

03063



UNIVERSIDAD NACIONAL AUTONOMA DE MEXICO

POSGRADO EN CIENCIA E INGENIERIA
DE LA COMPUTACION

1

APLICACION DE TSPi EN UN CURSO DE INGENIERIA
DE SOFTWARE NIVEL LICENCIATURA

T E S I S

QUE PARA OBTENER EL GRADO DE
MAESTRA EN CIENCIAS

PRESENTA:
CARMEN DOLORES ALVAREZ SANCHEZ

2299622

DIRECTORA DE LA TESIS: M. EN C. MA. GUADALUPE E. IBARGUENGOITIA GONZALEZ

MEXICO, D.F.

2001



Universidad Nacional
Autónoma de México

Dirección General de Bibliotecas de la UNAM

Biblioteca Central



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

Todo el material contenido en esta tesis esta protegido por la Ley Federal del Derecho de Autor (LFDA) de los Estados Unidos Mexicanos (México).

El uso de imágenes, fragmentos de videos, y demás material que sea objeto de protección de los derechos de autor, será exclusivamente para fines educativos e informativos y deberá citar la fuente donde la obtuvo mencionando el autor o autores. Cualquier uso distinto como el lucro, reproducción, edición o modificación, será perseguido y sancionado por el respectivo titular de los Derechos de Autor.

Dedicatoria

Dedico este trabajo a mi papi Sergio Alvarez y a mi hermano Sergio Anselmo, por confiar en mí, por lo que soy, y por lo que Dios me ha permitido lograr con ustedes a mi lado, **Mil Gracias Siempre.**

A la memoria de mi mami y mis abuelos por forjarme en este camino.

A mi hermana Alma Delia y a Lizbeth por ser parte de mi familia y por cada palabra de ánimo. Y a mi niña linda María Delia, por la alegría que ha llegado con ella a mi familia.

Agradecimientos

A Dios, por ser la luz que me guía y que me ha permitido la culminación de este trabajo.

A mi directora de tesis, la Maestra Lupita por el tiempo dedicado y el apoyo en el desarrollo de esta tesis.

A la Dra. Hanna Oktaba, por invitarme a vivir la aplicación de TSPI en el grupo de la Facultad de Ciencias y por el apoyo brindado.

Al M. en C. Gustavo Arturo Márquez Flores, al Dr. Fernando Gamboa Rodríguez y al Dr. Manuel Romero Salcedo por realizar la revisión de mi tesis y formar parte de mi jurado.

A la Universidad Nacional Autónoma de México y al CONACyT por permitirme realizar estos estudios y ser parte de este posgrado.

A cada una de las lindas personas que conocí en este tiempo: a Lulú, Violeta y Juanita por hacernos sentir como en casa y por esa gran paciencia y ánimo brindado siempre. A cada uno de mis compañeros porque es lindo haberlos conocido y haber convivido estos últimos años con ustedes.

En especial a mis amigos Mary Casas y Jaime Támez por esas palabras de esperanza, por el apoyo y ayuda incondicional en cada momento, mil gracias.

A José Manuel Martínez por su amor, paciencia y por compartir cada momento a mi lado.

Contenido

Introducción.....	1
PARTE I	
Presentación de TSPI	
Capítulo 1. Panorama sobre el Proceso de Software en Equipo.....	5
1.1. ¿Qué es TSPI?.....	5
1.2. Los Principios de TSPI.....	5
1.3. El diseño de TSPI.....	6
1.4. La estructura de TSPI y su flujo.....	7
1.5. El proceso de TSPI.....	9
Capítulo 2. La Lógica de TSPI.....	11
2.1. ¿Por qué los proyectos fallan?.....	11
2.2. Problemas comunes en los equipos.....	11
2.3. ¿Qué es un equipo?.....	11
2.4. Cómo TSPI construye equipos.....	12
Capítulo 3. Lanzamiento.....	13
3.1. ¿Por qué es necesario realizar un lanzamiento de equipo?.....	13
3.2. Objetivos del equipo.....	13
3.3. Guiones de lanzamiento de TSPI.....	15
Capítulo 4. Estrategia de Desarrollo.....	19
4.1. La estrategia de TSPI.....	19
4.2. El diseño conceptual.....	19
4.3. Administración de riesgos.....	19
4.4. Una estrategia de reuso.....	19
4.5. Los guiones de Estrategia.....	19
Capítulo 5. El Plan de Desarrollo.....	23
5.1. La necesidad de planear.....	23
5.2. Los guiones del Plan de Desarrollo.....	24
5.3. Seguimiento del trabajo.....	26
5.4. Plan de calidad.....	27
Capítulo 6. Definiendo los Requerimientos.....	35
6.1. ¿Qué son los requerimientos?.....	35
6.2. Necesidad de los Requerimientos.....	35
6.3. Cambios en los requerimientos.....	35
6.4. La Especificación de Requerimientos de Software (ERS).....	36
6.5. Los Guiones de Requerimientos de TSPI.....	37
Capítulo 7. Diseñando en equipo.....	43
7.1. Principios de diseño.....	43
7.2. Diseñando en equipo.....	43
7.3. Estándares de diseño.....	44
7.4. Diseñando para reuso.....	46
7.5. Diseñando para usabilidad.....	47
7.6. Diseñando para pruebas.....	47
7.7. Revisiones e inspecciones en el diseño.....	47
7.8. Los guiones de diseño de TSPI.....	47

Capítulo 8. Implementación del Producto.....	53
8.1. Criterio de diseño terminado.....	53
8.2. Estándares de implementación.....	53
8.3. La estrategia de implementación.....	55
8.4. Revisiones e inspecciones.....	56
8.5. Los guiones de Implementación.....	57
Capítulo 9. Pruebas de Integración y del Sistema.....	63
9.1. Principios de las pruebas.....	63
9.2. Estrategia de pruebas de TSPI.....	63
9.3. La estrategia de construcción e integración.....	63
9.4. La estrategia de pruebas del sistema.....	64
9.5. Planeación de las pruebas.....	65
9.6. Seguimiento y mediciones en las pruebas.....	65
9.7. Documentación.....	66
9.8. Los guiones de pruebas de TSPI.....	67
Capítulo 10. El Postmortem.....	73
10.1. Propósito de Mejora del Proceso.....	73
10.2. Los guiones de TSPI.....	73
Capítulo 11. Roles de Equipo de TSPI.....	79
11.1. Rol: Líder de Equipo.....	79
11.2. Rol: Administrador de Planeación.....	84
11.3. Rol: Administrador de Desarrollo.....	88
11.4. Rol: Administrador de Calidad y Proceso.....	91
11.5. Rol: Administrador de Apoyo.....	93
PARTE II	
Aplicación de TSPI al curso de Ingeniería de Software	
Capítulo 12. Aplicación de TSPI.....	97
12.1. Antecedentes Académicos.....	97
12.2. Método de trabajo.....	99
12.3. Reportes semanales de las actividades desempeñadas por los equipos y el Instructor.....	103
12.4. Aspectos Sociales, Técnicos y del Proceso de los equipos.....	115
12.5. Reporte Postmortem.....	120
12.6. Evaluación de los equipos.....	125
12.7. Evaluación final.....	131
Conclusiones.....	135
Referencias bibliográficas.....	137
Anexos	
Anexo A. Definición de Abreviaturas y Términos.....	139
Anexo B. Análisis del Proceso, Revisiones e Inspecciones.....	145
Anexo C. Formas y guiones.....	155
Anexo D. Encuesta al grupo de alumnos de la materia Ingeniería de Software 2000-I.....	173
Lista de Figuras	
Figura 1.1 Representación gráfica de la estructura de TSPI con un diagrama de clases.....	8
Figura 1.2 El flujo de TSPI.....	8
Figura 8.1 Mapa de Calidad de componentes.....	59
Figura 12.1 Número de alumnos que cursaron las materias.....	98

Figura 12.2 Representación del conocimiento de los alumnos sobre las fases.....	116
Figura 12.3 Concentrado numérico sobre conocimientos adquiridos en cada área.....	117
Figura 12.4 Áreas de Ingeniería de Software aplicadas en el curso.....	118
Figura 12.5 Evaluación del nivel de inglés para asimilar el material bibliográfico.....	120
Figura B.1 Trabajo requerido durante el primer ciclo.....	149
Figura B.2 Dificultad en cada rol durante el primer ciclo.....	149
Figura B.3 Trabajo requerido durante el segundo ciclo.....	149
Figura B.4 Dificultad en cada rol durante el segundo ciclo.....	150
Figura B.5 Porcentaje de criterios durante el primer ciclo.....	150
Figura B.6 Porcentaje de criterios durante el segundo ciclo.....	151
Figura B.7 Porcentaje de contribución durante el primer ciclo.....	151
Figura B.8 Porcentaje de contribución durante el segundo ciclo.....	151
Figura B.9 Porcentaje de Apoyo y Ayuda durante el primer ciclo.....	152
Figura B.10 Porcentaje de Apoyo y Ayuda durante el segundo ciclo.....	152
Figura B.11 Porcentaje del Desempeño durante el primer ciclo.....	153
Figura B.12 Porcentaje del Desempeño durante el segundo ciclo.....	153

Lista de Formas

Forma C.1 Forma INFO.....	156
Forma C.2 Forma SEMANA.....	159
Forma C.3 Forma ESTRA.....	160
Forma C.4 Forma RESTM.....	161
Forma C.5 Forma para el Plan de Calidad, RESCA.....	164
Forma C.6 Forma TAREA.....	165
Forma C.7 Forma CALENDARIO.....	166
Forma C.8 Forma para reporte de inspección, REPINS.....	167
Forma C.9 Forma Registro de Defectos, REGD.....	168
Forma C.10 Forma Registro de Tiempo, REGT.....	169
Forma C.11 Forma Evaluación del Equipo por Colegas, EEC.....	170

Lista de Tablas

Tabla 1.1 Guión de Desarrollo de TSPi, DES.....	10
Tabla 3.1 Estándar de roles y sus responsabilidades.....	13
Tabla 3.2 Guión de Lanzamiento para el primer ciclo, LAN1.....	16
Tabla 3.3 Guión de Lanzamiento para ciclos posteriores, LANn.....	17
Tabla 4.1 Guión para la estrategia de desarrollo del primer ciclo, ESTRA1.....	20
Tabla 4.2 Guión para la estrategia de desarrollo de ciclos posteriores, ESTRAn.....	21
Tabla 5.1 Guión para el plan de desarrollo para el primer ciclo, PLAN1.....	30
Tabla 5.2 Guión para el plan de desarrollo para ciclos posteriores, PLANn.....	31
Tabla 5.3 Tabla de instrucciones de la forma RESTM.....	32
Tabla 5.4 Tabla de instrucciones de la forma RESPL.....	33
Tabla 6.1 Tabla del contenido de la ERS.....	37
Tabla 6.2 Guión de desarrollo de requisitos para el primer ciclo, REQ1.....	40
Tabla 6.3 Guión de desarrollo de requisitos para ciclos posteriores, REQn.....	41
Tabla 7.1 Estándares de tipos de defectos de PSP.....	45
Tabla 7.2 Guión de diseño para el primer ciclo, DIS1.....	50
Tabla 7.3 Guión de diseño para ciclos subsecuentes, DISn.....	51
Tabla 8.1 Guión de Implementación para el primer ciclo, IMP1.....	61
Tabla 8.2 Guión de Implementación para ciclos posteriores, IMPn.....	62
Tabla 9.1 Guión de Pruebas de Integración y del Sistema para el primer ciclo, PRUEBA1.....	70
Tabla 9.2 Guión de Pruebas de Integración y del Sistema para ciclos posteriores, PRUEBAn.....	71
Tabla 10.1 Guión de Postmortem para el primer ciclo, PM1.....	77
Tabla 10.2 Guión de Postmortem para ciclos posteriores, PMn.....	78
Tabla 11.1 Guión del rol Líder de Equipo.....	84
Tabla 11.2 Guión del rol Administrador de Planeación.....	87
Tabla 11.3 Guión del rol Administrador de Desarrollo.....	91
Tabla 11.4 Guión del rol Administrador de Calidad y Proceso.....	93
Tabla 11.5 Guión del rol Administrador de Apoyo.....	96
Tabla 12.1 Materias y los semestres en que se deben cursar.....	97
Tabla 12.2 Claves de las Materias.....	98

Tabla 12.3 Guión inicial de Desarrollo de TSPi del Instructor, DES.....	100
Tabla 12.4 Guión final de Desarrollo de TSPi del Instructor, DES.....	102
Tabla 12.5 Documentos de la tercera semana.....	104
Tabla 12.6 Documentos de la quinta semana.....	105
Tabla 12.7 Documentos de la segunda reunión en la quinta semana.....	105
Tabla 12.8 Documentos de la sexta semana.....	106
Tabla 12.9 Documentos de la octava semana.....	107
Tabla 12.10 Evaluación del trabajo requerido durante el primer ciclo.....	126
Tabla 12.11 Evaluación de dificultad en cada rol durante el primer ciclo.....	126
Tabla 12.12 Evaluación del trabajo requerido durante el segundo ciclo.....	127
Tabla 12.13 Evaluación de dificultad en cada rol durante el segundo ciclo.....	127
Tabla 12.14 Evaluación de criterios durante el primer ciclo.....	128
Tabla 12.15 Evaluación de criterios durante el segundo ciclo.....	128
Tabla 12.16 Evaluación sobre la contribución de cada rol durante el primer ciclo.....	129
Tabla 12.17 Evaluación sobre la contribución de cada rol durante el segundo ciclo.....	129
Tabla 12.18 Evaluación del apoyo y ayuda durante el primer ciclo.....	129
Tabla 12.19 Evaluación del apoyo y ayuda durante el segundo ciclo.....	130
Tabla 12.20 Desempeño por rol durante el primer ciclo.....	130
Tabla 12.21 Desempeño por rol durante el segundo ciclo.....	130
Tabla B.1 Integrantes de los equipos en los dos ciclos.....	145
Tabla B.2 Problemas y Mejoras durante el primer ciclo.....	147
Tabla B.3 Análisis de formas de planeación.....	148
Tabla B.4 Datos estimados y reales.....	148
Tabla C.1 Tabla de Instrucciones para llenar la forma INFO.....	155
Tabla C.2 Guión para efectuar las reuniones semanales.....	157
Tabla C.3 Tabla de instrucciones para llenar la forma SEMANA.....	158
Tabla C.4 Tabla de instrucciones para llenar la forma ESTRA.....	160
Tabla C.5 Criterio estándar de calidad.....	162
Tabla C.6 Tabla de instrucciones para el registro de defectos en la forma REGD.....	167
Tabla C.7 Tabla de instrucciones para el registro de tiempo en la forma REGT.....	169
Tabla C.8 Tabla de instrucciones de la forma Evaluación del Equipo por Colegas, EEC.....	171

Introducción

El auge del desarrollo de software ha provocado generar sistemas cuyos resultados finales no han sido completamente satisfactorios al cliente. Ciertas veces el producto obtenido nunca se ha puesto en aplicación, por no cumplir los requisitos de quién lo solicitó o porque continuamente se le han estado realizando modificaciones que van surgiendo conforme transcurre el tiempo. Tales consecuencias, en ocasiones han sido generadas porque no existe una planeación clara sobre las actividades que se deben realizar y esas son desempeñadas en un ambiente sin ningún tipo de organización.

Debido a la necesidad de que los estudiantes del área de computación adquieran conocimientos y experiencia en el área de la Ingeniería de Software y puedan ingresar a la industria como ingenieros de software mejor preparados, se tomó la decisión de aplicar TSPi a un grupo de estudiantes mexicanos de la licenciatura en Ciencias de la Computación.

Se decidió utilizar TSPi (Proceso de Software en Equipo a nivel introductorio) porque a través de su aplicación se pueden cumplir los principios que CMM y PSP proponen para guiar a los ingenieros en el desarrollo de sus actividades y que en conjunto realicen un buen trabajo en equipo. La creación de estos procesos y modelo fue un esfuerzo del Software Engineering Institute (SEI).

CMM es un modelo de madurez de capacidad. Describe los elementos clave de un proceso de software eficaz y fue desarrollado por petición del Departamento de Defensa de los Estados Unidos para evaluar la capacidad de sus contratistas de software. Está constituido por cinco niveles de madurez, los cuales en conjunto integran dieciocho áreas clave, tales niveles deben ser alcanzados ascendentemente por una organización. Aunque CMM proporciona un marco de mejora, se enfoca a lo que las organizaciones deberían hacer y no indica cómo lo deberían hacer. Cuando en la industria se empezó a utilizar CMM, se generaron problemas al momento de aplicar los principios que el modelo indica. Por petición de pequeños grupos, en ocasiones, se tenía que dirigir a los ingenieros en las actividades que debían realizar, pero ellos preguntaban cómo las debían realizar. Tales peticiones requirieron un proceso de enseñanza detallado, en el cual debían integrarse prácticas de trabajo reales de los ingenieros de software.

PSP es un proceso personal de software, el cual integra métodos y prácticas de trabajo que deben realizar los ingenieros de software individualmente. PSP se utiliza como guía para realizar el trabajo personal. Al aplicar este proceso, surgió la necesidad de que la organización brindará gran apoyo a los ingenieros que lo estaban utilizando, por eso, se aconsejó que el área administrativa mantuviera un control y seguimiento sobre el desempeño de quienes estaban aplicándolo, la razón consistió en que los ingenieros encontraban dificultad o no deseaban seguir un método de trabajo disciplinado, y de nuevo aplicaban los antiguos hábitos de trabajo.

Pero aún, cuando los ingenieros estaban utilizando PSP y se mantenía un seguimiento y control sobre su desempeño, se identificó la necesidad de saber cómo combinar los procesos personales de todos los integrantes de un equipo. Aún este problema también se identificó en los niveles más altos de CMM. Por tales razones, se inició el desarrollo de TSP.

El Proceso de Software en Equipo (TSP) indica como construir un equipo, lo guía en el lanzamiento de los proyectos, en la planeación y administración de su trabajo e indica como ser un buen integrante del mismo; lo más importante, es que indica al área administrativa cómo guiar y apoyar a cada equipo y cómo mantener un ambiente de trabajo en el cual se promueva un alto desempeño en el equipo.

TSP tiene como objetivos construir equipos dirigidos por sí mismos, que tengan la capacidad de planear y de mantener un seguimiento del propio trabajo, de establecer objetivos, procesos y planes propios; de mostrar a los administradores la manera en la que deberán administrar, motivar y ayudar a los equipos a lograr un alto desempeño; a acelerar la mejora en el proceso de software, utilizándose como guía para lograr una alta madurez y llegar al nivel cinco de CMM; y a facilitar la enseñanza en las universidades, para que los estudiantes desarrollen destrezas de trabajo en equipo, las cuales puedan aplicarse en el área de la industria.

Retomando el último objetivo, se hace la referencia de que TSP está introduciéndose en los ambientes industrial y académico, por tal razón, se realizó la introducción a TSP, llamada TSPi, la cual integra los principios de TSP y por este motivo se generó un libro de texto que se ha puesto a disponibilidad de las universidades para propagar la enseñanza del proceso mediante el trabajo en equipo, el cual pueda utilizarse en el desarrollo de productos software.

Por lo anteriormente mencionado, se decidió aplicar TSPi a un grupo de estudiantes mexicanos. Por lo tanto, los objetivos de esta tesis, consisten en presentar la traducción de los capítulos referentes al libro de texto, las formas y guiones de uso más común para la utilización en cursos posteriores, con el propósito de apoyar a estudiantes de habla hispana; recopilar la experiencia de la aplicación de TSPi, realizando un análisis sobre la misma, mediante la cual sea posible identificar los aciertos, ventajas y dificultades del proceso; y proporcionar una guía para realizar un análisis de la aplicación de TSPi en cursos posteriores, mediante las diversas secciones que integran la segunda parte.

TSPi es un proceso a nivel introductorio para desarrollar software en equipo, en el mismo se propone que un producto se genere en un monto de tiempo pequeño, en dos o tres ciclos. Este proceso proporciona un conjunto estructurado de pasos, indica qué se debe hacer en cada uno y cómo relacionarlos para obtener un producto final. Propone las fases que se deben seguir y en cada una presenta técnicas para desarrollar las actividades requeridas. Propone métricas y establece métodos de revisiones e inspecciones para asegurar la calidad de los productos generados en cada ciclo, y por lo tanto, la calidad del producto final. Establece diversos roles que deberán desempeñar los estudiantes en los distintos ciclos. Y proporciona información para lograr ser un integrante de equipo efectivo.

El trabajo que a continuación se presenta está integrado de dos partes, las cuales se definen brevemente. La Primera Parte, **Presentación de TSPi**, consiste en presentar un resumen de los capítulos que incluye la bibliografía de TSPi. Los capítulos del 1 hasta el 10 se refieren a las fases del proceso, y el capítulo 11 contiene la información correspondiente a los roles propuestos. En la cada capítulo se presentan los guiones correspondientes, los cuales se refieren al primero y posteriores ciclos de desarrollo; se presenta una breve explicación sobre los puntos más importantes del capítulo y se hace referencia a las formas utilizadas.

La Segunda Parte, **Aplicación de TSPi al curso de Ingeniería de Software**, presenta diversos análisis generados a partir de las actividades desarrolladas por los equipos de trabajo en el curso de Ingeniería de Software. Se presentan los antecedentes académicos de los alumnos que participaron; el método de trabajo que se siguió para el desarrollo del proceso; los acontecimientos semanales desde el inicio hasta el final del período escolar, en cada uno de ellos se muestra el seguimiento de las actividades de los alumnos y del Instructor, indicando los problemas y otros asuntos acontecidos. También se presentan análisis de diversos rubros; iniciando con los aspectos sociales, técnicos y del proceso que intervinieron en el curso y que se generaron a partir de una encuesta aplicada a los alumnos; se presenta el análisis del reporte de Postmortem, el cual consiste en evaluar el trabajo desarrollado en cada ciclo y detectar las posibles mejoras que se pudieran aplicar en el ciclo siguiente; un tercer análisis se refiere a la evaluación aplicada a los alumnos mediante la forma Evaluación del Equipo por Colegas (EEC), y por último, se presenta el análisis referente a la evaluación final, la cual se generó a partir de las secciones de esta parte.

