño (Design Review) - Revisar el diseño siguiendo el script de revisión de diseño y el checklist de revisión de diseño - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación - Compilación - Compilación (Compile) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación - Compilación - Compilación - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación - Compilación - Compilación (Compile) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	2	Revisión del Diseño	
- Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	3		
- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
Codificación (Code) - Escribir el código de acuerdo al diseño y siguiendo los estándares de codificación y contabilización - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			• •
Contabilización Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisión del Código (Code Review) Revisión del Código (Code Review) Revisión del Código (Code Review) Compilación Compilación (Compile) Contabilización Co			, , , , , , , , , , , , , , , , , , , ,
- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	1		, ,
- Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisión del Código (Code Review) - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	-		
- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Revisión del Código (Code Review) - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Revisión del Código (Code Review) - Revisar el código siguiendo el script de revisión de código y el checklist de revisión de código (Code Review) - Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			• •
- Corregir todos los defectos que se encuentren - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) - Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)	5		
- Detener el reloj y marcar la fase como terminada - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación (Compile) - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Compilación - Compilación (compile) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
Compilación (Compile) - Compilar el código escrito (repetir hasta que el compilador no marque errores) - Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
- Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log)			
	6		
			-Detener el reloj y marcar la fase como terminada

 $^{^{6}}$ Traducido y adaptado del Proceso PSP 2.1 del material académico proporcionado por el SEI.

Script del Proceso PSP2.1: (continuación)

7	Pruebas Unitarias (Test)	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Ejecutar todos los casos de prueba diseñados (repetir hasta que ya no se encuentren errores) Cada vez que se corrige un defecto, registrarlo en la bitácora de defectos (Defect Log) Detener el reloj y marcar la fase como terminada
8	Postmortem (Postmortem)	-Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Capturar en la Plantilla de Estimación de Tamaño (Size Estimating Template) y en el Resumen del Plan (Project Plan Summary) el tamaño real del programa -Revisar que se hayan registrado en forma completa y correcta todos los tiempos y defectos -Elaborar una Propuesta de Mejora de Proceso (Process Improvement Proposal) -Detener el reloj y marcar la fase y el programa como terminados
Criterios de Salida		 - Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos y con los estándares de codificación y contabilización - Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales - Plantilla de Estimación de Tamaño (Size Estimating Template) completa - Plantillas de diseño completas - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas - Reporte de pruebas completo - Checklists de revisión de diseño y de revisión de código completos - Formato de Propuesta de Mejora de Proceso (Process Improvement Proposal) completo

Script PSP2.1 para la Planeación:

Propósito	Guiar el proceso de planeación en PSP
Criterios de Entrada	- Descripción del Problema
	-Process Dashboard con las fases del PSP2.1 asignadas al programa a desarrollar
	- Datos históricos de tiempo y tamaño

#	Paso	Descripción
1	Requerimientos del	- Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Obtener o producir el documento de los requerimientos del programa a desarrollar
	Programa	-Asegurar que los requerimientos son claros y no ambiguos
	Trograma	- Obtener respuesta a todas las dudas
		-Realizar un diseño conceptual del programa a desarrollar
		-Capturar los estimados de LDC de todas las partes en la Plantilla de Estimación de Tamaño
	Estimada da	(Size Estimating Template)
2	Estimado de Tamaño	-Utilizar el método PROBE para estimar las LDC agregadas y modificadas para este programa
	Tallialio	(en Process Dashboard utilizar el "PROBE Wizard")
		- Para los métodos A y B, calcular el intervalo de 70% de predicción del tamaño (en Process
		Dashboard utilizar el "PROBE Wizard")
		- Utilizar el método PROBE para estimar el tiempo que tomará desarrollar el programa
		-Capturar el tiempo estimado en la Plantilla de Estimación de Tamaño (Size Estimating
	Estimado de Tiempo	Template)
3		- Para los métodos A y B, calcular el intervalo de 70% de predicción del tiempo (en Process
		Dashboard utilizar el "PROBE Wizard")
		- Verificar en la forma Resumen del Plan (Project Plan Summary) que la información planeada
		es correcta
4		-Basado en los datos históricos de defectos totales y los inyectados y removidos en cada
	Estimado de	fase, Process Dashboard calcula el cantidad de defectos a inyectar y remover en cada fase
	defectos	de este programa
		- Detener el reloj y marcar la fase de planeación como terminada

Script PSP2.1 para la Planeación: (continuación)

Criterios de Salida	-Documento de requerimientos completo y entendido
	- Diseño conceptual del programa
	-Plantilla de Estimación de Tamaño (Size Estimating Template) completa
	-Forma Resumen del Plan (Project Plan Summary) con estimados de tiempo y tamaño
	completos y con intervalos de predicción del tamaño y del tiempo
	-Bitácora de tiempo (Time Log) completa para esta fase

Script PSP2.1 para la revisión del diseño:

Propósito		Guiar el proceso de revisión del diseño detallado
Criterios de Entrada		-Diseño del programa completo
		- Diseño de los casos de prueba completos
		- Checklist de revisión de diseño
		- Documento de requerimientos completo y entendido
		-Estándar de Tipo de Defectos
#	Paso	Descripción
		- Descansar un momento antes de revisar el diseño
1	Preparación	-Iniciar el reloj (pausarlo cada vez que haya una interrupción) -Analizar la complejidad del diseño y el checklist de revisión, para decidir la mejor estrategia