Posteriormente se presentan las conclusiones del presente trabajo, las referencias utilizadas y los anexos. En el Anexo A, **Definición de Abreviaturas y Términos**, se incluyen los términos y abreviaturas utilizados en el mismo, se indican los nombres asignados para hacer referencia a los términos en inglés, manejados en el libro de texto. En el Anexo B, **Análisis del Proceso, Revisiones e Inspecciones**, se incluye información relacionada con el análisis de la segunda parte, integrando gráficas que representan los valores obtenidos a partir de la evaluación de los equipos, mediante la forma EEC y la Encuesta al Grupo de Alumnos de la Materia Ingeniería de Software 2000-I. En el Anexo C, **Formas y Guiones**, se integran las formas y guiones de uso común. Es importante mencionar que no se incluyeron todas las formas y guiones a los que se hace referencia en el libro de TSPi; solamente se encuentran aquellas de uso básico. Y en el Anexo D, se presenta la encuesta aplicada a los alumnos del curso.

Parte I
PRESENTACIÓN DE TSPi

Capítulo 1. Panorama sobre el Proceso de Software en Equipo

Objetivo: Describir el proceso TSPi, las razones de su diseño y estructura para desarrollar un proyecto iterativo e incremental.

1.1. ¿Qué es TSPi?

El Proceso de Software en Equipo, a nivel introductorio (TSPi) es un marco de trabajo definido para cursos de Ingeniería de Software realizados en equipo, a nivel de maestría o para los últimos semestres de licenciatura.

Se presenta como una guía de los pasos que deben seguir, porque indica cómo aplicar el conocimiento de la ingeniería de software y los principios del proceso en un ambiente de equipo de trabajo. También define los roles para cada uno de los integrantes. Está diseñado con base en TSP, el cual es un proceso industrial para equipos, integrado aproximadamente por 20 ingenieros o más, quienes desarrollan sistemas intensivos de software a gran escala, se diseñó para aplicarlo a proyectos, cuya terminación frecuentemente, consiste en varios años. Por lo tanto, TSPi es una versión a escala reducida de TSP y contiene los mismos conceptos y métodos básicos.

Necesidad de un proceso en la Ingeniería

Construir un equipo no es nada fácil. Con frecuencia los integrantes pierden demasiado tiempo en establecer un mecanismo de trabajo que solucione las diferentes necesidades, las cuales se clasifican en los siguientes tipos: *laborales*, *afectivas* y *de apoyo*.

Las *necesidades laborales* indican como trabajar juntos, definir funciones, establecer una estrategia para hacer el trabajo, coordinar las tareas y reportar el progreso. Las *necesidades afectivas*, indican como construir y mantener relaciones de trabajo efectivas, las cuales requieren de objetivos en común, un plan de acción y una dirección apropiada. Las *necesidades de apoyo*, indican como entender las habilidades y debilidades de los demás, apoyar a los compañeros del equipo y ser capaces de pedir ayuda cuando sea necesario.

TSPi mejora la productividad; aunque en un principio, la planeación y los pasos para formar el equipo parezcan tomar gran tiempo, son una parte esencial para hacer un proyecto en equipo.

1.2. Los Principios de TSPi

TSPi se basa en los siguientes cuatro principios básicos:

1. Tener como guía un proceso definido. Con ello se adquiere un aprendizaje más efectivo y una retroalimentación rápida. Eso es posible, con el marco de trabajo definido, medible y repetible de los guiones y formas del proceso. Y también, con los resultados de las evaluaciones realizadas en cada ciclo.
2. Establecer una combinación de objetivos específicos, un buen ambiente de trabajo, una guía y un líder capaz de lograr un equipo de trabajo productivo.
3. Definir roles, prácticas y métodos de desarrollo, para generar beneficios necesarios al enfrentarse con problemas del proyecto actual y que permitan obtener soluciones efectivas.
4. Construir la instrucción sobre un esquema disponible de conocimiento anterior, para hacerla más efectiva. TSPi se basa en este aspecto por la experiencia obtenida a partir del trabajo con equipos de software y en cursos académicos.

1.3. El diseño de TSPi

A continuación se describen las siete decisiones de diseño.

1.3.1. Proporcione un marco de trabajo simple que se construya sobre la base de PSP

El propósito de un proceso es ayudar a hacer una tarea como desarrollar un producto o aprender a hacer un proyecto en equipo. Aunque TSPi proporciona formas y guiones, la mayoría son similares a los que se utilizan en PSP¹. Haber sido entrenado anteriormente en ese proceso, facilita la comprensión de TSPi. Es por eso, que un entrenamiento previo en PSP es un requisito para TSPi.

Con la asignación de roles, TSPi permite enfocarse en los elementos que personalmente se deberán aprender. Esos roles especifican quien hará cada una de las tareas de planeación, seguimiento, calidad, apoyo y liderazgo. También reducen el monto de material que se debe de entender como un todo completo. Cada integrante debe concentrarse en las tareas específicas de acuerdo a su rol y no preocuparse por las responsabilidades de los demás, por lo menos, no aún.

Durante un proyecto multiciclo de TSPi, se aconseja obtener experiencia desempeñando varios roles. Eso proporcionará un amplio conocimiento en el proceso y permitirá tener un entendimiento más profundo sobre los aspectos del desarrollo de software en equipo. También ayudará a obtener una mejor apreciación de las habilidades e intereses personales.

1.3.2. Desarrolle productos en varios ciclos

En un curso completo de TSPi, se realizarán dos o tres ciclos de desarrollo. En cada uno se definen los requerimientos, diseño, implementación y un proceso de pruebas.

En el primer ciclo, se construye un subconjunto con funciones mínimas del producto final. El segundo ciclo se construye con base en los resultados del primer ciclo; entonces se pueden cambiar los roles del equipo, ajustar el proceso o aplicar más disciplina en los métodos de calidad. Si existiera tiempo se podrían repetir esos ciclos por tercera y cuarta vez.

Con el desarrollo de varios ciclos se tendrá una evidencia clara y convincente de que es mejor aplicar en el equipo y así se tendrá la confianza de continuar usando esos métodos en la práctica.

1.3.3. Establezca medidas estándares para la calidad y desempeño

Las medidas son una parte esencial para hacer un trabajo de alta calidad. PSP proporciona las medidas básicas y la medición de destrezas que se necesitan para medir y evaluar la calidad de trabajo de cada integrante del equipo. TSPi indica cómo interpretarlas y aplicarlas a un proyecto.

El énfasis que hace TSPi sobre los objetivos y medidas ayuda a conocer los beneficios de las medidas de calidad y el valor del proyecto en la planeación y seguimiento.

1.3.4. Proporcione medidas precisas para los equipos y los estudiantes

Con las medidas de TSPi, el desempeño de cada uno de los integrantes del equipo será obvio. Aunque el propósito principal de esas medidas, es ayudar a hacer un mejor trabajo, cada uno conocerá lo que los demás están haciendo. Por esto, si alguien no desarrolla sus actividades adecuadamente tendrá comentarios de los demás integrantes del equipo.

¹ Personal Software Process es un proceso que tiene como objetivo controlar las actividades para administrar un proyecto y generar un producto de alta calidad, mediante la disciplina en administración y registro de tiempo y conteo de defectos. Básicamente consiste en el registro de la siguiente información: estimación del tiempo de actividades similares, planeación y administración de tiempo, acuerdos establecidos, generación de planes y asignación de tiempo del proyecto, estimación del tamaño, calidad de los productos, establecimiento y evaluación de mejoras para las actividades.

1.3.5. Aplique evaluaciones de rol y de equipo

TSPi proporciona la forma Evaluación del Equipo por Colegas (Forma C.11). El objetivo es evaluar el desempeño de cada rol y el proceso de trabajo. Lo cual permitirá a cada integrante tener conocimientos sobre esa información.

1.3.6. Requiera disciplina en el proceso

Porque la Ingeniería de Software no implica un desempeño personal disciplinado, debido a que el proceso de software no impone una disciplina natural en los ingenieros y porque un trabajo disciplinado requiere conseguir altos estándares y un apoyo competente, se aconseja al instructor de TSPi seguir el proceso y reunir los datos de las formas para un análisis posterior. De lo contrario, no se cosecharán los beneficios completos de TSPi.

1.3.7. Proporcione una guía en los problemas del trabajo en equipo

Ocasionalmente, existen problemas entre los integrantes de un equipo, ya sea, por ser personas sin experiencia o por su propia personalidad. Sin embargo, con guía y apoyo, la mayoría de los ingenieros pueden ser integrantes efectivos. La fuerza más poderosa para resolver los problemas del equipo es la presión por un par. La mayoría de las personas están interesadas en conocer la opinión de su par y ansían ser respetadas e incluidas en un grupo. Dado un tiempo y con un guía propio, se puede aprender a ser integrante completamente efectivo de un equipo.

1.4. La estructura de TSPi y su flujo

La figura 1.1 muestra la estructura de TSPi. El proceso está constituido por múltiples ciclos de desarrollo. En cada uno se definen los objetivos del producto que se cumplirán. Posteriormente, en cada ciclo se desarrollan las siete fases: Estrategia, Planeación, Requerimientos, Diseño, Implementación, Pruebas y Postmortem. La estructura de TSPi está siendo modelada con un diagrama de clases², donde se utiliza el símbolo de agregación³, representado por una línea dirigida con punta en forma de rombo, y cuyo significado corresponde a las frases "está constituido" y "está integrada".

² Diagrama que muestra un conjunto de clases y sus relaciones.

³ El símbolo de agregación forma parte de los diagramas de clases. Modela una relación "Todo/Parte", en donde la clase señalada por el rombo representa un elemento más grande (todo), la cual está formada por elementos más pequeños (partes) que están representadas por la clase que está en el otro extremo de la línea dirigida.

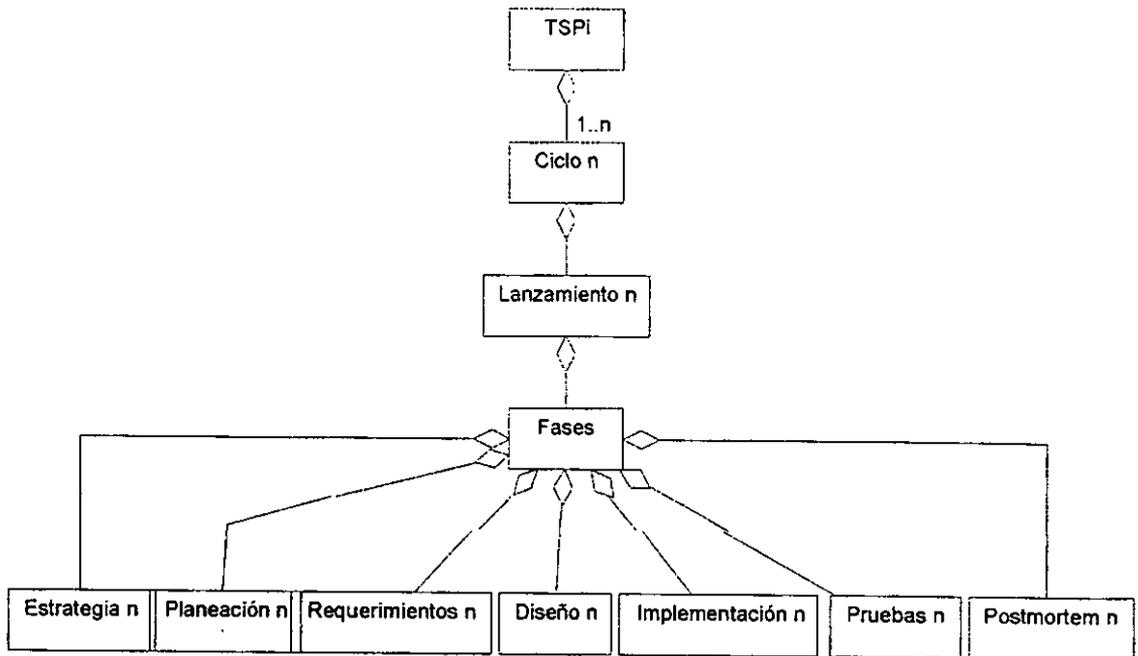


Figura 1.1 Representación gráfica de la estructura de TSPi con un diagrama de clases

La siguiente figura muestra el flujo de TSPi.

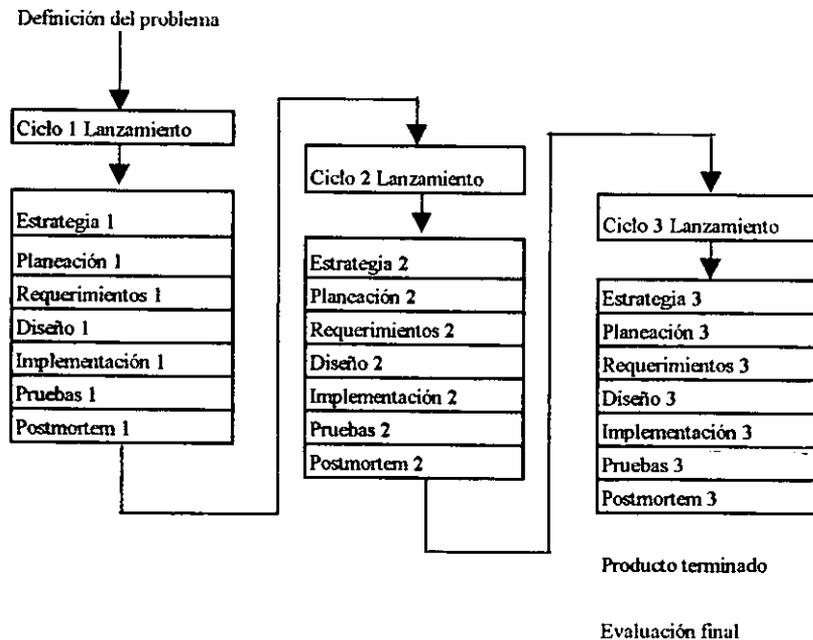


Figura 1.2 El flujo de TSPi

Al inicio se realiza la definición del problema que el cliente está requiriendo. Se describen los objetivos del producto y los equipos generan la estrategia de desarrollo. Se aconseja que en la primera versión del producto se defina un número menor de funcionalidades para el primer ciclo. Después el equipo estimará el tamaño de las mismas y decidirá cuáles integrar en cada ciclo. En el segundo ciclo, los ingenieros repiten los mismos pasos, pero esta vez, el producto base del ciclo anterior se utiliza para generar un producto con incremento de funcionalidades. Si se tuviese más tiempo, se podrían desarrollar ciclos subsecuentes.

1.5. El proceso de TSPi

Para la aplicación del proceso, TSPi utiliza un conjunto de guiones en los cuales se definen y describen las actividades que se deben desarrollar en cada fase del proceso.

Cada guión inicia con una descripción del objetivo. Contiene un criterio de entrada y de éxito, donde se especifica lo que se necesita tener antes de iniciar el guión y lo que se debería de terminar. La sección general proporciona información general acerca del guión.

La columna de la izquierda contiene un número que indica el orden de los pasos del guión, así como la semana correspondiente de acuerdo al calendario asignado. La segunda columna describe los nombres de las actividades (en la tabla 1.1 se indican las fases del proyecto), y la tercera columna describe textualmente las actividades a desarrollar en cada paso.

A continuación se muestra el guión de desarrollo, en el cual se indica el flujo completo de TSPi durante el proyecto completo.

Objetivo		Guiar al equipo en el desarrollo de un producto de software
Criterio de entrada		<ul style="list-style-type: none"> El instructor guía y apoya a uno o más equipos de 5 estudiantes. Los estudiantes tienen el entrenamiento en PSP. El instructor tiene materiales, facilidades y recursos para apoyar a los equipos. <p>El instructor ha guiado a los alumnos en la definición de los requerimientos generales del producto.</p>
General		<p>El proceso de TSPi está diseñado para:</p> <ol style="list-style-type: none"> Desarrollar el producto de software de tamaño pequeño a mediano en 2 o 3 ciclos de desarrollo. Desarrollar un producto más pequeño en cada ciclo. Producir los elementos del producto (documento de requerimientos, especificación de diseño, plan de pruebas, etc.) en cada ciclo. <p>Siga los guiones que se apliquen a su proyecto y modo de operación.</p>
Semana	Paso	Actividades
1	Revisión	<ul style="list-style-type: none"> Introducción al curso y revisión de PSP. Lea los capítulos 1 y 2 del libro (Panorama y Lógica sobre TSPi, (respectivamente) y el Apéndice A (del libro) o prepare el documento de requerimientos validado por el cliente.
2	LAN1	<ul style="list-style-type: none"> Se revisan los objetivos del curso y se asignan los equipos de estudiantes y los roles. Se lee el capítulo 3 del libro (Lanzamiento), el Apéndice B y uno de los capítulos 11-15 del libro.
	ESTRA1	<ul style="list-style-type: none"> Se produce el diseño conceptual, se establece la estrategia de desarrollo, se hacen estimaciones del tamaño y se evalúan riesgos. Se lee el capítulo 4 (Estrategia).
3	PLAN1	<ul style="list-style-type: none"> Se producen planes de equipo y de los ingenieros para el ciclo 1. Se lee el capítulo 5 (Plan de Desarrollo) y el Apéndice C (del libro).
4	REQ1	<ul style="list-style-type: none"> Se definen e inspeccionan los requerimientos para el ciclo 1. Se produce el plan de pruebas del sistema y el material de apoyo. Se lee el capítulo 6 (Requerimientos) y las secciones de pruebas del capítulo 9 (Pruebas).
5	DIS1	<ul style="list-style-type: none"> Se produce e inspecciona el diseño de alto nivel para el ciclo 1. Se produce el plan de pruebas de integración y el material de apoyo. Se lee el capítulo 7 (Diseño).

6	IMP1	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 1. • Se produce el plan de pruebas unitarias y el material de apoyo. • Se lee el capítulo 8 (Implementación).
7	PRUEBAS1	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 1. • Se produce la documentación para el usuario para el ciclo 1 • Se lee el capítulo 9 (Pruebas).
8	PM1	<ul style="list-style-type: none"> • Se dirige la evaluación postmortem y se escribe el reporte final del ciclo. • Se producen las evaluaciones de roles y de equipos para el ciclo 1. • Se leen los capítulos 10 (Postmortem), 16, 17 y 18 (estos tres últimos están en el libro).
	LAN2	<ul style="list-style-type: none"> • Se reorganizan equipos y roles para el ciclo 2. • Se leen los capítulos del 11 al 15 (Roles - del libro).
	ESTRA2, PLAN2	<ul style="list-style-type: none"> • Se produce la estrategia y el plan para el ciclo 2. • Se evalúan riesgos.
9	REQ2	<ul style="list-style-type: none"> • Se ajustan los requerimientos y el plan de pruebas del sistema para el ciclo 2.
	DIS2	<ul style="list-style-type: none"> • Se produce e inspecciona el diseño de alto nivel para el ciclo 2. • Se actualiza el plan de pruebas de integración del ciclo 2.
10	IMP2	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 2, se produce el plan de pruebas unitarias.
	PRUEBAS2	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 2. • Se produce la documentación de usuario para el ciclo 2.
11	PM2	<ul style="list-style-type: none"> • Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 2. • Se producen las evaluaciones de roles y de equipos para el ciclo 2.
	LAN3	<ul style="list-style-type: none"> • Se reorganizan los equipos y roles para el ciclo 3.
	ESTRA3, PLAN3	<ul style="list-style-type: none"> • Se produce la estrategia y los planes para el ciclo 3. • Se evalúan riesgos.
12	REQ3	<ul style="list-style-type: none"> • Se ajustan los requerimientos y el plan de pruebas del sistema para el ciclo 3.
	DIS3	<ul style="list-style-type: none"> • Se produce e inspecciona el diseño de alto nivel para el ciclo 3. • Se actualiza el plan de pruebas de integración del ciclo 3.
13	IMP3	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 3, se produce el plan de pruebas unitarias.
	PRUEBAS3	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 3.
14	PRUEBAS3	<ul style="list-style-type: none"> • Se produce la documentación de usuario para el producto final. • Se revisa y actualiza el manual de usuario con respecto a su utilidad y exactitud.
15	PM3	<ul style="list-style-type: none"> • Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 3. • Se produce la evaluación de roles y de equipos para el ciclo 3. • Se revisan los productos generados y el proceso utilizado. • Se identifican las lecciones aprendidas y propuestas para la mejora de proceso.
Criterio de éxito		<ul style="list-style-type: none"> • Haber terminado un producto o parte del mismo con documentación para el usuario. • La carpeta de proyecto completa y actualizada. • Evaluaciones de equipos y los reportes de ciclos documentados.

Tabla 1.1 Guión de Desarrollo de TSPi, DES

Capítulo 2. La Lógica de TSPi

Objetivo: Definir los equipos, explicar cómo trabajan e indicar algunos problemas comunes.

2.1. ¿Por qué los proyectos fallan?

Un problema significativo de las personas que trabajan en los equipos de software es la incapacidad para manejar la presión del tiempo. Los equipos necesitan saber cómo trabajar eficientemente y generar productos de calidad, especialmente cuando están bajo un horario intenso. Al guiarlos a través de un proceso de planeación y una estrategia; TSPi les indica cómo manejar la presión. Para hacer el trabajo, establecen una estrategia, estiman el tamaño de los productos que construirán y realizan un plan.

2.2. Problemas comunes en los equipos

- Liderazgo inefectivo, provoca que los equipos tengan problemas para apegarse a los planes y mantener una disciplina personal.
- Falta de cooperación entre los integrantes que no muestran voluntad para trabajar cooperativamente con el equipo.
- Diferente grado de desempeño en el equipo, sucede cuando los integrantes tienen diferentes destrezas y habilidades, así como diferentes motivaciones y energía.
- Incumplimiento en los tiempos de entrega y metas concretas, así como establecimientos de tiempos que nunca se cumplen; todo eso es consecuencia de tener asignado un líder sin experiencia, carencia de objetivos claros, de un proceso o plan definido.
- La falta de calidad en los productos es consecuencia de inspecciones superficiales en los requerimientos, un diseño documentado pobremente, o prácticas de implementación mal hechas.
- Generalmente, durante las fases de diseño o implementación los ingenieros piensan en mejorar los productos. Cuando esto sucede la razón consiste en que no hubo una definición clara de los requerimientos del cliente, y eso se puede confundir entre las adiciones reales y modificaciones por parte de los ingenieros
- Evaluación irreal entre los integrantes del equipo al rechazar evaluar el desempeño de sus compañeros.

2.3. ¿Qué es un equipo?

Un equipo está formado por lo menos de dos personas, cuyo trabajo está enfocado hacia un objetivo en común, donde a cada una se le han asignado roles o funciones específicas que deben desempeñar; y existe dependencia entre ellas mismas para poder finalizar tal objetivo [Ayer].

TSPi se diseñó para equipos de cinco o seis integrantes, tomando en cuenta la importancia del tamaño de un equipo, donde lo más conveniente es que esté formado de cuatro a ocho personas.

Condiciones básicas en un equipo de trabajo

Las condiciones que a continuación se mencionan son necesarias para que un grupo opere como equipo [Cummings, Dyer, Mohrman].

1. Clara definición de las tareas, de los roles y conocimiento del objetivo a cumplir por parte de cada integrante.
2. Conocimiento de cada integrante sobre el trabajo a cumplir y el rol de los demás.
3. Control del equipo sobre las tareas a realizar porque cada integrante conoce sus responsabilidades, cómo debe realizarlas, a partir de que fecha iniciarlas y cuándo finalizarlas.

2.4. Cómo TSPi construye equipos

TSPi guía a los equipos en su propia construcción estableciendo objetivos, asignando roles, estableciendo planes y manteniendo comunicación entre los integrantes del equipo.

Objetivos	TSPi proporciona un conjunto inicial de objetivos para el equipo y cada uno de sus integrantes, los cuales son necesarios para lograr un acoplamiento eficaz entre ellos mismos. En ciclos subsecuentes el equipo creará nuevos objetivos de acuerdo a la experiencia de la primera iteración.
Roles	El siguiente paso es indicar las responsabilidades mediante la asignación de roles a cada integrante. TSPi propone los siguientes roles: Líder del equipo (LE), Administrador de Desarrollo (AD), Administrador de Planeación (AP), Administrador de Calidad y Proceso (ACP) y Administrador de Apoyo (AA).
Planes	Posteriormente se define la estrategia para lograr los objetivos. Eso consiste en dividir el trabajo total en partes para los diferentes ciclos de desarrollo. El equipo define el contenido funcional del producto para cada ciclo, el tamaño esperado, las maneras en que se integrarán y evaluarán las piezas para generar el producto final. Posteriormente se realiza el plan de desarrollo del proyecto; en el cual se especifica el tamaño de los productos de cada ciclo, el tiempo para producir cada uno, el orden de las actividades y los responsables de cada tarea.
Comunicación	Mantener una comunicación efectiva es un problema en los equipos de trabajo [Poumaghshbanb]; para eso TSPi aconseja realizar reuniones semanales entre los integrantes.
Comunicación externa	Se refiere a la comunicación entre el equipo y otras partes (administrador o instructor). Con frecuencia, los equipos se comunican con el instructor o administrador cuando tienen problemas. TSPi pide al Líder de Equipo que haga un reporte que consiste en un resumen semanal y lo presente al instructor. Ahí se muestra lo que el equipo ha realizado y lo que sus integrantes planean para la siguiente semana.

Capítulo 3. Lanzamiento

Objetivo: Describir el proceso de lanzamiento del equipo.

En el lanzamiento se definen los objetivos del equipo y cómo establecerlos; se concluye con una descripción de los pasos del proceso de lanzamiento y los guiones correspondientes.

3.1. ¿Por qué es necesario realizar un lanzamiento de equipo?

Los equipos necesitan establecer sus relaciones de trabajo, determinar los roles de sus integrantes y estar de acuerdo en los objetivos. El estándar de roles de TSPi y sus responsabilidades se muestran en la siguiente tabla y se presenta mayor información en el Capítulo 11.

Responsabilidad.	LE	AD	AP	ACP	AA
Construye y mantiene un equipo efectivo.	X				
Resuelve asuntos entre los integrantes del equipo.	X				
Realiza seguimiento y reporte del progreso del equipo.	X				
Dirige cada reunión.	X				X
Se reúne con el instructor.	X				X
Realiza mantenimiento a la carpeta del proyecto.	X				
Ayuda a asignar las tareas en el equipo.	X				
Dirige todo el trabajo de desarrollo.		X			
Dirige la planeación del equipo y el seguimiento del progreso.			X		
Dirige el seguimiento y planeación de calidad.				X	
Proporciona apoyo al proceso del equipo.				X	
Dirige cada inspección.				X	
Mantiene los estándares y glosarios.				X	
Realiza el reporte de cada reunión.				X	
Alerta al equipo en los problemas de calidad.				X	
Obtiene las herramientas y apoyo necesarios.					X
Maneja la administración de configuración.					X
Dirige la Mesa de Control de Configuración.					X
Dirige el reuso del equipo.					X
Realiza el seguimiento de asuntos y riesgos.					X
Mantiene el glosario del sistema.					X
Desarrolla el producto.	X	X	X	X	X
Realiza planes personales.	X	X	X	X	X
Realiza seguimiento en el trabajo personal.	X	X	X	X	X
Genera productos de calidad.	X	X	X	X	X
Sigue prácticas personales disciplinadas.	X	X	X	X	X

Tabla 3.1 Estándar de roles y sus responsabilidades

3.2. Objetivos del equipo

Los objetivos son un paso esencial en la formación del equipo y se definen al inicio de cada proyecto porque establecen el marco para la estrategia y el plan; y proporcionan la base para cualquier producto que el equipo generará.

Pasos para establecer objetivos

1. Definir los objetivos del proyecto.
2. Especificar cómo medirlos.
3. Describir la razón por la que se van a utilizar los objetivos creados por el equipo y no los proporcionados por TSPi.
4. Proporcionar una copia de los objetivos a los integrantes del equipo y al instructor.
5. Proporcionar una copia de los objetivos al AA para que los anexe en la carpeta del proyecto.

Estableciendo objetivos guiados por TSPi

En primer lugar se debe considerar entregar al cliente un producto que cumpla con sus expectativas. Para ello TSPi considera los siguientes objetivos básicos:

1. Generar un producto de calidad.
2. Realizar un proyecto bien administrado y que sea productivo.
3. Terminar a tiempo.

El segundo paso consiste en medir los objetivos anteriores. TSPi sugiere las siguientes medidas.

1. Generar un producto de alta calidad.
 - 1.1. Porcentaje de defectos encontrados antes de la primera compilación: 80%.
 - 1.2. Número de defectos encontrados en la prueba del sistema: 0.
 - 1.3. Requerimientos funcionales incluidos al finalizar el proyecto: 100%.
2. Realizar un proyecto bien administrado y productivo.
 - 2.1. Error en el tamaño del producto estimado: <20%.
 - 2.2. Error en las horas de desarrollo estimadas: <20%.
 - 2.3. Porcentaje de datos registrados e incluidos en la carpeta del proyecto: 100%.
3. Terminar a tiempo.
 - 3.1. Días anteriores o posteriores para terminar el ciclo de desarrollo: <4.

Estableciendo objetivos para los integrantes del equipo

Se espera que cada integrante del equipo conozca los objetivos del proyecto, pero además cumpla los siguientes objetivos como integrante y sepa como medirlos.

1. Ser un integrante de equipo efectivo y cooperativo.
 - 1.1. Promedio en la forma Evaluación del Equipo por Colegas (EEC) por rol sobre el apoyo y ayuda: >3.
 - 1.2. Promedio en la forma EEC por rol en la contribución completa: >3.
2. Realizar un trabajo personal disciplinado.
 - 2.1. Porcentaje de datos personales registrados en la carpeta del proyecto: 100%.
 - 2.2. Porcentaje de semanas en las cuales se realizaron y finalizaron las actividades registradas en cada forma personal SEMANA: 100%.

Se debe realizar un seguimiento del tiempo, tamaño y los datos de defectos por trabajo personal registrándolos en las formas de TSPi. En la forma SEMANA se debe reportar el progreso del trabajo realizado contra lo planeado en la reunión semanal.

3. Planear y mantener un seguimiento del trabajo personal.
 - 3.1. Porcentaje personal de los datos del proyecto registrados en la formas RESPL y RESCA: 100%.
 - 3.2. Porcentaje de las tareas del proyecto con los datos actuales y planeados, terminados y registrados en la forma TAREA: 100%.

Cada integrante debe establecer un plan de trabajo personal y registrarlo en las formas TAREA y CALENDARIO (semanalmente) y mantener un seguimiento del valor ganado contra lo planeado.

4. Generar productos de calidad.
 - 4.1. Promedio del porcentaje de defectos encontrados antes de la primera compilación: >70%.
 - 4.2. Densidad de defectos encontrados durante la compilación: <10/KLOC.
 - 4.3. Densidad de defectos encontrados durante las pruebas unitarias: <5/KLOC.
 - 4.4. Densidad de defectos encontrados después de las pruebas unitarias: 0.

Se deben revisar personalmente cada uno de los productos y realizar inspecciones de equipo, cuyos resultados deben registrarse en la forma REPINS. Se deben producir planes de prueba aceptables y reportar los resultados en la forma REGPR.

3.3. Guiones de lanzamiento de TSPi

Para el lanzamiento del proyecto se establecen los pasos en el guión LAN1, a continuación se describen cada una de las actividades de dicho guión. En ciclos posteriores se utiliza el guión LANn.

Información del integrante	Cada integrante debe llenar la forma INFO, la cual será útil al instructor para formar los equipos y asignar los roles de acuerdo a la compatibilidad de horarios entre los integrantes, el interés de cada uno en determinado rol y su experiencia.
Objetivos del producto	El instructor describe el producto que se construirá y responde dudas acerca del curso, producto o proceso.
Asignaciones de equipo	En la segunda reunión, el instructor forma los equipos y asigna los roles.
Objetivos del equipo	El instructor explica el proceso para establecer los objetivos y se definen los objetivos de equipo y de cada rol.
Reuniones del equipo	Las reuniones son la base para la comunicación, planeación y toma de decisiones entre los integrantes. La primera se realiza en el lanzamiento. Cada equipo debe establecer un tiempo estándar y un lugar para realizarlas; y debe utilizar el guión SEMANA en cada una ellas. Antes de la reunión cada integrante debe actualizar TAREA y CALENDARIO, y resumir el contenido en una forma SEMANA personalizada. Con los datos de todas las formas SEMANA, el AP crea SEMANA, TAREA y CALENDARIO del equipo. La función más importante de la reunión semanal es conjuntar y analizar los datos del equipo a partir de la semana anterior hasta la fecha actual, para verificar el estado del trabajo planeado de cada integrante y si existe la necesidad de realizar ajustes a tales planes.
Requerimiento de datos	Los integrantes del equipo deben estar de acuerdo en los datos que proporcionarán al AP y la línea que seguirán para tal entrega. El AP necesita tales datos para generar el estado actual de cada semana y el LE para entregar el reporte semanal al instructor.
Inicio del proyecto	El proceso de lanzamiento ha finalizado y el equipo está listo para iniciar el proyecto usando el guión ESTRA. Antes de iniciar, es necesario estar familiarizado con la carpeta del proyecto.
La carpeta del proyecto	Permite estandarizar el almacenamiento de toda la información del proyecto. El responsable del mantenimiento de la carpeta será el LE y los demás integrantes deberán proporcionarle todo el material necesario.

Objetivo	Para iniciar al equipo en el primer ciclo de desarrollo.	
Criterio de entrada	<ul style="list-style-type: none"> Tener conocimientos sobre PSP. Los integrantes han leído los capítulos 1, 2, 3 y el Apéndice A. 	
General	<p>Este guión inicia con los proyectos de los equipos. Los objetivos principales son describir el curso.</p> <ul style="list-style-type: none"> Formar los equipos y asignar roles. Explicar los objetivos del producto a desarrollar. Establecer reuniones del equipo y reporte de tiempos. <p>Los pasos 1, 2 y 3 se realizan en la primera sesión, y los pasos del 4 al 8 durante la segunda sesión.</p>	
Paso	Actividades	Descripción
1	Panorama del curso	<p>El instructor describe los objetivos del curso:</p> <ul style="list-style-type: none"> Definiendo a los integrantes de los equipos. Indicando cómo su trabajo será evaluado. Indicando los principios básicos de trabajo en equipo. Definiendo el proceso de TSPi.
2	Información del integrante	<p>El instructor explica el criterio para hacer las asignaciones en los equipos.</p> <ul style="list-style-type: none"> La información necesaria para hacer las asignaciones. Los roles del equipo, sus responsabilidades y cualidades. <p>El instructor pide a los estudiantes:</p> <ul style="list-style-type: none"> Completar y regresar la forma INFO. Leer el capítulo 4 y la sección referente a la administración de configuración de software. Leer los capítulos de los roles que les interesen.
3	Objetivos del Producto	<p>El instructor describe los objetivos del producto:</p> <ul style="list-style-type: none"> Los objetivos críticos del producto que deben satisfacerse. Los objetivos opcionales y deseables. El criterio para evaluar el producto final.
4	Asignaciones del equipo	El instructor forma los equipos y asigna un rol a cada integrante.
5	Objetivos del equipo	El instructor explica porque es necesario establecer objetivos en el equipo y en cada rol.
6	Reuniones del equipo	<p>El instructor explica aspectos sobre las reuniones de equipo y cómo deberán ser dirigidas.</p> <ul style="list-style-type: none"> El propósito, horario y reportes de las mismas. Los requerimientos de datos semanales.
7	La primera reunión	<p>El LE dirige la primera reunión.</p> <ul style="list-style-type: none"> Se discuten los roles de los integrantes. Se discuten y se acuerdan los objetivos del ciclo 1. Se establece un tiempo estándar para la reunión semanal. Se acuerda el tiempo para que todos los integrantes proporcionen sus datos semanales al AP.
8	Requerimiento de datos	<p>El AP revisa:</p> <ul style="list-style-type: none"> Los datos proporcionados por cada integrante semanalmente. Los reportes que generará y entregará al equipo a partir de esos datos.
9	Inicio del proyecto.	El equipo empieza a trabajar usando el guión ESTRA1.
Criterio de éxito	<ul style="list-style-type: none"> Cada integrante ha completado y entregado la forma INFO. Los equipos de desarrollo están formados y los roles asignados. El instructor ha descrito los objetivos completos del producto. El instructor ha revisado y explicado el TSPi, los objetivos de los equipos y roles. El equipo está de acuerdo en los objetivos del ciclo 1, en el tiempo asignado para las reuniones semanales y en la entrega de datos para realizar los reporte semanales. 	

Tabla 3.2 Guión de Lanzamiento para el primer ciclo, LAN1

El guión LANn, que se muestra en la siguiente tabla se utiliza en el lanzamiento de ciclos posteriores.

Objetivo		Guiar a los equipos en el segundo o en ciclos de desarrollo posteriores
Criterio de entrada		Los equipos han terminado un ciclo de desarrollo anterior de TSPI.
General		Este guión inicia el segundo o el siguiente ciclo del proyecto. Los objetivos principales son: <ul style="list-style-type: none"> • Revisar el ciclo anterior. • Reorganizar los equipos, si es necesario, y asignar nuevos roles. • Establecer la reunión de equipos y reporte de tiempos. Los pasos 1 y 2 se terminan durante la primera sesión y los pasos del 3 al 8 durante la segunda.
Paso	Actividades	Descripción
1	Lecciones aprendidas	El instructor revisa los resultados del ciclo anterior y <ul style="list-style-type: none"> • Discute problemas del proceso y las causas comunes. • Sugiere cómo minimizar los problemas. • Revisa términos referentes al trabajo en equipo que causaron problemas en el ciclo anterior y explica cómo manejarlos.
2	Información del estudiante	El instructor revisa: <ul style="list-style-type: none"> • Las razones para hacer cambios de rol. • Las lecciones aprendidas sobre los roles en los ciclos anteriores. • Los problemas en el desempeño de los roles y cómo manejarlos en el ciclo siguiente. El instructor pide a los integrantes que: <ul style="list-style-type: none"> • Completen y regresen una nueva forma INFO. • Incluyan cambios en los horarios personales. • Elijan el rol de su preferencia.
3	Asignaciones del equipo	El instructor puede asignar a los estudiantes su equipo y rol (esas elecciones las pueden realizar los integrantes de equipos). Los estudiantes leen los capítulos de acuerdo al rol asignado o seleccionado.
4	Estableciendo objetivos	El instructor explica la necesidad de esforzarse en realizar un mejor desempeño en cada ciclo de desarrollo.
5	Reuniones de equipo	El instructor discute las necesidades de las reuniones semanales: el horario, el reporte semanal y los requerimientos de datos semanales.
6	La primera reunión	El LE convoca a su nuevo equipo a reunión. <ul style="list-style-type: none"> • Se discuten los roles. • Se revisan y actualizan los objetivos del equipo. • Se establece un tiempo estándar para la reunión semanal. • Se acuerda que cada semana los integrantes del equipo proporcionen los datos semanales al AP.
7	Requerimiento de datos	El AP revisa: <ul style="list-style-type: none"> • Los datos de cada integrante semanalmente. • Que los reportes sean generados y proporcionados al equipo a partir de esos datos.
8	Inicio del proyecto	El equipo empieza a trabajar usando el guión ESTRAN.
Criterio de éxito		<ul style="list-style-type: none"> • Cada estudiante ha completado y entregado la forma INFO. • Los equipos de desarrollo se formaron y se asignaron nuevos roles. • El equipo está de acuerdo en los objetivos del nuevo ciclo, en las reuniones semanales y en los datos necesarios que se deben entregar semanalmente.

Tabla 3.3 Guión de Lanzamiento para ciclos posteriores, LANn

Capítulo 4. Estrategia de Desarrollo

Objetivos: Crear una estrategia para realizar el trabajo. Crear el diseño conceptual del producto. Hacer estimaciones preliminares sobre el tamaño del producto y el tiempo de desarrollo. Finalmente, documentar la estrategia.

Este capítulo explica porque es necesaria la planeación antes de iniciar un proyecto, discute la administración de riesgos y la importancia del reuso. Posteriormente, se describen los guiones de la estrategia de TSPi.

Es conveniente planear antes de iniciar un proyecto porque al momento de desarrollar el plan los equipos obtienen una apreciación común del trabajo que deben hacer. Tal documento se utiliza para mantener el seguimiento de las actividades a realizar y establecer fechas de terminación para las mismas. También permite que los equipos desde un inicio mantengan ciertos cuidados sobre los problemas comunes que pueden ocurrir.

4.1. La estrategia de TSPi

Consiste en aplicar un proceso cíclico para desarrollar el producto. En cada ciclo se deciden cuáles funcionalidades desarrollar. En el primero de ellos se diseña, implementa y evalúa una primera versión del producto; en el segundo ciclo, se incrementan las funcionalidades del producto para generar una segunda versión. Si hay tiempo, se produce una tercera versión.

4.2. El diseño conceptual

El diseño conceptual es el punto inicial para la planeación del proyecto. Es importante que los participantes determinen cómo construirán el producto, cuáles serán los componentes principales y las funcionalidades de cada uno; por último, deberán estimar el tamaño y tiempo requerido para el desarrollo.

4.3. Administración de riesgos

Un *asunto* se define como un suceso real; cada equipo tiene que planear asignaciones de tiempo para manejar los asuntos conocidos. Un *riesgo* se detecta cuando algo puede o no suceder; para evitarlos o controlarlos, es conveniente tomarlos en cuenta desde el principio y determinar las precauciones adecuadas para ello.

Al establecer una estrategia se debe definir el criterio de la misma; identificar diferentes estrategias que se podrían utilizar; identificar riesgos y beneficios en cada una; hacer una evaluación comparativa de las mismas; decidir cuál utilizar y documentarla.

4.4. Una estrategia de reuso

Es conveniente que desde el inicio de un proyecto de software se tome en cuenta el reuso en el código. Identificar casos donde un solo componente o función pudiera satisfacer varias necesidades funcionales, permitiría ahorrar tiempo en el desarrollo de los ciclos.

4.5. Los guiones de Estrategia

El proceso de estrategia se describe en los guiones ESTRA1, para el primer ciclo de desarrollo y ESTRAn, para ciclos subsecuentes, los cuales se presentan a continuación.