#	Paso	Descripción
		- Descansar un momento antes de revisar el diseño
1	Dronaración	-Iniciar el reloj (pausarlo cada vez que haya una interrupción)
	Preparación	-Analizar la complejidad del diseño y el checklist de revisión, para decidir la mejor estrategia
		de revisión
		-Seguir el checklist de revisión de diseño
		-Revisar parte por parte
		- Dentro de cada parte, revisar categoría por categoría
2	Revisión	- Utilizar tablas de rastreo u otros métodos para verificar que el diseño es correcto
		-Cada vez que se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos
		(Defect Log)
		- Marcar cada categoría al terminar de revisarla
		-Revisar que se hayan corregido todos los defectos encontrados
3		-Revisar todos los cambios realizados
	Revisión de	-Si se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos y, si procede,
3	Correcciones	llenar el campo "Fix Defect" del defecto que se estaba arreglando cuando se inyectó este
		nuevo defecto
		-Detener el reloj y marcar la fase de revisión de diseño como terminada
Criterios de Salida		- Diseño del programa completamente revisado
		- Diseño de las pruebas completamente revisado
		-Checklist de revisión marcado en forma completa de acuerdo a las partes que se revisaron
		- Pesultados documentados del análisis del diseño

Criterios de Salida	- Diseño del programa completamente revisado
	-Diseño de las pruebas completamente revisado
	-Checklist de revisión marcado en forma completa de acuerdo a las partes que se revisaron
	-Resultados documentados del análisis del diseño
	-Todos los defectos encontrados fueron corregidos
	-Todos los cambios fueron revisados
	-Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas

Script PSP2.1 para la revisión del código:

Propósito	Guiar el proceso de revisión del código
Criterios de Entrada	-Código del programa completo
	-Checklist de revisión de código
	-Documento de requerimientos completo y entendido
	-Estándar de Tipo de Defectos
	- Estándares de Codificación y Contabilización

#	Paso	Descripción
1	Preparación	 - Descansar un momento antes de revisar el código - Iniciar el reloj (pausarlo cada vez que haya una interrupción) - Analizar la complejidad del código y el checklist de revisión, para decidir la mejor estrategia de revisión
2	Revisión	- Seguir el checklist de revisión de código - Revisar parte por parte - Dentro de cada parte, revisar categoría por categoría - Cada vez que se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos (Defect Log) - Marcar cada categoría al terminar de revisarla
3	Revisión de Correcciones	 Revisar que se hayan corregido todos los defectos encontrados Revisar todos los cambios realizados Si se encuentra un defecto, corregirlo y registrarlo en la bitácora de defectos y, si procede, llenar el campo "Fix Defect" del defecto que se estaba arreglando cuando se inyectó este nuevo defecto Detener el reloj y marcar la fase de revisión de código como terminada
Criterios de Salida		- Código del programa completamente revisado - Checklist de revisión marcado en forma completa de acuerdo a las partes que se revisaron - Todos los defectos encontrados fueron corregidos - Todos los cambios fueron revisados - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas

Script PSP2.1 para el Postmortem:

Propósito	Guiar el proceso de postmortem en PSP
Criterios de Entrada	-Documento de requerimientos completo y entendido
	 -Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos y con los estándares de codificación y contabilización -Process Dashboard con todos los datos recolectados al estar realizando el programa -Checklists de revisión de diseño y de revisión de código completos -Plantillas de diseño completas

#	Paso	Descripción
1	Registro de Defectos	 Iniciar el reloj (pausarlo cada vez que haya una interrupción) Revisar el Resumen del Plan (Project Plan Summary) para verificar que todos los defectos encontrados en cada fase están registrados Utilizando el mejor recuerdo, registrar en la bitácora de defectos (Defect Log) aquellos defectos que se hayan olvidado registrar
2	Consistencia de los datos de los defectos	 -Verificar que en la bitácora de defectos (Defect Log) los datos de cada defecto estén correctos y completos -Verificar que en el Resumen del Plan (Project Plan Summary) la cantidad de defectos inyectados y removidos por fase sean razonables y completos -Calcula el yield del proceso y verifica que el valor es razonable y correcto -Utilizando el mejor recuerdo, corregir cualquier dato que falte o esté incorrecto

Script PSP2.1 para el Postmortem: (continuación)

3	Tamaño	-Medir el tamaño real del programa -Determinar el tamaño de las partes base (LDC base, borradas, modificadas, agregadas), partes nuevas (cantidad de ítems y tamaño en LDC) y partes reusadas (LDC)Capturar estos tamaños en la Plantilla de Estimación de Tamaño (Size Estimating Template) -Calcular el Tamaño TOTAL del programa y capturarlo en el Resumen del Plan (Project Plan Summary)
4	Registro de tiempo	-Verificar la bitácora de tiempo (<i>Time Log</i>) buscando errores u omisiones -Utilizando el mejor recuerdo, corregir cualquier dato de tiempo que falte o esté incorrecto
5	Mejora de Proceso	 -Analizar las métricas de la forma Resumen del Plan para encontrar una o más áreas de mejora en el proceso personal -Documentar la(s) propuesta(s) de mejora en la forma Propuesta de Mejora de Proceso (Process Improvement Proposal) describiendo el (los) problema(s), la(s) sugerencia(s) de mejora y/o la(s) lección(es) aprendida(s) -Detener el reloj y marcar la fase de postmortem como terminada -Marcar el programa como terminado
Criterios de Salida		 - Programa que pasa exitosamente todas las pruebas unitarias y que cumple con los requerimientos y con los estándares de codificación y contabilización - Resumen del Plan (Project Plan Summary) lleno con los datos estimados y reales - Plantilla de Estimación de Tamaño (Size Estimating Template) completa - Bitácora de tiempo (Time Log) y de defectos (Defect Log) completas - Plantillas de diseño completas - Reporte de pruebas completo - Checklists de revisión de diseño y de revisión de código completos - Formato de Propuesta de Mejora de Proceso (Process Improvement Proposal) completo