Objetivo	Guiar al equipo a producir una estrategia de desarrollo con TSPI y estimaciones preliminares de tamaño y tiempo	
Criterio de entrada	<ul style="list-style-type: none"> • Los estudiantes han leído el capítulo 4. • El instructor ha revisado y discutido el proceso de TSPI. • El instructor ha descrito los objetivos completos del producto. • Los equipos de desarrollo han sido formados y los roles asignados. • Los equipos están de acuerdo en los objetivos de su trabajo. 	
General	<p>La estrategia de desarrollo especifica el orden en el que las funcionalidades del producto son definidas, diseñadas, implementadas y evaluadas.</p> <ul style="list-style-type: none"> • Cómo el producto se incrementará en ciclos futuros. • Cómo dividir el trabajo de desarrollo entre todos los integrantes. <p>La estrategia de desarrollo se produce al inicio del proceso para guiar la estimación en tamaños y la planeación de recursos.</p> <ul style="list-style-type: none"> • Si la estrategia cambia durante la planeación, se deberán actualizar los requerimientos o el desarrollo. <p>Las estimaciones preliminares en tamaño y tiempo.</p> <ul style="list-style-type: none"> • Crear el plan de trabajo para cada ciclo de desarrollo. • Proporcionar las bases para asignar el trabajo entre todos los integrantes. 	
Paso	Actividades	Descripción
1	Panorama de la estrategia	<p>El instructor describe la estrategia de desarrollo.</p> <ul style="list-style-type: none"> • Qué es la estrategia, cómo es producida y cómo es usada. • El criterio para una estrategia efectiva. • La lógica para determinar las estimaciones en tiempo y tamaño, y las maneras de realizarlas.
2	Establezca un criterio de estrategia	<ul style="list-style-type: none"> • El AD dirige al equipo en la definición de los criterios de estrategia. • El ACP documenta los criterios y proporciona copias a todos los integrantes y al instructor.
3	Produzca el diseño conceptual	El AD dirige al equipo para que produzca el diseño conceptual del producto completo.
4	Seleccione la estrategia de desarrollo	<p>El AD dirige al equipo a producir la estrategia de desarrollo. Esto involucra:</p> <ul style="list-style-type: none"> • Proponer y evaluar estrategias diferentes. • Asignar funcionalidades del producto en cada ciclo de desarrollo. • Definir cómo subdividir e integrar el producto.
5	Produzca la estimación preliminar	<p>El equipo es dirigido por el AP para realizar las estimaciones preliminares en tamaño y tiempo.</p> <ul style="list-style-type: none"> • Para los productos del ciclo actual. • Para los productos de ciclos posteriores.
6	Evalúe riesgos	Identifique y evalúe riesgos del proyecto, regístrelos en la forma RAR.
7	Documente la estrategia	El mediador de la reunión (ACP) documenta la estrategia seleccionada en la forma ESTRA.
8	Produzca el Plan de Administración de Configuración	<p>El AA produce el plan de administración de configuración.</p> <ul style="list-style-type: none"> • Identifica la Mesa de Control de Configuración y sus procedimientos. • Especifica y facilita las herramientas de soporte necesarias. • Revisa los procedimientos con el equipo para el común acuerdo.
Criterio de éxito	<ul style="list-style-type: none"> • La estrategia de desarrollo terminada y documentada. • Estimaciones en tamaño y tiempo terminadas y documentadas para todos los elementos del producto que se generarán en el ciclo siguiente. • Procedimiento de administración de configuración documentado. • Registro de los riesgos y asuntos en la forma RAR. • El diseño conceptual y la forma ESTRA terminados. • Carpeta del proyecto actualizada. 	

Tabla 4.1 Guión para la estrategia de desarrollo del primer ciclo, ESTRA1

Objetivo		Guiar al equipo a actualizar la estrategia y las estimaciones en tamaño y tiempo para ciclos subsecuentes.
Criterio de entrada		<ul style="list-style-type: none"> • Los ciclos de desarrollo anteriores han sido evaluados y completados. • Reasignación de nuevos roles y lectura de los capítulos correspondientes a tales roles.
General		<p>El grado en el cual la estrategia de desarrollo debería ser modificada depende de los resultados de los ciclos anteriores.</p> <ul style="list-style-type: none"> • Si la estrategia se siguió y se utilizó, se aconseja no cambiarla, de lo contrario ajústela. <p>Las estimaciones deberían actualizarse para reflejar los resultados de ciclos anteriores.</p> <ul style="list-style-type: none"> • Mida los productos del primer ciclo y use los datos en la actualización.
Paso	Actividades	Descripción
1	Panorama de la estrategia	<p>El instructor describe las razones por las cuales la estrategia debe ser actualizada.</p> <ul style="list-style-type: none"> • Cuando los cambios son apropiados y porque se deben evitar. • Detecta problemas en la estrategia. • La lógica para determinar el tamaño del ciclo siguiente.
2	Actualice la estrategia de desarrollo	<p>El AD dirige al equipo en la actualización de la estrategia de desarrollo.</p> <ul style="list-style-type: none"> • De acuerdo a los resultados en revisiones de ciclos anteriores para realizar los cambios necesarios. • A definir cómo subdividir e integrar los productos. • Si los cambios no son necesarios, no debe realizar los pasos 3 y 5.
3	Produzca las estimaciones actuales en tamaño actualizado y tiempo	<p>El AP dirige al equipo a realizar modificaciones en las estimaciones de tamaño y tiempo para:</p> <ul style="list-style-type: none"> • Todos los productos del ciclo siguiente. • Los productos de ciclos subsecuentes.
4	Evalúe riesgos	Revise, actualice y evalúe riesgos del proyecto y regístrelos en la forma RAR.
5	Documente la estrategia	El mediador de la reunión (ACP) documenta la estrategia seleccionada en la forma ESTRA.
6	Revise y actualice el Plan de administración de configuración.	<p>El AA dirige al equipo en la revisión del plan de administración de configuración y en la identificación de modificaciones necesarias.</p> <ul style="list-style-type: none"> • Considerando la mesa de control de configuración y los procedimientos de control de cambios. • Considerando la utilización de herramientas de soporte actuales u otras.
Criterio de éxito		<ul style="list-style-type: none"> • La estrategia de desarrollo actualizada y documentada. • Estimaciones en tamaño y tiempo actualizadas y documentadas para todos los elementos del producto que se generarán en el ciclo de desarrollo siguiente. • Estimaciones terminadas y documentadas para los productos que se generarán en ciclos de desarrollo posteriores. • Procedimiento de la administración de configuración actualizado. • Riesgos y asuntos actualizados, y registrados en la forma RAR. • Diseño conceptual y la forma ESTRA actualizados. • Carpeta del proyecto actualizada.

Tabla 4.2 Guión para la estrategia de desarrollo de ciclos posteriores, ESTRAN

Capítulo 5. El Plan de Desarrollo

Objetivo: Realizar el plan de desarrollo del proyecto y el plan de calidad. Aplicar el criterio de estándares de calidad para generar un producto de alta calidad.

Este capítulo proporciona los guiones PLAN1 y PLANn, para el primero y ciclos posteriores, respectivamente. Indica cómo mantener un seguimiento del trabajo realizado contra lo planeado, y se detalla todo lo referente al plan de calidad.

Para realizar un plan, TSPi se basa en los métodos utilizados en los proyectos industriales de TSP.

5.1. La necesidad de planear

Plan balanceado

Un proceso de planeación es el contexto mediante el cual se diseña la manera en la que se hará un trabajo. Al planear es común entablar compromisos en los cuales se establezcan fechas y se detecten los recursos necesarios. Una de las causas principales que originan problemas de horario es la carga de trabajo no balanceada, sobre todo cuando se asigna mayor trabajo a algunos ingenieros que a otros. Un plan balanceado es aquel en el que todos los integrantes del equipo terminan las tareas planeadas en el orden apropiado y, aproximadamente al mismo tiempo.

Manteniendo seguimiento del trabajo realizado contra el planeado

Un plan permite mantener un seguimiento y administración sobre el trabajo necesario en el desarrollo de software.

En TSPi, las principales medidas para mantener un seguimiento en los proyectos son el valor planeado (VP) y el valor ganado (VG), mediante las cuales se puede saber si las tareas se han cumplido dentro del tiempo estipulado en el plan de desarrollo. VP es el porcentaje que deriva de las horas planeadas en cada tarea con respecto a las horas totales del plan de trabajo. VG es la suma de todos los VPs de acuerdo a las tareas que se terminaron completamente (para mayor información sobre el método de VG, consulte el Capítulo 6 de *A Discipline for Software Engineering [Humphrey 95]*).

Planeando con detalle

TSPi requiere que cada integrante estime 10 o menos horas a las tareas asignadas, aunque habrá tareas como los requerimientos y el diseño de alto nivel que necesitarán más de 10 horas porque serán desempeñadas por varios ingenieros. Para tareas como escribir programas pequeños, TSPi usa la forma Resumen del Plan de Proyecto (RESPL), la cual se utiliza para planear el proyecto completo del equipo. De nuevo se vuelve utilizar esa forma para realizar planes más precisos. Al mismo tiempo, se puede rehacer el plan de calidad usando la forma RESCA. Para tareas que no requieran actividades de programación (como escribir documentos, procesos de diseño o planes de pruebas) en las que se estime utilizar más de 10 horas, TSPi proporciona la forma RESTAREA.

Es conveniente asignar una o dos horas semanalmente para las tareas no planeadas, a las cuales se les llama de administración y miscelánea A&M (management & miscellaneous).

Nivel de estimación

Con frecuencia los programas desarrollados durante el curso, estarán integrados en tres niveles: el sistema, módulos (o componentes del sistema) y los objetos (o funciones del módulo). Generalmente, TSPi se refiere a partes y ensambles. Por lo tanto un programa que se descompone en módulos se considera una parte. Cuando un programa está formado por varias partes, se llama ensamble. Porque los proyectos desarrollados con este proceso tienen dos niveles de ensamble, se deberá hacer un plan del sistema estimando los tamaños de las partes a nivel módulo.

5.2. Los guiones del Plan de Desarrollo

Los guiones PLAN1 y PLANn se muestran a continuación. El guión PLAN1 se utiliza para la planeación del proyecto durante el primer ciclo, y el PLANn para ciclos posteriores. Aunque el proceso de planeación es similar en ambos guiones, los siguientes párrafos describen los guiones del PLAN1, la mayoría de ellos pueden aplicarse al PLANn.

Criterio de entrada

Antes de iniciar con el guión PLAN1, hay que asegurarse de haber terminado la estrategia de desarrollo y el diseño conceptual.

Haga una lista de los productos que serán generados en el ciclo de desarrollo y estime sus tamaños

- Asigne nombres a las partes del sistema (especificadas en el diseño conceptual) y estime el tamaño de cada una.
- Identifique otros productos que se puedan desarrollar en este ciclo y estime sus tamaños (documentos referentes a los requerimientos, las páginas de diseño de alto nivel, documentación para el usuario, materiales de prueba, código, etc.).

Registre los datos de los productos que se construirán y su tamaño durante este ciclo de desarrollo en la forma RESTM

Las instrucciones se indican en la tabla 5.3, puede utilizar las estimaciones de tamaño hechas en la fase de estrategia o modificarlas si desea.

Produzca la lista de tareas

La lista de tareas contiene los pasos principales de todos los guiones de desarrollo de TSPi, así como de otras tareas adicionales que deberá incluir.

Estime horas de trabajo para las tareas del equipo y de cada ingeniero

Utilice la forma TAREA para estimar cuánto tiempo dedicará a cada tarea de la lista. Para producir esas estimaciones se recomienda que todo el equipo trabaje en ello siguiendo los pasos que a continuación se indican.

1. Por cada tarea registre en "Fase" la fase del proceso en el cual se hará.
2. En la columna "Parte" indique el nombre de la parte generada por la tarea.
3. En "# Ingenieros" indique el número de ingenieros que se involucrarán en la tarea.
4. Registre las horas estimadas por cada ingeniero en las columnas correspondientes a cada rol.
5. Si el tiempo estimado para las tareas es mayor a 10 horas, se deben descomponer en subtareas más pequeñas y estimar el tiempo requerido para cada una.
6. - Registre el total de horas para cada tarea.

Registre las horas semanalmente en la forma CALENDARIO

Estime cuántas horas cada integrante dedicará al proyecto cada semana. Se aconseja que para el primer ciclo se incluyan horas para 8 o más semanas, por lo menos después del primer plan.

Produzca las formas TAREA y CALENDARIO de todo el equipo

Después de haber registrado las horas semanales en la forma CALENDARIO del equipo, calcule el valor planeado y la fecha de terminación para cada tarea, después genere las formas TAREA y CALENDARIO terminadas.

Siga los pasos mencionados a continuación para generar el producto en el tiempo disponible:

1. Obtenga el total de horas planeadas de todos los integrantes del equipo en cada ciclo de vida del proyecto.

2. Compare el tiempo total del primer paso con el tiempo total requerido para cumplir con todas las tareas de cada ciclo.
3. Si el tiempo en cada tarea excede el tiempo total planeado deberá continuar con el siguiente paso.
4. Si el trabajo es bastante, deberá decidir entre trabajar más horas o reducir el trabajo planeado; lo cual se debe registrar en la forma calendario y verificar si está de acuerdo al tiempo disponible.
5. Si se decide reducir la cantidad de trabajo, en la fase de estrategia revise las funciones que se implementarán en cada ciclo. Haga otra vez el plan.

Después de terminar las formas TAREA y CALENDARIO del equipo en este ciclo, se tienen que completar las partes de tamaño y tiempo en la forma RESPL.

Produzca el plan de calidad

A continuación se describen los pasos para realizar el Plan de Calidad, las secciones del plan se describen a detalle mas adelante, en este capítulo.

1. Estime cuántos defectos se introducen aproximadamente en cada fase. Base la estimación en el tiempo dedicado en cada fase multiplicándolo por la tasa de inyección de defectos, utilice la tabla Criterio Estándar de Calidad (se encuentra en el Anexo C, Tabla C.5).
2. Estime la tasa de eliminación de defectos que el equipo aplicará en cada fase, de acuerdo al criterio que se muestra en la tabla Criterio Estándar de Calidad.
3. Establezca los objetivos para las tasas en la fase de pruebas de acuerdo a los defectos encontrados, tomando en cuenta el criterio de la tabla mencionada anteriormente.
4. Genere el plan de calidad.
5. Compare los valores del plan de calidad con los que integra la tabla Criterio estándar de calidad.
6. Si las tasas de eliminación de defectos y revisiones e inspecciones no están cercanas al criterio, ajuste el tiempo dedicado a la parte de eliminación de defectos o modifique las tasas de eliminación de defectos hasta que los valores estén cercanos.
7. Si las tasas de defectos son menores, revise los datos para asegurarse que todos los criterios se cumplieron.
8. Si los valores de los defectos/KLOC son altos en la compilación o en pruebas revíselos hasta que estén dentro del rango. También revise las tasas para las revisiones e inspecciones; si son menores increméntelas hasta que sean las adecuadas. Posiblemente, también se deberá incrementar el tiempo asignado a revisiones o inspecciones para mantener las tasas de eliminación de defectos dentro del rango.
9. Repita los pasos desde el número 4 hasta el 8 hasta que el plan de calidad cumpla con los objetivos del equipo.
10. Genere el plan de calidad final en la forma RESCA.

Para mayor información sobre los cálculos revise las instrucciones para llenar la forma RESCA.

Haga planes individuales (por cada ingeniero)

Es conveniente que en los pasos 6 y 7 el equipo trabaje en conjunto. Inicie estimando horas para cada tarea y horas disponibles semanalmente por cada ingeniero, entonces asigne tareas a cada uno de ellos semanalmente. Entregue copias de las formas TAREA y CALENDARIO a cada ingeniero, donde cada uno deberá eliminar las horas que los demás ocuparán. Añada tiempo para actividades que no se hayan cubierto mediante la fase A&M. Finalmente, genere las formas TAREA y CALENDARIO de cada ingeniero.

Balancee la carga de trabajo

Revise que la carga de trabajo sea balanceada después de haber completado las formas TAREA y CALENDARIO. El primer paso es identificar aquellas tareas cuya fecha de terminación rebasa el calendario establecido. Entonces divida esas tareas entre dos o más ingenieros. Actualice los datos de tiempos en las formas TAREA, RESPL y RESCA.

Paso final. Producir y distribuir los planes

En este paso se debe producir el plan consolidado del equipo, esto consiste en obtener: el Resumen del Plan del Proyecto a nivel equipo (RESPL), el Plan de calidad a nivel equipo (RESCA) y las formas TAREA y CALENDARIO del equipo y de cada ingeniero. Esos planes muestran los datos del producto en tiempo, tamaño y defectos; el valor planeado semanalmente por el equipo; las horas semanales por tarea; el tiempo planeado por cada tarea y el tiempo en que cada una se terminará.

Las instrucciones para la forma RESPL se muestran en la tabla de instrucciones de la forma RESPL (tabla 5.4).

Criterio de éxito

El criterio de éxito para los guiones PLAN1 y PLANn requiere terminar las formas TAREA y CALENDARIO del equipo y de cada ingeniero. Los planes deben estar balanceados e indicar que el trabajo se terminará en el tiempo establecido. Se deben poner copias en la carpeta del proyecto de las formas Resumen del Tamaño (RESTM), Resumen del Plan del Proyecto (RESPL) y Plan de calidad (RESCA).

Nota. Las formas RESPL y RESCA no se incluyeron en la presente tesis, lo puede consultar en la bibliografía utilizada (Introduction to the Team Software Process).

5.3. Seguimiento del trabajo

La razón para dar seguimiento a un proyecto consiste en determinar el estado y calidad del trabajo. Los pasos en el proceso de seguimiento del proyecto son los siguientes:

1. Registre el tiempo en la forma de registro de tiempo REGT.
Como se mantuvo seguimiento de tiempo en REGT, los datos se utilizan para actualizar los tiempos actuales de las tareas en la forma personal TAREA y los tiempos semanales actuales en la forma personal CALENDARIO.
2. Registre el número de semana en la cual se terminaron las tareas.
Cuando una tarea se termine registre el número de semana en la forma TAREA. Calcule el valor ganado en total de la semana.
3. Registre datos en la forma SEMANA.
A partir de los datos registrados en TAREA y CALENDARIO llene la forma SEMANA. Dé una copia al AP.
4. Registre los defectos en la forma de registro de defectos REGD.
Los defectos de la forma REGD, regístrelos en RESPL donde fueron inyectados y eliminados. También los datos de la forma REPINS deberá incluirlos en REGD durante la reunión de inspección.
5. Registre el tamaño por componente.
Después de desarrollar un producto registre su tamaño en la forma RESTM (por ejemplo, páginas de requerimientos y diseño de alto nivel). Cada vez que modifique el tamaño del producto o de una parte, actualícelo en la forma RESTM. Utilice los datos para complementar las formas RESPL y RESCA.
6. Actualice la forma RESPL.
Conforme avanza en el trabajo, actualice las formas RESPL con los datos de tamaño, tiempo y defectos. En otra forma RESPL por separado, totalice los tiempos de todos los ingenieros por cada fase y ensamble.
7. Genere la forma RESCA.

- Con los datos de tiempo, tamaño y defectos genere una copia de RESCA por cada ensamble.
8. Produzca el resumen consolidado del estado del equipo.
Semanalmente cada ingeniero debe entregar una copia del trabajo personal planeado al AP, quién utilizará esos datos para generar las formas TAREA y CALENDARIO del equipo; posteriormente deberá generar la forma SEMANA del equipo.
El resumen del estado del equipo consiste en actualizar las formas TAREA, CALENDARIO, RESPL y RESCA con los datos planeados y actuales.
RESPL y RESCA se generan en cada ensamble y TAREA y CALENDARIO son generadas por equipo y por cada ingeniero.
La forma RESPL a nivel ensamble contiene los datos para generar la forma RESCA a nivel ensamble.
Para generar el ensamble RESPL se proporcionan formas adicionales a partir de todas las formas RESPL; las cuales son resumen de tiempo (RESTD), resumen de defectos inyectados (RESDI) y resumen de defectos eliminados (RESDE). Esas formas contienen los datos consolidados de tiempo, tamaño y defectos de todas las partes del ensamble. Utilice los totales para completar la forma RESPL a nivel ensamble.
Utilice el guión SEMANA y las instrucciones de esa forma para reportar al equipo y al instructor las actividades semanales.

5.4. Plan de calidad

El guión PLAN1 indica cómo producir el plan de calidad. A continuación se describe el contenido y los criterios para generar un buen plan de calidad.

Proporciones de resumen

Las tres medidas de resumen proporcionan una perspectiva completa en la calidad del proceso. La medida de LOC/hr miden la productividad completa del equipo, un número mayor indica una productividad mayor y costos menores. El porcentaje de reuso mide el porcentaje de las LOC totales del producto que fueron reutilizadas a partir de productos previamente desarrollados. Similarmente, el porcentaje de reuso nuevo mide la contribución del ciclo en la mejora de productividad para ciclos o proyectos futuros.

Por ciento libre de defectos (PLD)

Mide el porcentaje de los componentes de un producto que no tuvieron defectos en una fase determinada. Por ejemplo, si un ensamble estuvo conformado por cinco partes y cuatro de ellas tuvieron defectos de compilación en el primer ciclo, tal ensamble tuvo 20% libre de defectos en la compilación. Los productos de calidad alta deberían alcanzar o exceder un 90% en las pruebas del sistema.

Defectos por página

Esta medida muestra el número promedio de defectos eliminados en cada página de los documentos de requerimientos y de diseño de alto nivel.

Defectos por KLOC

Un defecto es cualquier elemento que puede afectar la buena calidad de los productos. Un defecto mayor es cualquier problema que al momento de arreglarse origina cambios en el programa ejecutable.

El número de defectos por KLOC encontrados en la fase de pruebas indica la calidad del producto que entra y sale de esa fase. Cuando un producto tiene bastantes defectos al momento de realizar las pruebas se encontrarán muchos de ellos, pero también habrá otros que quedarán si detectarse.

La experiencia ha mostrado que cuando los productos tienen menos de 0.5 defectos/KLOC en las pruebas de construcción e integración y menos de 0.2 defectos/KLOC en las pruebas del sistema los usuarios no llegan a encontrar tales defectos.

Proporciones de defectos

Proporciona un panorama acerca de la calidad en las revisiones del diseño y código. Por ejemplo, la experiencia muestra que cuando los ingenieros encuentran dos veces los mismos defectos tanto en la revisión de código como en la compilación, se dice que han hecho una buena revisión de código. Esto daría una tasa de defectos en la revisión/compilación de código de 2.0.

Tasas en tiempo de desarrollo

Otra manera de evaluar un proceso es identificar el tiempo permanecido en cada fase de desarrollo. Generalmente, cuando los ingenieros permanecen mayor tiempo en el diseño detallado que en la codificación, la experiencia muestra que producen un diseño de calidad; y cuando pasan más del 50% del tiempo que dedicaron al diseño detallado en su propia revisión, de acuerdo a la experiencia, se dice que la revisión de diseño fue esmerada.

Para los requerimientos, los datos son menos claros, pero se aconseja dedicar 25% o más del tiempo que se utilizó en el desarrollo de los mismos para su revisión. Para el diseño de alto nivel se aconseja dedicar 50% del tiempo que se utilizó para el desarrollo (de tal diseño) en las revisiones e inspecciones. También la tasa que se aconseja aplicar para la revisión del código de acuerdo al tiempo de codificación dedicado debe ser mayor al 50%.

Cuando los ingenieros no tienen mucha experiencia en un lenguaje de programación es mejor utilizar los valores de defectos/KLOC obtenidos a partir de la compilación y pruebas unitarias como guías indicadoras de calidad.

Actividades de Validación de fallas (A/FR)

Otra tasa de tiempo de desarrollo útil es A/FR, la cual es una tasa de validación de fallas. Se refiere a la proporción de tiempo dedicado a las actividades de tipo-validación (tales como revisiones e inspecciones) hasta el tiempo dedicado a las actividades en las que se encuentran fallas (tales como compilación y pruebas). Se aconseja que el A/FR sea mayor a 2.0 para programas stand-alone. Para programas mayores como los producidos por TSPi, se considera adecuado 1.0. La razón es que los productos más complejos y grandes necesitan mayor tiempo de pruebas aún cuando tengan pocos o ningún defecto.

Tasas en revisión e inspección

Para llevar a cabo una alta calidad se debe hacer una planeación que permita permanecer tiempo suficiente en las revisiones e inspecciones. Se recomienda utilizar las siguientes líneas guía:

- Revisiones e inspecciones en requerimientos: <2.0 páginas por hora cuyo texto tendrá interlineado sencillo.
- Revisiones e inspecciones en diseño de alto nivel: <5 páginas por hora de diseño.
- Revisiones e inspecciones en diseño detallado: <100 líneas por hora en pseudocódigo.
- Revisiones e inspecciones de código fuente: <200 LOC/hora.

El tiempo de inspección total es igual a la suma de los tiempos dedicados en la preparación de las inspecciones más el tiempo que se utilizó en la reunión, multiplicado por el número de ingenieros que participaron.

Al momento de hacer revisiones no se asegura que se encontrarán defectos, por eso es necesario utilizar métodos de revisión efectivos. Para revisiones e inspecciones de código se recomienda utilizar una lista de verificación personal y actualizarla frecuentemente a partir de los defectos de compilación y pruebas encontrados en los programas que se han revisado e inspeccionado (mayor información se encuentra en *A Discipline for Software Engineering*, Capítulo 8 [Humphrey 95]). Para realizar revisiones de diseño efectivas es necesario que el diseño esté bien documentado y que se analice la lógica del programa, eso es posible utilizando métodos de análisis, tales como tablas de ejecución, tablas de seguimiento o análisis de máquinas de estado (más información se encuentra en los Capítulos 10 y 12 de la bibliografía mencionada anteriormente).

Tasas en inyección de defectos

Al tener datos sobre las tasas de inyección de defectos se tiene una base para estimar cuántos defectos se inyectarán al momento de programar, esto se refiere al número de defectos que se introducen en una medida de tiempo, por ejemplo, dos defectos por hora se introducen en el diseño detallado y seis defectos por hora en la programación. Sólo será necesario multiplicar la tasa de inyección de defectos esperada por el tiempo estimado que se dedicará a cada fase. Tal cálculo permitirá obtener una estimación del número posible de defectos que, generalmente, se pueden inyectar.

Tasas de eliminación de defectos

Permite calcular las tasas de eliminación de defectos al momento de programar. Aunque aquí los datos varían, eliminando desde cero hasta veinte defectos por hora; pero se ha encontrado que la tasa más común consiste en eliminar seis defectos por hora. También en las revisiones del diseño los datos varían, generalmente se eliminarían dos o más defectos por hora en los programas finales.

Producción por fase

En TSPi, la producción por fase se refiere al porcentaje de defectos de un programa que se eliminaron en una fase dada.

Si se tuvieron 19 defectos en un programa al realizar la revisión de código y se introdujo 1 defecto durante esa revisión, y se encontraron 15 de esos defectos en la revisión, se tendrá un 75% de producción en la revisión de código. Esto es:

$$\text{producción} = 100 * (\text{defectos encontrados}) / (\text{defectos del producto}) = 100 * 15 / (19 + 1) = 75\%$$

Esta misma fórmula se puede aplicar a cualquier fase del proceso. Se debe calcular la producción por fase, hasta que el programa se haya terminado, probado y utilizado por un período determinado; porque en caso contrario faltarían defectos por encontrar y con ello la producción sería menor.

Producción del proceso

La producción del proceso mide el porcentaje de defectos eliminados antes de determinada fase. Por ejemplo, la producción del proceso durante o antes de la compilación mide el porcentaje de defectos inyectados antes de la compilación y que fueron eliminados antes de la misma. TSPi tiene como objetivo eliminar el 97.5% o más defectos antes de la construcción e integración, y eliminar más del 99% antes de la prueba del sistema. Así se producirán productos de alta calidad. Una manera de alcanzar tales porcentajes sería obtener un 75% antes de la primera compilación y 85% antes de la primera prueba unitaria. Para alcanzar esos objetivos también es importante que los integrantes del equipo actualicen sus listas de verificación de revisión, hagan revisiones personales esmeradas e inspecciones en los requerimientos, el diseño de alto nivel, el diseño detallado y el código.

Partes de calidad baja

Cuando los datos obtenidos en calidad no son óptimos de acuerdo al estándar definido, localice las partes que tengan un alto número de defectos, tasas y producciones bajas. Después revise el diseño y el código fuente para detectar el origen del problema. Generalmente, pudo haber sucedido por haber realizado un diseño pobremente documentado; o que la codificación, revisiones e inspecciones hayan sido realizadas sin cumplir con las indicaciones establecidas. Cuando se detecten los problemas, y que generalmente sean los mismos, se arreglan antes de la construcción, integración y pruebas del sistema.

Criterio de calidad

El criterio de esta sección se resume en la tabla que contiene el criterio estándar de calidad. Se utiliza la primera vez que se aplica el proceso TSPi. Después de haber terminado el primer ciclo del proyecto y haber adquirido cierta experiencia se podrán hacer modificaciones a los valores del

criterio; aunque lo más recomendable sería después de haber aplicado dos o más ciclos. Todos los criterios deben documentarse en la carpeta y en el ciclo correspondiente.

Propósito		Guiar al equipo en la realización de tareas individuales y en equipo, en el calendario y en los planes de calidad para desarrollar el primer ciclo.
Criterio de entrada		<ul style="list-style-type: none"> El equipo tiene una estrategia de desarrollo y un diseño conceptual Los estudiantes han leído el Capítulo 5.
General		<p>El plan de tareas define:</p> <ul style="list-style-type: none"> El tiempo requerido para desempeñar cada tarea del proceso. El orden aproximado para realizar las tareas. El valor planeado en cada tarea. <p>El plan de calendario proporciona:</p> <ul style="list-style-type: none"> El tiempo planeado por ingeniero semanalmente. El total de horas planeadas del equipo semanalmente. El valor planeado semanalmente. <p>Si los planes de tarea y de calendario indican que el proyecto no se terminará a tiempo, deberá reajustar la estrategia y de nuevo volver a planear.</p>
Paso	Actividades	Descripción
1	Panorama de la planeación	<p>El instructor describe el proceso de planeación.</p> <ul style="list-style-type: none"> Los planes de tareas y de calendario y cómo se producirán. El plan de calidad y cómo se producirá.
2	Registre las estimaciones de tamaño en la forma ESTRA	<p>Con el diseño conceptual y la forma ESTRA producida en la fase de estrategia, el AP dirige al equipo a:</p> <ul style="list-style-type: none"> Identificar otros productos y sus tamaños. Registrar la forma ESTRA y otros datos de tamaño en RESTM.
3	Produzca el plan de tareas	<p>El AP dirige al equipo a:</p> <ul style="list-style-type: none"> Producir la lista de tareas con las estimaciones en tiempo del equipo y de cada ingeniero. Registrar esos datos en la forma TAREA.
4	Produzca el plan de calendario	<p>El AP obtiene el número de horas estimadas que cada integrante del equipo planea dedicar semanalmente y</p> <ul style="list-style-type: none"> Registra las horas semanales en la forma CALENDARIO. Produce las formas TAREA y CALENDARIO del equipo. Rehace el plan si las horas son inadecuadas.
5	Produzca el plan de calidad.	<p>El ACP guía al equipo a:</p> <ul style="list-style-type: none"> Revisar los objetivos de calidad del equipo. Estimar la producción de defectos inyectados y de defectos eliminados. Generar y evaluar los planes de RESPL y RESCA. Hacer los ajustes necesarios al proceso para obtener un plan satisfactorio.
6	Produzca los planes individuales por cada ingeniero	<p>El AP ayuda a los ingenieros a hacer los planes personales.</p> <ul style="list-style-type: none"> Asignando las tareas entre los integrantes del equipo. Estimando el tiempo para desempeñar cada tarea. Registrando los datos en las formas TAREA y CALENDARIO. <p>Produciendo el calendario de valor planeado y las fechas de finalización de las tareas.</p>
7	Balancee la carga de trabajo del equipo	<p>El AP guía al equipo a:</p> <ul style="list-style-type: none"> Identificar desbalances en la carga de trabajo. Reasignar tareas para reducir el calendario. Producir planes balanceados para los ingenieros. Producir un plan consolidado para el equipo (formas TAREA, CALENDARIO, RESPL y RESCA).
Criterio de éxito		<ul style="list-style-type: none"> Las formas TAREA y CALENDARIO terminadas por el equipo y por cada ingeniero. Las formas RESPL, RESCA y RESTM terminadas. La carpeta del proyecto actualizada.

Tabla 5.1 Guión para el plan de desarrollo para el primer ciclo, PLAN1

Propósito		Guiar al equipo en la realización de tareas individuales y en equipo, un calendario y planes de calidad para desarrollar un segundo o ciclos posteriores.
Criterio de entrada		Tener planes terminados y datos actuales para los ciclos.
General		Los planes de tareas de ciclos subsecuentes se basan en: <ul style="list-style-type: none"> Las tareas y tiempos reales de ciclos anteriores. El plan de calendario se basa en: <ul style="list-style-type: none"> Las horas reales que los integrantes del equipo han dedicado en ciclos anteriores. El plan de calidad se basa en: <ul style="list-style-type: none"> El criterio de calidad del equipo. Los objetivos de calidad originales del equipo. Si los planes de tarea y calendario indican que el ciclo no será terminado a tiempo se deberá reajustar la estrategia y replanear.
Paso	Actividades	Descripción
1	Panorama de la planeación	El instructor describe brevemente problemas con los planes anteriores y sugiere cómo mejorarlos en este ciclo.
2	Actualice las estimaciones de tamaño	Con la forma ESTRA actualizada, el AP guía al equipo a actualizar la lista de productos y las estimaciones en tamaño. <ul style="list-style-type: none"> Registrando esos datos en la forma RESTM.
3	Produzca el plan de tarea actualizado	El AP guía al equipo a: <ul style="list-style-type: none"> Producir la lista de tareas para el siguiente ciclo con las estimaciones en tiempo. Asignar esas tareas entre los integrantes del equipo, estimando el tiempo que cada uno dedicará. Registrar los datos en la forma TAREA.
4	Produzca el plan de calendario actualizado	El AP obtiene el número de horas estimadas por cada integrante del equipo semanalmente y <ul style="list-style-type: none"> Registra las horas semanalmente en la forma CALENDARIO. Produce las formas TAREA y CALENDARIO del equipo. Produce un calendario del equipo con valor planeado. Rehace el plan si las horas son inadecuadas.
5	Produzca el plan de calidad actualizado	El ACP guía al equipo a: <ul style="list-style-type: none"> Comparar la calidad de desempeño del equipo con los objetivos. Establecer objetivos mejorados para el siguiente ciclo de desarrollo. Estimar la producción de defectos inyectados y de defectos eliminados. Generar y evaluar los planes RESPL y RESCA. Hacer los ajustes necesarios para obtener un plan satisfactorio.
6	Produzca los planes individuales por cada ingeniero	El AP ayuda a los ingenieros a hacer los planes personales: <ul style="list-style-type: none"> Asignando las tareas entre los integrantes del equipo. Estimando el tiempo para desempeñar cada tarea. Registrando los datos en las formas TAREA y CALENDARIO. Produciendo el valor planeado establecido en los planes en calendario y en tareas.
7	Balancee la carga de trabajo del equipo	El AP guía al equipo a: <ul style="list-style-type: none"> Revisar los planes de equipo e individuales. Identificar desbalances en la carga de trabajo. Reasignar tareas para reducir el horario. Producir planes balanceados para los ingenieros. Producir un plan consolidado para el equipo (formas TAREA, CALENDARIO, RESPL y RESCA).
Criterio de éxito		<ul style="list-style-type: none"> Las formas TAREA y CALENDARIO terminadas, del equipo y de cada ingeniero. Las formas RESPL, RESCA y RESTM terminadas. La carpeta del proyecto actualizada.

Tabla 5.2 Guión para el plan de desarrollo para ciclos posteriores, PLANn

Propósito	<ul style="list-style-type: none"> • Esta forma resume los datos para el tamaño del producto. • El nivel más bajo resume los datos sobre el tamaño de cada parte. • Los niveles más altos resumen los datos de tamaño para un ensamble y sus partes.
General	<ul style="list-style-type: none"> • Se utiliza esta forma para documentar el tamaño de las partes de un ensamble. • Se registran los totales obtenidos de la forma RESTM en RESPL, en el nivel ensamble.
Encabezado	<ul style="list-style-type: none"> • Se registra el nombre, fecha, nombre de equipo y del instructor. • Se registra la parte o ensamble y su nivel. • Se registra el número de ciclo.
Plan/Ensamble/ Actual	<p>Cuando esta forma se utilice para un plan, ensamble o datos actuales tome en cuenta las siguientes consideraciones:</p> <ul style="list-style-type: none"> • Para un plan: en la planeación a nivel sistema use RESTM para los tamaños estimados del sistema y todas sus partes. • Para un ensamble: use una forma RESTM por separado para los tamaños estimados en las partes de cada ensamble. • Para datos actuales: use una forma RESTM para los tamaños actuales de las partes de cada ensamble.
Nombre del Producto y partes	<ul style="list-style-type: none"> • Se indican los nombres de cada producto o parte. • Se incluyen productos a nivel sistema, tales como, ERS y DAN.
Medida del tamaño	<ul style="list-style-type: none"> • Se registran las medidas de tamaño de cada producto.
Tamaño del programa	<ul style="list-style-type: none"> • Por cada producto se registran el tamaño en LOC reales o estimadas, eliminadas, añadidas, modificadas, de reuso, y nuevas y cambiadas.
Totales	<ul style="list-style-type: none"> • Se registran los totales en cada categoría.

Tabla 5.3 Tabla de instrucciones de la forma RESTM

Propósito	<ul style="list-style-type: none"> • Esta forma contiene los datos planeados y actuales para los ensambles del programa.
General	<ul style="list-style-type: none"> • Un ensamble podría ser un sistema con productos múltiples, un producto con componentes múltiples o un componente con módulos múltiples. • Una parte podría ser un objeto, módulo, componente o producto. • Nota: Generalmente, los ensambles o módulos de nivel menor no tienen datos a nivel sistema, tales como requerimientos, diseño de alto nivel o pruebas del sistema.
Encabezado	<p>Se registran:</p> <ul style="list-style-type: none"> • El nombre del integrante, fecha, nombre de equipo y nombre del instructor. • El nombre del ensamble y su nivel. • El número de ciclo.
Columnas	<ul style="list-style-type: none"> • Plan: esta columna contiene los datos planeados de las partes o ensambles. • Actual: Para los ensambles, esta columna contiene la suma de los datos reales de las partes o ensambles (a nivel más bajo son los módulos).
Tamaño del producto	<ul style="list-style-type: none"> • Para documentos de texto y diseño se registran sólo los datos referentes a los tamaño nuevos y cambiados. • Para partes o ensambles del programa se registran todos los datos de LOC indicados. • Se obtienen los datos de la forma RESTM.
Tiempo en fase	<ul style="list-style-type: none"> • Se registran el tiempo estimado y actual por fase. • En un nivel menor se obtienen esos datos a partir de las formas TAREA. • Para ensambles de nivel mayor se obtienen los datos de tiempo de las partes a partir de los totales de la forma RESTDES y los datos a nivel ensamble a partir de la forma TAREA. • Por ejemplo, el diseño de alto nivel se tomaría a partir del ensamble de la forma TAREA y el tiempo de la prueba unitaria del total del módulo se tomaría de la forma RESTDES. • % actual: se registra el porcentaje del tiempo de desarrollo actual por fase.
Defectos inyectados	<ul style="list-style-type: none"> • Se registran los defectos inyectados estimados y actuales por fase. • Se registran las estimaciones de defectos al producir el plan de calidad. • Para los módulos, obtenga los datos actuales a partir de las formas REGD.

	<ul style="list-style-type: none"> • Para los ensambles obtenga los datos de los defectos a nivel módulo a partir de los totales de la forma RESDI y los datos a nivel ensamble a partir del ensamble de la forma REGD. • Por ejemplo, los defectos del diseño de alto nivel se tomarían a partir del ensamble de la forma REGD y los defectos de codificación del módulo total a partir de la forma RESDI. • % actual: se registra el porcentaje de defectos actuales inyectados por fase.
Defectos eliminados	<ul style="list-style-type: none"> • Se registran los defectos estimados y actuales eliminados por fase. • Se registran estimaciones de los defectos al producir el plan de calidad. • Para módulos se obtienen los datos actuales a partir de las formas REGD. • Para ensambles se obtienen los datos de los defectos a nivel módulo a partir de los totales de la forma RESDE y los datos a nivel ensamble a partir de la forma REGD. • Por ejemplo, los defectos de revisión del diseño de alto nivel se tomarían a partir de la forma REGD y los defectos de revisión del módulo código a partir de la forma RESDE. • % actual: se registra el porcentaje de defectos actuales eliminados por fase.

Tabla 5.4 Tabla de instrucciones de la forma RESPL

Capítulo 6. Definiendo los Requerimientos

Objetivo: Describir el proceso de requerimientos en TSPi. Explicar qué son, por qué son necesarios y discutir las características más importantes.

En este capítulo se explica el guión REQ1 y REQn, también se describen los pasos del proceso de requerimientos.

6.1. ¿Qué son los requerimientos?

En esta fase, el equipo genera la Especificación de Requerimientos de Software (ERS). En tal documento se hace una descripción clara de lo que será el producto, deberá incluir el criterio preciso para evaluarlo cuando esté terminado y asegurar que las funcionalidades sean las correctas. También proporciona retroalimentación al cliente acerca de lo que se pretende construir. En el curso de TSPi el instructor actúa como el cliente.

Se debe saber exactamente lo que el producto debe hacer antes de construirlo. En el inicio de esta fase se realiza la definición de necesidades por parte del cliente, si existieran necesidades que no son claras se debe redactar un documento con preguntas que hagan referencia a todas aquellas dudas para obtener una respuesta posteriormente. Después de haberlas aclarado, de nuevo se escriben los requerimientos en lenguaje coloquial y se revisan con los usuarios para verificar si realmente es lo que desean. En esta fase se ocupa una semana para el ciclo 1 y media semana para los ciclos 2 y 3.

6.2. Necesidad de los Requerimientos

Durante el desarrollo de requerimientos se revisan las necesidades del cliente y se formulan preguntas acerca de las funciones del producto. Tales preguntas se deben discutir entre todos los integrantes del equipo, lo cual servirá para detectar dónde se necesita información adicional. Si fuera necesario se formularán nuevas preguntas. A través de este proceso se logra un acuerdo común entre el equipo acerca de lo que se va a construir; por esta razón, es importante que cada integrante participe en la definición de los requerimientos y definan un proceso bien estructurado para construirlos.

6.3. Cambios en los requerimientos

Con frecuencia se suele hacer cambios a los requerimientos; generalmente, los usuarios no pueden saber lo que necesitan en forma precisa hasta que tratan de usar el producto terminado. Sin embargo, cuando se introduce un sistema nuevo se cambia la manera en que ellos trabajan. Conforme el trabajo de desarrollo progresa empiezan a apreciar la manera en cómo afectará su ambiente, mientras más claro es eso para los usuarios, pensarán en nuevas funciones y características, por esto, los requerimientos constantemente cambian hasta que se congelan en un producto.

La parte difícil del proceso de requerimientos de software es entender que es lo que los usuarios creen que necesitan y ayudarlos a definir esas necesidades en términos de funciones que generen un producto útil. El siguiente paso es realizar la ERS que represente el acuerdo común entre los usuarios y el equipo de trabajo acerca de lo que necesitan.

Es importante iniciar con un acuerdo preciso de lo que es el producto y así tener un mecanismo que proteja los requerimientos originales de los desacuerdos encontrados; eso será posible con la

ERS clara y precisa, con la cual se podrán administrar los cambios y proteger tal documento contra un cliente que insista en interpretar de manera diferente una función que ya ha sido diseñada.

En la ERS se describe la interpretación de la definición de necesidades; después de que los clientes hayan leído ese documento y estén de acuerdo en su contenido se podrá argumentar que cualquier cambio costará tiempo y/o dinero, en esto radica la importancia de dicho documento.

Extracción de los requerimientos

Generalmente, en proyectos de software industriales el primer paso en el proceso de requerimientos es la extracción de los mismos. En este momento se interroga a los clientes, usuarios u otros involucrados importantes para descubrir lo que necesitan realmente.

Los pasos principales de la extracción de los requerimientos son los siguientes:

1. Evaluar la factibilidad del sistema.
2. Conocer y comprender los asuntos organizacionales.
3. Identificar a todos los involucrados en el proyecto.
4. Registrar las fuentes de los requerimientos.
5. Definir el ambiente operativo del sistema.
6. Evaluar asuntos del negocio.
7. Definir las restricciones del dominio.
8. Registrar la razón de los requerimientos.
9. Realizar un primer prototipo de los requerimientos (probablemente pobremente entendido).
10. Definir los escenarios de uso.
11. Definir los procesos operacionales.

6.4. La Especificación de Requerimientos de Software (ERS)

En este documento el equipo describe las funciones que se planean desarrollar y cómo se pretende generar el producto, evitando describir la implementación, puesto que es una responsabilidad de la fase de diseño.

Existen varios estándares para especificar los requerimientos [Davis, página 202; Sommerville, página. 42; Pressman]. En TSPi, el equipo se concentrará en los requerimientos funcionales y operativos, mencionando a continuación los principales:

- Requerimientos funcionales: entradas, salidas, procesos y casos de uso
- Requerimientos de interfaz externos: usuario, hardware, software y comunicaciones
- Restricciones de diseño: formatos de archivo, lenguajes, estándares, compatibilidad, etc.
- Atributos: disponibilidad, seguridad, mantenimiento, conversión, etc.
- Otros requerimientos: base de datos, instalación, etc.

A continuación se muestran los puntos clave del contenido de ERS.

1. Tabla de contenido.
2. Introducción.
 - Propósito del documento Especificación de Requerimientos de Software
 - Definición del problema
 - Información del equipo
3. Requerimientos funcionales.
 - Definición de los requerimientos funcionales del sistema
 - Requerimientos del ciclo 1
 - Requerimientos del ciclo 2
 - Estructura top-down
4. Definición de reglas utilizadas en los requerimientos.
5. Requerimientos de interfaz externa.
 - Interfaz de usuario
 - Formatos en pantalla
6. Restricciones en diseño/implementación.
 - Acuerdos en estándares
 - Restricciones en desarrollo
7. Requerimientos especiales del sistema.
 - Documentación
 - Compatibilidad
8. Referencias y fuentes de información.

Tabla 6.1 Tabla del contenido de la ERS

Seguimiento en los requerimientos

Para asegurar el seguimiento funcional, enumere los párrafos y secciones de la ERS para identificar cada requerimiento. En la fase de estrategia el equipo realizó un seguimiento de los mismos, el cual continua en esta fase y en el diseño. Se sugiere etiquetar la definición de requerimientos, si el documento define varios de ellos en un párrafo, escriba el número de ese párrafo y el número de definición - por ejemplo, 4.5s2; etiqueta que se refiere a la función descrita en la segunda definición del párrafo 4.5. Mientras sea posible, se aconseja separar las definiciones funcionales de la manera descrita anteriormente.

Balanceando la carga de trabajo

Frecuentemente, en este período unos cuantos ingenieros son los responsables del trabajo y los demás tienen tiempo para desempeñar otras actividades; ellos deberían trabajar en tareas relacionadas con su rol, como por ejemplo: definición de estándares, organización de facilidades de soporte, generación de un prototipo de las funciones de alto riesgo o inicio en el diseño de alto nivel.

6.5. Los Guiones de Requerimientos de TSPi

TSPi tiene dos guiones para el desarrollo de requerimientos: REQ1 y REQn, los cuales se muestran más adelante. REQ1 produce la ERS para el primer ciclo de desarrollo y REQn actualiza tal documento para el ciclo siguiente. Generalmente las actividades en los dos guiones son las mismas, más adelante se describen las secciones que se concentran en el guión REQ1, se anotaron las áreas principales donde difieren las acciones de REQ1 y REQn.

En REQ1 se detallan solamente las funciones definidas en la estrategia de desarrollo para el primer ciclo. Es conveniente añadir definiciones de otras funciones que puedan ser útiles durante el diseño de la primera versión del producto.

A continuación se definen las secciones de los guiones de esta fase.

Criterio de entrada

Se debe tener un criterio de entrada en la fase de requerimientos que cumpla lo siguiente:

- Tener la estrategia de desarrollo y el plan.
- Tener el diseño conceptual generado durante el desarrollo de la estrategia.
- Haber leído este capítulo y la definición de necesidades.

Revisión de la definición de necesidades

En esta parte, el equipo ya ha leído la definición de necesidades, ha creado el diseño conceptual para el producto, ha generado una estrategia de desarrollo y ha producido un plan de trabajo. El siguiente paso consiste en definir los requerimientos del producto. En TSPi, el primer paso es revisar la definición de necesidades e identificar todas las dudas que posteriormente puedan plasmarse en un conjunto de preguntas para aclararse. Entonces, el AD dirige al equipo en esa revisión y se asegura que todos los integrantes comprendan claramente las preguntas y de que sean anotadas para discutirse con el instructor.

En esa revisión el equipo se debe centrar en aquellas funciones que se van a incluir en el primer ciclo de desarrollo. Sin embargo, no se puede tener una certeza absoluta de cuáles funciones serán necesarias hasta que se haya terminado el diseño.

Aclaración en la definición de necesidades

Después de revisar la definición de necesidades, el AD revisa la lista de preguntas y las notas que se tienen que aclarar con el instructor; para ello, debe tener una reunión con el instructor (puede estar el equipo completo también) y posteriormente documentar las respuestas y dar copias a cada integrante y al instructor.

Asignación de tareas de los requerimientos

El AD dirige al equipo a definir un formato para el documento ERS y a identificar todas las tareas que se deben realizar. Después de dividir las secciones el LE ayuda a asignarlas a los integrantes y a establecer acuerdos en fechas de término de cada una.

Documentación de los requerimientos

En la documentación de ERS se realiza una definición breve y clara de lo que se pretende construir. Por ejemplo, para los casos de uso, liste los pasos en un guión simple como el de PSP, use tablas y listas con viñetas y trate de evitar párrafos con demasiado texto. Una manera de asegurar el acuerdo entre el equipo es definir un caso de uso por cada función, los cuales pueden servir como escenarios de pruebas del sistema para verificar las funciones del producto. Cada tarea terminada se debe entregar al AD, quién las debe unir en un borrador de ERS y dar copias a los demás ingenieros para realizar la revisión.

Plan de pruebas del sistema

A partir de la fase de requerimientos se empiezan a planear las pruebas del sistema, para asegurar que contenga los acuerdos pactados entre los usuarios y el equipo de trabajo. Los casos de uso deberían servir como material base para el PPS, también se deben considerar pruebas de límite y pruebas de estrés, probar bajo condiciones de error considerando la usabilidad y los resultados de recuperación; todo eso se debe mencionar en el plan y explicar cuáles si y cuáles no se evaluarán y las razones de eso (mayor información se puede obtener en el capítulo Pruebas de Integración y del Sistema, en la sección Estrategia de Pruebas del Sistema y Planeación de las pruebas).

Inspección de los requerimientos y del Plan de Pruebas del Sistema

Se debe realizar una inspección en equipo del borrador de ERS y del Plan de Pruebas del Sistema. Se recomienda tiempo suficiente para tal inspección (media hora por página con espaciado sencillo). El ACP debe dirigir la inspección y utilizar el guión de inspección (INS). En la forma REPINS deben registrar todos los defectos encontrados y el tiempo dedicado. El objetivo de la inspección es encontrar problemas e inconsistencias antes de iniciar la labor de diseño. El equipo decide quién corregirá los problemas identificados, tales correcciones se hacen después de la inspección.

Actualización de requerimientos

Después de que se aplicó la inspección en los requerimientos y se obtuvo la aprobación, se deben actualizar los documentos de ERS y el Plan de Pruebas del Sistema de acuerdo a los problemas encontrados durante la inspección. Conviene asegurarse que la definición de necesidades tenga un seguimiento mediante referencias etiquetadas, las cuales serán útiles en el diseño. Las secciones corregidas se proporcionan al AD, quien las reúne para obtener el documento final de ERS y así distribuir copias a todo el equipo y al instructor.

Revisión de ERS por parte del usuario

Después de producir, inspeccionar y corregir la ERS y el Plan de Pruebas del Sistema, los usuarios finales deben leer la ERS y estar de acuerdo en que describe lo que ellos quieren. Esto es una base para generar un producto de calidad y es una línea base contra la negociación de cambios en los requerimientos. Después de la revisión se deben hacer los cambios y distribuir la ERS aprobada.

Línea base de los requerimientos

El AA declara línea base el documento ERS, el equipo puede solicitar cambios solamente utilizando el procedimiento de control de cambios mediante la forma CAMC. Debido al acuerdo con el usuario, en la ERS ya se tiene una línea base firme para negociar el costo de cambios.

Criterio de éxito

Como conclusión en la fase de requerimientos, el equipo debe tener

- El documento ERS completado, inspeccionado y actualizado y el Plan de Pruebas del Sistema bajo control de configuración.
- Los datos de tiempos personales, defectos y tamaños registrados en las formas de TSPi.
- La forma de inspección completada con la inspección de los requerimientos.
- Copias de todas las formas, de la ERS y del Plan de Pruebas del Sistema en la carpeta del proyecto.

A continuación se muestran los guiones correspondientes a esta fase.

Objetivo		Guiar al equipo en el desarrollo e inspección de los requerimientos durante el primer ciclo de un proyecto desarrollado en equipo
Criterio de entrada		<ul style="list-style-type: none"> • El equipo tiene una estrategia de desarrollo y un plan. • Los estudiantes han leído el Capítulo 6, las secciones de pruebas del Capítulo 9 y la definición de necesidades.
General		<p>El proceso de desarrollo de requerimientos produce la Especificación de Requerimientos de Software (ERS), donde se definen:</p> <ul style="list-style-type: none"> • Las funciones del producto. • Las descripciones de los casos de uso para funciones normales y anormales. <p>El equipo debe tener cuidado al incrementar los requerimientos.</p> <ul style="list-style-type: none"> • Sin experiencia en aplicaciones similares, parecerá que funciones simples pueden necesitar más trabajo que el esperado. • Conviene añadir funciones en incrementos pequeños. • Si hay tiempo suficiente, se añaden incrementos.
Paso	Actividades	Descripción
1	Panorama en el proceso de requerimientos	<p>El instructor describe el proceso de requerimientos y sus productos.</p> <ul style="list-style-type: none"> • Cómo realizar el proceso de requerimientos. • Cómo realizar y documentar la inspección de requerimientos.
2	Revisión de la definición de necesidades	<p>El AD dirige al equipo en la revisión de la definición de necesidades del producto y en el desarrollo de las preguntas que se harán al instructor acerca de:</p> <ul style="list-style-type: none"> • Las funciones que se realizarán en las diferentes versiones del producto. • El uso de las funciones.

3	Aclaración en la definición de necesidades	El AD proporciona preguntas al instructor, quién discute las respuestas con el equipo.
4	Tareas de los requerimientos	El AD dirige al equipo en: <ul style="list-style-type: none"> • La creación del documento de ERS.
5	Asignación de tareas	El LE ayuda a asignar las tareas entre los integrantes del equipo y a comprometerse en las fechas de entrega.
6	Documentación de los requerimientos	Cada integrante del equipo: <ul style="list-style-type: none"> • Produce y revisa la sección asignada de ERS. • Proporciona tal sección al AD. El AD produce el borrador de ERS.
7	Plan de Pruebas del Sistema	El AD dirige al equipo en la generación y revisión del Plan de Pruebas del Sistema.
8	Inspección de los requerimientos y del Plan de Pruebas del Sistema	El ACP dirige al equipo a: <ul style="list-style-type: none"> • Inspeccionar el borrador de ERS y el Plan de Pruebas del Sistema (ver guión INS). • Identificar preguntas y problemas. • Definir quién resolverá cada pregunta, problema y cuándo. • Documentar la inspección en la forma INS.
9	Actualización de requerimientos	El AD obtiene las secciones actualizadas de ERS y <ul style="list-style-type: none"> • Las une en un solo documento. • Verifica el seguimiento en la definición de necesidades u otras fuentes.
10	Revisión de ERS por parte del usuario	<ul style="list-style-type: none"> • El AD proporciona una copia de la ERS final al instructor para su aprobación. • Después de la aprobación, el equipo arregla los problemas identificados.
11	Los requerimientos como línea base	<ul style="list-style-type: none"> • El AA integra el documento ERS como línea base.
Criterio de éxito		<ul style="list-style-type: none"> • Los documentos ERS y PPS terminados e inspeccionados. • La forma INS terminada a partir de la inspección de requerimientos. • Datos de tiempos, defectos y tamaños registrados en las formas. • La carpeta del proyecto actualizada.

Tabla 6.2 Guión de desarrollo de requisitos para el primer ciclo, REQ1

Objetivo		Guiar al equipo en la actualización e inspección de los requerimientos para un segundo o ciclos de desarrollo posteriores
Criterio de entrada		El equipo tiene una estrategia de desarrollo y un plan actualizado
General		<p>Actualice la especificación de requerimientos de software para reflejar:</p> <ul style="list-style-type: none"> • Problemas de los requerimientos en los ciclos anteriores. • Funciones de ERS previamente especificadas que no fueron desarrolladas. • Funciones de ERS previamente no especificadas que actualmente no son requeridas. <p>El equipo debe tener cuidado al incrementar los requerimientos.</p> <ul style="list-style-type: none"> • Sin experiencia en aplicaciones similares, parecerá que funciones simples pueden necesitar más trabajo que el esperado. • Conviene añadir funciones en incrementos pequeños. • Si hay tiempo suficiente, añada incrementos más adelante. <p>La ERS actualizada define funciones nuevas del producto, incluyendo descripciones de casos de uso añadidas con acciones del usuario normales y anormales.</p>
Paso	Actividades	Descripción
1	Consideraciones en la actualización de requerimientos	El instructor describe problemas con el proceso de requerimientos anteriores que deberían corregirse en este ciclo.
2	Revisión de la definición de necesidades	El AD dirige al equipo en la revisión de la definición de necesidades del producto y en la formulación de preguntas nuevas acerca de: <ul style="list-style-type: none"> • Las funciones que se realizarán en cada versión del producto. • Cómo se van a usar esas funciones.

3	Aclaración en la definición de necesidades	El AD proporciona preguntas al instructor, quién discute las respuestas con el equipo.
4	Actualización de tareas	El AD dirige al equipo a: <ul style="list-style-type: none"> • Identificar los cambios en requerimientos. • Actualizar las asignaciones funcionales de componentes.
5	Asignación de tareas	El LE ayuda a asignar las tareas entre los integrantes del equipo y a comprometerse en las fechas de entrega.
6	Actualización de la documentación	<ul style="list-style-type: none"> • Cada integrante del equipo actualiza y revisa su parte de la ERS y la proporciona al AD. • El AD produce la ERS actualizada.
7	Plan de Pruebas del Sistema	El AD dirige al equipo en la actualización y revisión del Plan de Pruebas del Sistema.
8	Actualización de inspección	El ACP dirige al equipo a: <ul style="list-style-type: none"> • Inspeccionar el borrador de ERS y el Plan de Pruebas del Sistema (ver guión INS). • Identificar preguntas y problemas. • Definir quién resolverá cada pregunta, problema y cuándo. • Documentar la inspección en la forma INS.
9	Actualización de requerimientos	El Administrador de Desarrollo <ul style="list-style-type: none"> • Une las secciones para integrar la ERS actualizada. • Verifica que se haga el seguimiento en la definición de necesidades u otras fuentes.
10	Revisión de ERS por parte del usuario	<ul style="list-style-type: none"> • El AD proporciona una copia de la ERS final al instructor (usuario) para su aprobación. • Después de la aprobación, el equipo arregla los problemas identificados.
11	Los requerimientos como línea base	<ul style="list-style-type: none"> • El AA integra el documento ERS como línea base
Criterio de éxito		<ul style="list-style-type: none"> • Los documentos ERS y PPS actualizados e inspeccionados. • La forma INS terminada a partir de la inspección de requerimientos. • Datos de tiempo, defecto y tamaño registrados en las formas. • La carpeta del proyecto actualizada.

Tabla 6.3 Guión de desarrollo de requisitos para ciclos posteriores, REQn

Capítulo 7. Diseñando en equipo

Objetivo: Proporcionar un diseño completo y de alta calidad que se utilice como base para la fase de Implementación.

Este capítulo se enfoca en los principios de diseño y su proceso; los cuales se pueden utilizar con cualquier método de diseño que parezca más apropiado. En TSPi, la fase de diseño se enfoca a la estructura completa del sistema. Aquí se produce el documento Especificación de Diseño de Software (EDS), el cual documenta el diseño de alto nivel (DAN). En la fase de implementación se hace referencia al diseño detallado. Este capítulo incluye los guiones DIS1 y DISn.

En equipo se puede trabajar más rápido dividiendo el producto en partes componentes y que cada uno de los integrantes diseñen e implementen una o más de esas partes.

7.1. Principios de diseño

El diseño es el proceso creativo mediante el cual se decide cómo construir un producto, debe contener una especificación completa y precisa de la construcción del producto. Un diseño completo define las partes principales de un producto, describe cómo esas partes interactúan y especifica cómo unirlos para producir el resultado final.

El diseño de alto nivel difiere del diseño detallado y de la implementación sólo en el alcance y detalle. El diseño de alto nivel (DAN) tiene que producir una especificación que varios ingenieros puedan usar para que eso les permita diseñar cada parte independientemente de los demás.

Cuando el diseño de alto nivel es completo y preciso, los ingenieros rápidamente pueden producir los diseños detallados de los componentes; para eso, necesitan conocer las especificaciones funcionales completas de cada componente, sus interfaces y comportamientos de estado. Entonces, para generar el producto final, los ingenieros de implementación necesitan un diseño detallado que defina la estructura lógica, las condiciones de inicialización y de cada paso, la estructura de estado detallada y las transiciones de estado para cada programa. Entonces, los ingenieros de implementación producen el código que implementa el diseño.

Su objetivo es que el código fuente realice correctamente todas las funciones especificadas, que utilice adecuadamente todas las facilidades del sistema, que incorpore las funciones de reuso disponibles y que siga todos los estándares de codificación y del sistema y sus convenciones. El producto final implementado tiene que ser un programa fuente que se compile y ejecute sin errores o problemas.

En TSPi, la primera actividad de diseño ocurre durante la fase de estrategia, cuando se produce el diseño conceptual, el cual es la base para definir la estrategia de desarrollo y producir el plan. Después de las fases de estrategia, planeación y requerimientos continua la fase de diseño, donde los guiones DIS1 y DISn guían a través del proceso de diseño de alto nivel.

7.2. Diseñando en equipo

Al trabajar en equipo surgen las siguientes preguntas: ¿quién debería diseñar cada parte?, ¿en qué orden deberían hacer el trabajo? y ¿cómo se unirán las partes?.

Usando el equipo completo

Un problema común en el diseño de sistemas grandes de software es la necesidad de definir la estructura completa del sistema antes de especificar cualquier otra cosa.

Se recomienda que todo los integrantes del equipo colaboren en el diseño de estructura completo, también que se identifiquen tareas que el resto de los integrantes puedan hacer, mientras uno o

dos de los ingenieros definen esa estructura y especifican los componentes del sistema con suficiente detalle para poder obtener un diseño especificado completamente.

Otras tareas que los ingenieros pueden hacer consisten en realizar estudios de diseño, desarrollar estándares y definir cómo aplicar el reuso.

Estudios de diseño

Los estudios de diseño se tienen que iniciar con ideas preliminares sobre los componentes y las funciones del producto, las cuales generalmente, se definen en la primera sesión de diseño del equipo. Mientras los diseñadores del sistema producen las especificaciones de los componentes externos, otros de ellos pueden identificar las alternativas posibles para diseñar esos componentes y construir prototipos.

Usando los talentos del equipo

El problema más crítico en equipos de trabajo es la contribución de todos sus integrantes. Cuando se trabaja en equipo, en ocasiones existe un rechazo a ofrecer sugerencias e ideas; principalmente, en equipos de software este es un gran problema porque las decisiones de diseño mayores tienen que establecerse desde un principio.

Cada integrante debería contribuir con su experiencia y conocimiento. La persona que dirige la reunión en equipo debería hacer pequeñas pausas para preguntar si alguien tiene ideas o conocimientos sobre los temas discutidos. Los equipos que siguen estos consejos son más productivos.

7.3. Estándares de diseño

Existen varios estándares de diseño, de los cuales se consideraron importantes los siguientes:

- **Convenciones en nombramiento.**
Es el primer estándar en el cual se especifica la estructura de nombramiento. El AA debe establecer un glosario del sistema. Se deben definir los nombres de tipos de programa jerárquicos (tales como sistema, producto, componente, módulo u objeto), las convenciones usadas en nombres de programas, archivos, variables y parámetros; y los procedimientos para establecer, controlar y cambiar nombres.
- **Formatos de interfaz.**
Aquí se definen los formatos y el contenido de las interfaces de los componentes. Esto consiste en indicar parámetros para variables, códigos de error u otras condiciones. Cuando las interfaces están especificadas consistentemente se tendrán menos errores y se localizarán rápidamente durante las revisiones e inspecciones.
- **Mensajes del sistema y de error.**
Es conveniente establecer formatos estándares y procedimientos para mensajes de error y del sistema. Un sistema útil debe tener consistencia y los mensajes deben ser comprendidos fácilmente.
- **Estándares de diseño.**
Se sugiere utilizar el estándar de tipos de defectos de PSP. Más adelante se muestran en la tabla 7.1.
- **Conteo de LOC.**
Generalmente los estándares de tamaño se definen en la fase de implementación.

- Estándares de representación de diseño.
El estándar de representación del diseño define el producto que se generará a partir del diseño. Es importante definir y utilizar esos estándares porque una representación de diseño imprecisa o ambigua puede dirigir a problemas graves en la implementación y en las pruebas.

No. de tipo	Nombre del tipo	Descripción
10	Documentación	Comentarios, mensajes.
20	Sintaxis	Ortografía, tipos de puntuación, formatos de instrucción.
30	Construcción, paquete	Administración de cambios, bibliotecas, control de versión.
40	Asignación	Declaración, nombres duplicados, alcance, límites.
50	Interfaz	Llamadas de procedimientos y referencias, I/O, formatos de usuario.
60	Revisión	Mensajes de error, revisiones inadecuadas.
70	Datos	Estructura, contenido.
80	Función	Lógica, apuntadores, ciclos, recursión, proceso, defectos de función.
90	Sistema	Configuración, memoria.
100	Ambiente	Diseño, compilación, prueba u otros problemas de soporte del sistema.

Tabla 7.1 Estándares de tipos de defectos de PSP

Estándares de representación de diseño

Con una representación de diseño el equipo determina si realizó o no un diseño completo - por ejemplo, al inicio de cada revisión de diseño o inspección. Por eso, es necesario documentar cada diseño. Una manera de hacerlo es utilizar las siguientes cuatro plantillas de diseño de PSP: Escenario operacional, Especificación funcional, Especificación de estado y Especificación lógica o algunos otros tipos de representaciones.

Esto permite ahorrar tiempo y producir un diseño revisado, el cual pueda ser verificado antes de la implementación.

Escenarios operacionales de PSP o casos de uso

Un escenario describe el comportamiento dinámico visible externamente del programa al describir una secuencia de acciones de entradas y las respuestas del sistema de acuerdo a esas acciones. Durante el diseño de alto nivel se producen escenarios que especifican la secuencia de pruebas para cada una de las funciones claves del programa. Se pueden utilizar al producir los planes de pruebas de integración y del sistema. Los escenarios de pruebas de nivel componente pueden exponer problemas de interfaz, funcionales o de usabilidad.

Análisis de máquinas de estado

Considerando el método de diseño utilizado, un análisis de una máquina de estado puede ayudar a descubrir los asuntos y problemas lógicos complejos y difíciles de detectar. Es útil cuando se tienen dudas sobre el comportamiento de un programa y es particularmente importante cuando los programas se incrementan mediante varios ciclos de desarrollo, puesto que, suele suceder que durante esos incrementos se introducen condiciones imposibles y contradictorias.

Produciendo diseños precisos

Con un diseño definido completamente, rápidamente se puede entender el comportamiento de un programa muy complejo. Los diseños precisos son críticamente muy importantes en los proyectos de equipo porque los diseños superficiales con frecuencia tienen problemas lógicos básicos que ningún ingeniero puede visualizar; esos problemas son difíciles de diagnosticar durante las pruebas y algunas veces no se pueden modificar sin rehacer porciones mayores de la implementación.

7.4. Diseñando para reuso

Para aprovechar el tiempo disponible de los integrantes del equipo durante el diseño de alto nivel se pueden identificar funciones comunes y proponer un conjunto inicial de partes de reuso.

El reuso es una fuente poderosa para incrementar la productividad del equipo. Permite ahorrar tiempo en el primer ciclo de desarrollo y en ciclos posteriores. En proyectos grandes se puede aplicar reuso a partir del desarrollo de requerimientos y de la estrategia. Los resultados principales que se obtienen son la definición en los estándares de interfaces y en las convenciones de nombramiento, para generar estándares de documentación, generar productos de alta calidad y proporcionar apoyo en la aplicación.

Estándares de interfaz de reuso

Uno de los estándares de reuso más importantes son las interfaces de *llamadas return*. En tales estándares se deben especificar cuáles parámetros son usados como variables, cuáles como parámetros de regreso y cuáles como mensajes especiales y condiciones de error. También es conveniente estandarizar mensajes, condiciones de error y formas para reaccionar ante tales condiciones. Finalmente, se aconseja establecer convenciones en el nombramiento de variables y parámetros, así como en las propias partes de reuso. Al desarrollar esas partes en un solo proyecto se tiene la ventaja de un marco de arquitectura común y un conjunto único de estándares del sistema, eso facilita compartir partes entre los ingenieros e incrementa la habilidad para identificar partes y rutinas comunes.

Estándares de documentación de reuso

Los estándares de documentación marcan la diferencia entre las partes de reuso y las que no lo son. Un buen reuso permite que los ingenieros puedan utilizar un programa sin tener que analizar el código fuente. Por eso, las partes de reuso deben contener una especificación completa del comportamiento externo, también es buena idea añadir una sección de comentarios al principio del programa fuente en cada parte.

Calidad en la parte de reuso

La alta calidad es una parte importante en la estrategia de reuso. Para obtener una calidad de nivel alto utilice un proceso definido completamente y conduzca a revisiones personales y a inspecciones entre colegas sobre el diseño y código. Ejecute pruebas unitarias para asegurar que la rutina trabaja bien para todas las variables y valores de los parámetros, esto consiste en hacer pruebas con límites menores, mayores y fuera del rango utilizado. Las partes deberían diseñarse con mensajes de error que identifiquen entradas o condiciones erróneas.

Deberá considerar las siguientes preguntas al decidir reutilizar un programa:

1. ¿El programa tiene las funciones correctas?
2. ¿La interfaz del programa es portable en la nueva aplicación?
3. ¿Es apropiado el desempeño del programa en la nueva aplicación?
4. ¿Todos los materiales necesarios están disponibles: código fuente, casos de pruebas, datos de pruebas e instrucciones de la aplicación?
5. ¿El programa es portable en cuanto a estándares, tales como el nivel del lenguaje; estándares de codificación; estándares en nombramiento y en archivos; y estándares en mensajes y ayuda?
6. ¿Se puede demostrar que el programa es de alta calidad?

Si existe alguna respuesta negativa lo más conveniente es dedicar tiempo considerable al reuso.

Soporte de aplicación

Maximizar el reuso consiste en localizar las partes disponibles y saber cómo usarlas. Una buena ayuda es un índice de las partes con especificaciones claras de cada funcionalidad. En TSPi, el AA es el responsable de proporcionar las partes de reuso y de mantener registros completos de las mismas.

7.5. Diseñando para usabilidad

Para generar productos útiles se deben producir escenarios por cada función clave del usuario, se deben analizar y asegurarse de que reflejan el tipo de sistema que los usuarios desean.

7.6. Diseñando para pruebas

Para proyectos con múltiples ciclos es importante realizar pruebas esmeradamente, desafortunadamente las pruebas unitarias requieren un código de prueba especial para proporcionar un ambiente de pruebas portable. Para algunos productos, el código de pruebas puede ser aproximadamente la mitad de las LOC totales del producto, por esta razón, se debe generar un diseño que produzca el código mínimo requerido.

Pruebas de caja blanca y caja negra

Existen dos tipos de pruebas: caja blanca y caja negra. La primera, verifica las especificaciones externas del programa y no considera su estructura interna. La segunda, considera los caminos y la estructura lógica del programa.

Las pruebas de caja negra a nivel sistema simulan la manera en la cual el usuario va a usar el sistema. Respecto a los módulos o componentes individuales requieren controladores de interfaz, así como programas de soporte especiales para simular un ambiente del sistema.

Las pruebas de caja blanca permiten evaluar cada aspecto de la estructura del programa. Para eso se pueden utilizar herramientas de pruebas de propósito general, pero especialmente un código de pruebas o facilidades de pruebas únicas.

7.7. Revisiones e inspecciones en el diseño

Las revisiones e inspecciones en el diseño ayudan a mejorar su calidad. Para la inspección lo primero que se necesita es tener un diseño bien documentado, y posteriormente preparar la inspección debida. Se tiene que revisar cada elemento del diseño, ya sean, las interfaces; examinar la inicialización de ciclos, cada paso y su terminación; y analizar el comportamiento de los estados.

Otros beneficios de la inspección

Aparte de que permite encontrar defectos, las inspecciones son valiosas por otras razones más. Un equipo de inspección capaz y con experiencia detecta ciertas anomalías en los requerimientos y en el diseño que el autor no podría haber encontrado.

7.8. Los guiones de diseño de TSPi

Los guiones de esta fase son DIS1 y DISn, los cuales se muestran en las tablas siguientes, son similares, excepto que DIS1 describe los pasos de diseño para el desarrollo en el ciclo 1 y DISn describe los pasos para ciclos subsecuentes.

Criterio de entrada

Este criterio requiere una estrategia y un plan de desarrollo terminados, estándares de diseño documentados y el documento de ERS terminado e inspeccionado.

Diseño de alto nivel (DAN)

En el ciclo 1 el equipo produce un diseño completo para todos los ciclos, se refina y documenta cada una de las funciones del primer ciclo. En ciclos posteriores, de nuevo se revisa el diseño y se documentan las funciones para el ciclo siguiente. Para generar el diseño de alto nivel se debe decidir la estructura completa del producto, nombrar los componentes, asignar funciones a los componentes, producir las especificaciones externas de los componentes, asignar funciones de

casos de uso a los componentes e identificar las tareas de diseño que se realizarán. El AD dirige al equipo en los puntos anteriormente mencionados.

Estándares de diseño

El ACP dirige al equipo en la generación de los estándares de diseño y el glosario de nombres.

Estructura completa del producto

Al definir la estructura completa del producto en el ciclo 1, primero se produce el diseño conceptual de alto nivel, en el cual se identifican las partes componentes, se nombran las funciones generales y se especifica la relación entre las mismas. Para producir un diseño se pueden usar gráficas de flujo, casos de uso o algunas otras metodologías. Sin embargo, al final, el equipo debe estar de acuerdo en la estructura de alto nivel, en los nombres de los componentes y en la asignación de funciones del producto entre esos componentes.

Para asignar las funciones del producto a los componentes utilice una tabla de seguimiento similar a la tabla definida en la estrategia. Utilice la forma ESTRA para indicar cada componente y las funciones correspondientes, guíese con los pasos que se mencionan:

- Agrupe las funciones por componentes, junto con las referencias de párrafos de ERS por cada función.
- Asigne una letra a cada componente para hacer referencia a las funciones que generará en cada ciclo.
- Estime las LOC y las horas de desarrollo por función.
- Indique si una función es generada mediante varios componentes.

La tabla también puede utilizarse durante la planeación de pruebas, documentación del programa y mantenimiento del producto.

Asignación de las tareas en la fase de diseño

El AD dirige al equipo en la generación de la Especificación de Diseño de Software (EDS) identificando el trabajo necesario y dividiéndolo en secciones. Con la ayuda del LE se asignan las secciones a los integrantes del equipo y se comprometen en la fecha de entrega.

La especificación de diseño (EDS)

Mientras se produce el documento EDS, defina las interfaces externas y las especificaciones funcionales de cada componente, genere un conjunto completo de escenarios que cubran las funciones externas de cada uno.

El paso final en el diseño de alto nivel es producir los diversos documentos de diseño especificando la lógica y la estructura del programa principal del sistema para asegurar que las funciones de los componentes y las interfaces están completamente especificadas.

Al producir la Especificación de Diseño de Software cada integrante debe revisar su trabajo y arreglar los problemas, después de eso se debe entregar al AD para que lo incorpore en el borrador completo de la EDS y distribuya copias al equipo.

Plan de Pruebas de Integración (PPI)

Es importante producir este documento al realizar las especificaciones de diseño. Con las pruebas de integración se van a revisar y verificar todas las interfaces entre los componentes del sistema, para ello también se debe indicar la manera para hacer las pruebas. Para asegurar que a todas las interfaces se les han aplicado las pruebas es conveniente inspeccionar el Plan de Pruebas de Integración al inspeccionar la EDS.

Inspección de diseño

El ACP dirige al equipo en la inspección del borrador de EDS y del PPI. El equipo completo debería participar en la inspección para encontrar el número máximo de defectos y asegurarse de que todos entienden el diseño. Se debe seguir el guión INS, documentar la inspección en la forma REPINS y registrar los defectos en la forma REGD.

Actualice el diseño

Después de la inspección de diseño, arregle los errores identificados en la EDS y en el PPI, si fuese necesario haga que uno o dos ingenieros revisen las correcciones, después de eso proporcione la EDS y las secciones del plan de pruebas al AD, quien produce y distribuye los documentos terminados.

Línea base del diseño

El AD declara al documento EDS como línea base y obtiene una copia oficial del mismo, el equipo podrá hacer modificaciones a tal documento mediante la forma CAMC.

Criterio de éxito

Los productos del diseño final son los siguientes:

- La Especificación de Diseño de Software terminada, inspeccionada y corregida, incluyendo todos los materiales de diseño necesarios.
- El Plan de Pruebas de Integración terminado, inspeccionado y corregido.
- Los estándares de diseño y el glosario de nombres.
- Las formas de inspección terminadas (REPINS).
- Las formas RESPL y RESCA actualizadas.
- Los datos de los ingenieros en tiempo, tamaño y defectos registrados en las formas debidas.
- Copia de todos los materiales de diseño en la carpeta del proyecto.

Objetivo		Guiar al equipo en el desarrollo e inspección de las especificaciones de diseño para un proyecto de desarrollo en equipo
Criterio de entrada		<ul style="list-style-type: none"> • Tener la estrategia de desarrollo y un plan. • Tener la ERS completada e inspeccionada. • Haber leído este capítulo.
General		<p>El proceso de diseño genera la Especificación de Diseño de Software (EDS), donde se define la estructura completa del producto en el ciclo 1:</p> <ul style="list-style-type: none"> • Especificación de los componentes del producto y sus interfaces. • Asignación de los casos de uso a sus componentes. • Estándares de archivos y mensajes, definiciones y convenciones en nombramiento. • Notación y estándares de diseño.
Paso	Actividades	Descripción
1	Revisión del proceso de diseño	<p>El instructor describe el proceso de diseño y sus productos.</p> <ul style="list-style-type: none"> • Cómo el proceso de diseño se realizó y un ejemplo de EDS. • Cómo la inspección del diseño se realiza y se reporta. • Como se establecieron los estándares y convenciones de diseño.
2	Diseño de alto nivel (DAN)	<p>El AD dirige al equipo a:</p> <ul style="list-style-type: none"> • Definir la estructura del producto en el ciclo 1. • Nombrar los componentes del producto. • Asignar casos de uso a esos componentes. • Identificar las tareas del diseño que se terminarán y documentarán.
3	Estándares de diseño	El ACP produce el glosario de nombres y los estándares de diseño.
4	Tareas de diseño	<p>El AD dirige al equipo a:</p> <ul style="list-style-type: none"> • Establecer el contenido del documento EDS y a identificar las actividades para producirlo.
5	Asignación de tareas	El LE asigna las tareas entre los integrantes del equipo y fija fechas de entrega.
6	La especificación del diseño	<p>Cada integrante del equipo:</p> <ul style="list-style-type: none"> • Genera y revisa la sección correspondiente del documento EDS. • Proporciona tal sección al AD. <p>El AD produce el borrador de EDS.</p>
7	Plan de Pruebas de	El AD dirige al equipo en la generación y pruebas del Plan de Pruebas del

	Integración	Integración.
8	Inspección del Plan de Pruebas de Integración y del diseño	El ACP dirige al equipo a: Inspeccionar el borrador de EDS y el Plan de Pruebas del Integración (ver guión INS), para : <ul style="list-style-type: none"> • Que cada caso de uso sea cubierto y referenciado en el diseño. • Que el diseño esté terminado y corregido. • Que el PPI sea adecuado. • Que cada problema sea arreglado y se fije al responsable de eso. La inspección se registra en la forma REPINS y los defectos en la forma REGD.
9	Actualización del diseño	El AD obtiene las secciones actualizadas de EDS y <ul style="list-style-type: none"> • Las combina en una EDS final. • Verifica el seguimiento con la ERS.
10	El diseño como línea base	<ul style="list-style-type: none"> • El AA integra el documento EDS como línea base.
Criterio de éxito		<ul style="list-style-type: none"> • La EDS y el Plan de Pruebas de Integración terminado e inspeccionado. • Los estándares de diseño y el glosario de nombres. • Las formas RESPL, RESCA y REPINS actualizadas. • La carpeta del proyecto actualizada..

Tabla 7.2 Guión de diseño para el primer ciclo, DIS1

Objetivo		Guiar al equipo en el desarrollo e inspección de las especificaciones de diseño de software para un segundo y ciclos posteriores de desarrollo
Criterio de entrada		El equipo tiene actualizados los documentos: estrategia de desarrollo, el plan y la ERS.
General		El proceso de diseño del ciclo siguiente genera la Especificación de Diseño de Software actualizada (EDS) la cual define: <ul style="list-style-type: none"> • La estructura completa del producto para el segundo o ciclos posteriores. • Componentes del producto nuevos o modificados. • Asignación de nuevos casos de uso en cada componente. • Especificación de la interfaz de componentes actualizada. • Actualización de estándares de archivos y mensajes, definiciones y convenciones en nombramiento. • Notación y estándares de diseño.
Paso	Actividades	Descripción
1	Revisión del proceso de diseño	El instructor describe en el proceso de diseño cíclico: <ul style="list-style-type: none"> • Elementos que se considerarán al incrementar el producto. • Problemas comunes de extensión y • Problemas con el proceso de diseño anterior, métodos o estándares que se corregirán en el ciclo actual.
2	Diseño de alto nivel	El AD dirige al equipo a: <ul style="list-style-type: none"> • Definir la estructura del segundo o ciclos posteriores. • Nombrar los componentes nuevos del producto. • Asignar casos de uso a componentes nuevos y modificados. • Identificar las tareas del diseño que se terminarán y documentarán.
3	Estándares de diseño	El ACP dirige al equipo en la actualización del glosario de nombres y los estándares de diseño.
4	Tareas de diseño	<ul style="list-style-type: none"> • El AD dirige al equipo a establecer el contenido del documento EDS y a identificar las actividades para producirlo.
5	Asignación de tareas	El LE asigna las tareas entre los integrantes del equipo y fija fechas de entrega.
6	La especificación del diseño	Cada integrante del equipo: <ul style="list-style-type: none"> • Produce y revisa la parte que le corresponde en el documento de EDS. • La proporciona al AD. El AD genera el borrador de EDS actualizado.
7	Plan de Pruebas de Integración	El AD dirige al equipo en el desarrollo y revisión del Plan de Pruebas del Integración.

8	Inspección del diseño y del Plan de Pruebas de Integración	<p>El ACP dirige al equipo en la inspección del borrador de EDS actualizado y el Plan de Pruebas de Integración (ver guión INS), para lograr que:</p> <ul style="list-style-type: none"> • Cada caso de uso nuevo sea cubierto y referenciado en el diseño. • Los cambios en el diseño estén terminados y correctos. • El PPI sea adecuado. • Cada problema sea arreglado y se defina al responsable de eso. • La inspección se registre en la forma REPINS y los defectos en la forma REGD.
9	Actualización del diseño	<p>El AD obtiene las secciones actualizadas de EDS y</p> <ul style="list-style-type: none"> • Las combina en una EDS actualizada. • Actualiza los casos de uso mediante referencias. • Presenta la EDS para control de cambios.
10	El diseño como línea base	<ul style="list-style-type: none"> • El AA integra el documento EDS como línea base.
Criterio de éxito		<ul style="list-style-type: none"> • El documento de EDS terminado e inspeccionado en este ciclo. • Las formas RESPL, RESCA y las formas de inspección actualizadas. • El glosario de nombres y la carpeta del proyecto actualizados.

Tabla 7.3 Guión de diseño para ciclos subsiguientes, DISn

Capítulo 8. Implementación del Producto

Objetivo: Describir el proceso de implementación a través del criterio de diseño, estándares de implementación, estrategias de implementación, revisiones e inspecciones.

Para describir el proceso de implementación de TSPi se utilizan los guiones IMP1 e IMPn, los cuales más adelante se describen.

8.1. Criterio de diseño terminado

Antes de iniciar la implementación se revisa que se tenga el diseño de alto nivel completo. Con frecuencia el diseño de alto nivel de grandes sistemas requiere de varias etapas. Primero, se subdivide el sistema en subsistemas, componentes o módulos; a la vez, se aplica el proceso de los guiones DIS1 y DISn (los cuales se muestran en el capítulo referente al diseño); finalmente se deben de tener las especificaciones externas de cada subsistema, componente o módulo y el diseño detallado del nivel lógico del sistema.

Niveles de diseño

Si los subsistemas son bastante grandes se debe repetir el proceso de diseño de alto nivel para cada subsistema o componente, se tienen que realizar las especificaciones externas para los componentes de los subsistemas y el diseño detallado del nivel lógico de cada uno. Dependiendo del tamaño del sistema se harán varias iteraciones y se finalizará con las especificaciones externas de los módulos atómicos de nivel menor en el sistema. Los niveles en sistemas grandes pueden ser sistema, subsistema, producto, componente y módulo.

Este proceso de diseño iterativo continua hasta que se hayan producido las especificaciones externas para los elementos básicos del sistema. Generalmente se recomienda que el tamaño consista de 150 o menos LOC; posiblemente contienen objetos o rutinas de nivel menor que ya han sido desarrolladas, las cuales pueden ser partes de reuso o funciones de bibliotecas disponibles.

Los guiones DIS1 y DISn se deben continuar aplicando hasta que toda la especificación en el nivel menor esté terminada, después de eso es cuando se puede iniciar con la implementación. Contrario a lo anterior, en ocasiones sucede que en sistemas grandes se tienen que implementar algunos elementos del producto antes de terminar de diseñar otros.

Implementación paralela

Con equipos grandes de desarrollo se puede iniciar la implementación de sistemas grandes dividiéndolo en componentes e implementándolos con sus especificaciones externas terminadas e inspeccionadas.

8.2. Estándares de implementación

Con unos cuantos minutos que el equipo dedique a la definición de estándares al inicio del proyecto ahorrará gran cantidad de tiempo. Primero, todos los integrantes deben estar de acuerdo en los estándares necesarios y en su contenido y pedir a uno o dos de los integrantes que realicen el desarrollo de los mismos. El ACP dirige las actividades de definición de estándares.

Los estándares de implementación agregan elementos a los estándares definidos durante la fase de diseño y los extienden. A continuación se tratan los siguientes temas relacionados con los estándares:

Revisión de estándares

Se debe revisar lo siguiente:

- Los estándares de nombres, de interfaces y de mensajes desarrollados durante la fase de diseño para asegurarse de que son apropiados y están siendo utilizados.
- La lista de rutinas de reuso para verificar que estén terminadas y que todos los integrantes del equipo las estén utilizando.
- El glosario de nombres para asegurarse que cada quien está usando el mismo nombre para el mismo producto y que todos los nombres nuevos se van agregando al glosario.
- Los nombres de componentes, los nombres de las variables compartidas, de los parámetros y de los archivos para verificar su consistencia.
- Las interfaces y los mensajes estándares para asegurarse de que ya han sido definidos, registrados en el glosario de nombres y que son conocidos y utilizados por los integrantes del equipo.

Estándares de codificación

Un estándar de codificación común asegura la consistencia en el código del equipo; eso facilita la realización de inspecciones, permite que se hagan de una manera más rápida y que sean efectivas. También facilita que los integrantes del equipo puedan compartir su código.

En un estándar de codificación bien estructurado se define una sección para comentarios, lo cual permite mayor flexibilidad al momento de realizar incrementos en ciclos subsecuentes.

Con estos estándares existe la posibilidad de considerar el reuso en un sistema, lo cual permite ahorro de tiempo para el diseño y la implementación.

Estándares de tamaño

Es necesario definir un estándar de tamaño común en el equipo. Debido a que la mayoría de los proyectos generan diferentes tipos de productos. Además de LOC, TSPi sugiere que se cuenten el número de páginas para medir el tamaño de los documentos generados; pudiéndose contar los documentos referentes a requerimientos (ERS), diseño de alto nivel (EDS), diseño detallado, pantallas y reportes, bases de datos, mensajes, guiones de pruebas y material de pruebas.

Para los requerimientos se pueden aplicar conteos de páginas de texto, párrafos enumerados o frases obligatorias. Esas últimas definen acciones que los productos planeados van a realizar. Generalmente, se utiliza una de estas frases por cada función.

En el diseño de alto nivel las medidas más simples son los conteos de páginas, las líneas de texto o los casos de uso. En el diseño detallado, las líneas en el texto del pseudocódigo. Para programas pequeños se pueden utilizar estimaciones en LOC.

Midiendo los tamaños de otros tipos de productos

La razón principal de las medidas de tamaño es que son útiles para hacer estimaciones y mantener seguimiento del trabajo. Como se asume que para el desarrollo de un producto se requiere de una cantidad significativa de trabajo se debe establecer una medida que se relacione con el tiempo requerido. Si no es posible fijar tal medida, puede utilizar un conteo de los elementos del producto. Por ejemplo, al desarrollar pantallas, se puede utilizar un contador de pantalla y datos históricos de acuerdo al tiempo promedio dedicado al desarrollo. Se dividen los datos en categorías, algunas pantallas tomarán un tiempo de desarrollo promedio mayor que otras. El método utilizado en PSP consiste en dividir los datos en diferentes categorías, ya sea en muy pequeños, pequeños, medios, grandes y muy grandes. Entonces se subdividen los datos históricos en esas categorías y se obtienen los tiempos de desarrollo promedio de cada una. Con esos mismos datos se pueden hacer divisiones de tipos, tales como, datos de entrada, de selección de menú, etc.

Otro ejemplo sería que en la fase de planeación se estimen cuántas pantallas serán requeridas y posteriormente, asignarlas dentro de las categorías mencionadas anteriormente. Mayor

información sobre la estimación de tamaño y el uso de los datos de tamaño se puede encontrar en *A Discipline for Software Engineering, Capítulos 4 y 5 [Humphrey 1995]*.

Estándares de defectos

Aquí en vez de añadir tipos nuevos de defectos se toma el material de pruebas y se hace una lista de todos los defectos. Para clasificar los defectos en tipos, primero es conveniente analizar los datos pertenecientes a los reportes de cada defecto y así identificar las áreas clave. Después de ello se debe encontrar el error en las revisiones de diseño y de código.

La razón al catalogar los defectos en tipos es que eso ayuda a analizar y mejorar el proceso de desarrollo, aunque con la clasificación de tipos se tendrá un volumen grande de datos se deben examinar los reportes para identificar los tipos claves; eso también ayudará a determinar como encontrar tales errores en las revisiones de diseño y de código.

Los estándares de tipos de defectos son útiles si son pequeños. Al agregar nuevos defectos se debe de justificar su necesidad.

Prevención de defectos

Entender las causas de los defectos puede ayudar a prevenirlos. A continuación se mencionan cuatro consejos para la prevención de defectos.

- Aprenda más acerca del lenguaje, ambiente o aplicación.
- Modifique el proceso.
- Utilice herramientas mejores.
- Refina su proceso y utilice métodos mejores.

Enseguida se mencionan algunas maneras para la prevención de defectos.

- Detecte los tipos de defectos que parecen ser los causantes de la mayoría de los problemas. Posiblemente, esos defectos originan la pérdida de la mayor parte del tiempo en las pruebas, son difíciles de diagnosticar y de modificar.
- Examine un conjunto de defectos correspondientes a un tipo para identificar las causas de cada defecto particular y después decida cuáles tomar.
- Cuando detecte un defecto que a su juicio se pueda prevenir haga la modificación al proceso.
- Si este método es efectivo empiece a buscar la siguiente categoría de defecto y proceda de la misma manera.

La clave en la prevención de defectos es buscar maneras de modificar lo que uno hace para prevenir el defecto. Después se incorpora el cambio en el proceso para posteriormente realizar un seguimiento que registre el desempeño del cambio. Si aún los tipos de defectos persisten, analice porque el cambio anterior no funcionó y modifique el proceso.

Siempre que se conozca la causa de un defecto se debe anotar en la sección de comentarios de la forma REGD. No se deben mezclar los tipos de defectos con sus causas en el mismo estándar porque defectos iguales se pueden encontrar en partes distintas y eso permite delinear un rango de datos de defectos.

8.3. La estrategia de implementación

La estrategia de implementación debería realizarse según la estrategia de diseño, eso quiere decir que los programas se deben implementar de acuerdo a la manera en que se diseñan. Para evitar problemas en la implementación y en las pruebas se sugiere tomar en cuenta los siguientes tópicos: Revisiones, Reuso y Pruebas.

Las revisiones en la estrategia de implementación

Al inicio del diseño se realiza un panorama general del sistema, el cual posteriormente se detalla. En las revisiones sucede lo contrario, se inicia trabajando desde el detalle hasta llegar al panorama general. Por ejemplo, al revisar un componente encontrará llamadas a funciones u objetos subordinados. La estrategia que el autor de TSPi ha encontrado más eficiente es aplicar revisiones de funcionalidades desde el nivel más bajo (bottom-up). Primero se deben revisar todos los objetos que se encuentran en el nivel inferior. Después de verificar que el desempeño de esas partes atómicas es el que se indicó en las especificaciones externas pase al nivel anterior (que sería un nivel superior). Cuando se encuentren esos objetos en las revisiones del nivel anterior evite revisarlos de nuevo.

Al seguir esta estrategia, primero, implemente los objetos del nivel más bajo y después las partes de los niveles superiores. Generalmente, esta es la manera más rápida de identificar problemas basándose en las especificaciones de los objetos que se encuentran en el nivel más bajo. Encontrar problemas desde un inicio permite que se puedan modificar antes de que se tenga la implementación adelantada. También ayuda a que el tiempo dedicado a las pruebas sea productivo y que no haya necesidad de volver a trabajar sobre lo mismo.

Como el nivel más bajo contiene partes atómicas que son muy fáciles de volver a utilizar, se dice que la estrategia de implementación de nivel más bajo facilita el reuso.

El reuso en la estrategia de implementación

Al utilizar prácticas de implementación simples se pueden crear programas que se caractericen por la facilidad de reuso. Por ejemplo, establecer un estándar de comentarios en el encabezado, lo cual consiste en agregar una sección que se encuentre en el inicio del código fuente el cual deberá contener el nombre del programa; la descripción del alcance; la especificación en los formatos de llamadas y regresos; el nombramiento de variables y parámetros; identificación de rangos de variables, tipos y formatos; la descripción de otras condiciones y excepciones, como restricciones en el valor de las variables y condiciones de error. Todas las restricciones se deberán escribir en letras mayúsculas, estilo negrita para que no pasen desapercibidas. El reuso puede ser de mejor calidad si se discuten los planes de implementación en reuniones de equipo diarias, en ellas se revisan las bibliotecas de reuso y los ingenieros que están implementado pueden averiguar si alguien tiene algún objeto o rutina de reuso que ellos estén necesitando, o saber, si algunas de las partes que están desarrollando son útiles a los demás. Si es así, el AA las debe añadir a la lista de partes de reuso con sus especificaciones funcionales; todo eso permite utilizar de la mejor manera el tiempo dedicado a la implementación.

Las pruebas en la estrategia de implementación

Es importante que antes de implementar se haya revisado la estrategia de desarrollo para asegurarse que se están implementando programas de acuerdo al orden que se estableció en los planes de pruebas.

8.4. Revisiones e inspecciones

En esta sección se discuten las revisiones de implementación y sus inspecciones

Defectos aleatorios

Muchos de los defectos de implementación se introducen al momento de escribir el código, a tales defectos se les llaman errores aleatorios, los cuales no tienen ningún fundamento lógico. Tales defectos de sintaxis no son detectados por el compilador.

Impacto en las pruebas

Los defectos aleatorios pueden causar gran impacto, sobre todo en sistemas grandes. Además son difíciles de detectar al momento de realizar pruebas, porque sólo se encuentran los errores que están bajo las condiciones de pruebas específicas. Para ello, se deberán probar todos los caminos lógicos del programa y probar con los valores de datos y condiciones de operación.

La dificultad de pruebas exhaustivas

El problema de los defectos aleatorios es que pueden originar problemas lógicos en los programas. Dicho de otra forma, al tener errores lógicos, la programación sería mucho más simple y las pruebas más efectivas. En programas complejos, la probabilidad de encontrar defectos aleatorios mediante pocas pruebas es muy pequeña. Por esa razón, es que las revisiones e inspecciones son importantes, porque un revisor en vez de buscar valores individuales y caminos lógicos, puede considerar todos los caminos y valores lógicos en determinado segmento de un programa. Por lo anterior, se dice que las revisiones e inspecciones son más eficientes que las pruebas y porque los revisores suelen encontrar defectos aleatorios en los programas. Se aconseja que los revisores utilicen listas de verificación en revisiones de código personales y las actualicen después de cada inspección.

Inspecciones de diseño en programas fuentes

Para asegurar la calidad del diseño, es conveniente que los documentos correspondientes se revisen e inspeccionen y se actualicen antes de la programación. Al hacer eso, no es necesario que se reinspeccione el diseño durante la implementación, a menos de que se hayan aplicado modificaciones al mismo.

8.5. Los guiones de la Implementación

A continuación se muestran los guiones IMP1 e IMPn, para la implementación en el primer ciclo y en ciclos posteriores.

Criterio de entrada

Para la implementación en el primer ciclo se debe de tener el plan de desarrollo y la estrategia terminados; las especificaciones de requerimientos y de diseño terminadas, revisadas y actualizadas; un estándar de codificación definido y documentado; copias disponibles en las cuales se integren la lista de especificación funcional del producto, el glosario de nombre y otros estándares que el equipo haya establecido.

El criterio de entrada en ciclos posteriores requiere el plan de desarrollo y la estrategia actualizados, las especificaciones de requerimientos y de diseño actualizadas, y las versiones actuales del glosario de nombres y estándares del equipo.

Planeando la implementación

Se deben asignar tareas a cada ingeniero. Lo mejor es que los integrantes que realizaron el diseño sean quienes realicen la implementación, lo cual se deja a criterio del equipo. Después de hacer tales asignaciones, cada quien planea la implementación. Para aquellas tareas que requieren pocas horas, la estimación en tiempo puede ser muy simple. Para tareas mayores, se aconseja desarrollar un plan como el de PSP, utilizando las formas RESPL y RESCA. Para aquellas tareas que no requieren programación, basta con hacer un plan utilizando la forma RESTAREA. Después actualice las formas TAREA y CALENDARIO para reflejar los detalles nuevos de la planeación y generar un plan con nuevo valor ganado. Después el AP deberá actualizar el plan del equipo.

Diseño detallado y revisión en diseño

A partir de aquí se desarrolla el diseño detallado para la implementación. Después se realiza la revisión personal de diseño. Se revisa con una lista de verificación, se revisan los ciclos y la lógica con tablas de seguimiento o rastreo de código (trace tables) o análisis en máquinas de estado.

Desarrollo de pruebas

Después de arreglar los problemas que se encontraron en la revisión, se desarrolla el código para pruebas unitarias. Generalmente, durante el desarrollo de las pruebas se encuentran más problemas del diseño que en la inspección de diseño o en pruebas unitarias, por eso es conveniente aplicar las pruebas antes de la inspección en el diseño detallado. La aplicación de la prueba debe basarse en el plan de pruebas e incluir revisiones de decisiones lógicas, caminos lógicos, detección en ciclos y condiciones de terminación. Se deberían revisar todas las variables y

parámetros desde los límites inferior hasta superior, y los que no se encuentran dentro de ese rango también. Finalmente, debe anticipar todas las condiciones de error posibles.

Inspección en diseño detallado

Después de la aplicación de pruebas, el ACP dirige la inspección del diseño detallado. Los datos se deben registrar en la forma REPINS y los defectos mayores en REGD. Se deben dar copias de tales formas al ACP y al LE para integrarlas en la carpeta. Si deciden realizar máquinas de estado y análisis de tablas para la inspección del código en un mismo segmento, es mejor que lo apliquen ingenieros distintos, porque así es mayor la probabilidad de encontrar problemas.

Revisiones en codificación y en código

Después de la inspección de diseño, se realiza la codificación. Antes de la revisión personal se calcula el número de defectos inyectados, eso lo puede realizar revisando datos históricos de defectos. Para eso, se propone que anote el número usual de defectos inyectados por hora de codificación y por miles de líneas de código.

Para la revisión de código, se aconseja estimar el 50% o 75% (preferentemente el 75%) del tiempo dedicado a la codificación. Se revisa la consistencia en nombres, inicializaciones, puntuaciones, apuntadores, llamadas de secuencias y de inclusión (include). Lo mejor es utilizar las listas de verificación como guía. Se lleva control sobre el tiempo dedicado a la búsqueda de defectos. Si no se encuentran defectos, pida a algún otro ingeniero que revise el código antes de que lo compile.

Inspección en código

Después de la compilación se compara el tiempo dedicado al diseño y su revisión, al código y su revisión con los tiempos del plan de calidad del equipo. También se revisan los niveles de defectos y las tasas de los mismos. Se aconseja utilizar la Tabla Criterio Estándar de Calidad (incluida en el Anexo C).

- El tiempo dedicado al diseño debería ser mayor que el tiempo de la codificación.
- El tiempo dedicado a la revisión del diseño debería ser mayor que el 50% del tiempo del diseño.
- El tiempo dedicado a las revisiones de código debería ser mayor que el 50% del tiempo de codificación y mayor que el 75%.
- Por lo menos, se encontraron dos veces un número similar de defectos en la revisión de código como en la compilación.
- Se encontraron más de 3 defectos por hora de revisión.
- La tasa de revisión fue menor a las 200 LOC por hora.

Si la tasa de calidad del programa consiste en un 70% sólo un ingeniero puede hacer la revisión del código. Si la tasa ha sido menor a ese porcentaje, es conveniente que el ACP dirija la inspección con dos o más ingenieros. Se utiliza el guión INS y las líneas guías del apéndice C (el cual no se muestra en este trabajo). Si persisten los problemas en la calidad, se aconseja que uno o más ingenieros hagan otra revisión antes de arreglar los defectos. Si aún persisten, se hace de nuevo otra revisión hasta que cada defecto haya sido encontrado por dos ingenieros, mínimamente.

Pruebas unitarias

Después de la inspección de código, el siguiente paso es ejecutar las pruebas unitarias. Para eso se utilizan los materiales de pruebas que hayan sido desarrollados y se sigue el plan de pruebas preparado durante el diseño detallado.

Revisión en la calidad del componente

Después de la inspección de código y de pruebas unitarias, el ACP revisa los datos del componente para determinar si cumple con los criterios establecidos en el plan de calidad del equipo, y de acuerdo a eso poder incluirlo como línea base. El plan de calidad contiene la tabla en la que se integran datos sobre niveles de defectos, tasas de defectos, tiempos de proceso y tasas de tiempo. Si existen problemas de calidad en esas áreas el equipo debe decidir volver a hacer las

inspecciones o la programación de nuevo. A continuación se muestra la siguiente figura con un criterio de sugerencia.

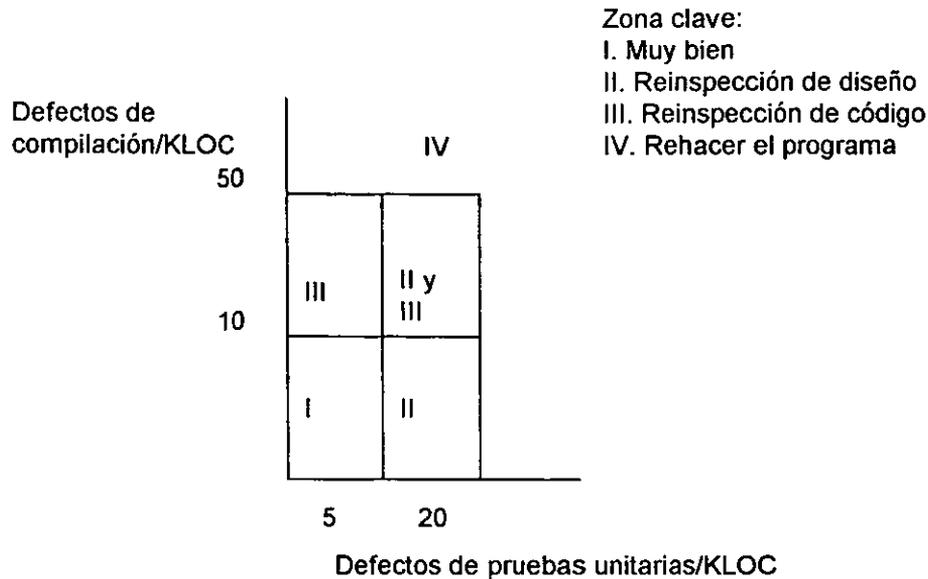


Figura 8.1 Mapa de Calidad de componentes

La zona II consiste en realizar otra vez la inspección en el diseño; se pueden utilizar las tablas de seguimiento o rastreo, tablas de ejecución o máquinas de estado. Eso lo pueden hacer varios revisores y que cada uno tome un segmento del programa y lo revise esmeradamente.

La zona III consiste en realizar inspecciones de código, aplicando el procedimiento descrito anteriormente.

La zona IV consiste en que se haga de nuevo el programa y que se reescriba cualquier código que parezca dudoso.

El objetivo de esta revisión de calidad por componente asegura que cada segmento lógico se analice por lo menos una vez con una tabla de ejecución o tabla de seguimiento o rastreo del código. Se aconseja que si la lógica es compleja, la revisión la realicen dos o más ingenieros para el mismo segmento.

Liberación del componente

Después de que el componente ha reunido el criterio de calidad, se entrega al AA para integrarlo en la línea base.

Criterio de éxito

Para finalizar la fase de implementación se deben tener los componentes terminados, inspeccionados y revisados; los mismos deben estar bajo control de configuración; las formas REPINS y REGD de las inspecciones de diseño, inspecciones de código y reinspecciones completadas; las formas RESPL, RESCA y RESTM del sistema y de todos sus componentes actualizadas; los planes de pruebas unitarias y materiales de apoyo; los datos de defectos en tamaño y tiempo; y la carpeta del proyecto actualizada.

Objetivo		Guiar al equipo en la implementación e inspección del software durante el primer ciclo del proyecto de desarrollo en equipo
Criterio de entrada		<ul style="list-style-type: none"> • El equipo tiene una estrategia de desarrollo y un plan. • Los documentos de especificación de requerimientos y de especificación de diseño, y el glosario de nombres. • El código documentado y otros estándares. • Los estudiantes han leído el Capítulo 8.
General		<p>El proceso de implementación genera un producto revisado, inspeccionado y al cual se le han aplicado pruebas unitarias, con lo cual debería:</p> <ul style="list-style-type: none"> • Cubrir completamente las funciones de los documentos Especificación de Requerimientos de Software y Especificación de Requerimientos de Diseño de Software, así como los casos de uso. • Estar conforme a lo establecido en los estándares de codificación y de diseño. • Seguir los procesos PSP2.1 o PSP3.
Paso	Actividades	Descripción
1	Panorama en el proceso de implementación	<p>El instructor describe el proceso de implementación incluyendo:</p> <ul style="list-style-type: none"> • La importancia en una implementación de calidad. • La necesidad y el contenido de los estándares de codificación. • La estrategia en el manejo de componentes con calidad baja.
2	Planeando la implementación	<p>El AD dirige al equipo de trabajo a:</p> <ul style="list-style-type: none"> • Definir y planear las tareas de implementación (RESPL, RESCA).
3	Asignación de tareas	<p>El LE ayuda a asignar las tareas entre los integrantes y ha fijar las fechas de terminación.</p>
4	Diseño detallado	<p>Los ingenieros producen el diseño detallado.</p> <ul style="list-style-type: none"> • Hacen la revisión de diseño utilizando métodos de revisión de diseño esmerados. • Completan las formas REGD y REGT.
5	Plan de pruebas unitarias	<p>Los ingenieros producen el plan de pruebas unitarias.</p>
6	Desarrollo de pruebas	<p>Los ingenieros se basan en el guión de pruebas unitarias (PU) para desarrollar los casos de pruebas unitarias, procedimientos de pruebas y datos de pruebas.</p>
7	Inspección de diseño detallado	<p>El ACP dirige al equipo en la inspección del diseño detallado de cada componente (guión INS, formas REPINS y REGD).</p>
8	Código	<p>Los ingenieros producen el código fuente de cada componente.</p> <ul style="list-style-type: none"> • Se hace la revisión de código usando una lista de verificación personal. • Se compila y arregla el código hasta que compile sin un error. • Se completan las formas REGD y REGT.
9	Inspección de código	<p>El ACP dirige al equipo en la inspección de código de cada componente (guión INS, formas REPINS y REGD).</p>
10	Pruebas unitarias	<p>Los ingenieros basándose en el guión de pruebas unitarias conducen dichas pruebas y completan las formas REGD y REGT.</p>
11	Revisión en la calidad de componentes	<p>El ACP revisa los datos de cada componente para determinar si reúne los criterios de calidad del equipo.</p> <ul style="list-style-type: none"> • Si es así, el componente es aceptado para las pruebas de integración. • Si no, el ACP recomienda que: • Otra vez el producto sea inspeccionado y que se apliquen las modificaciones pertinentes. • Se descomponga en piezas y sea desarrollado de nuevo.
12	Liberación de componentes	<ul style="list-style-type: none"> • Cuando los componentes son inspeccionados e implementados en forma satisfactoria los ingenieros lo entregan al AA. • El AA registra los componentes en el sistema de administración de configuración.
Criterio de éxito		<ul style="list-style-type: none"> • Componentes terminados, inspeccionados y bajo control de configuración.

	<ul style="list-style-type: none"> • Las formas REPINS de diseño e inspecciones de código completadas. • Planes de pruebas unitarias y materiales de apoyo. • Las formas RESPL, RESCA, RESTM, REGD y REGT actualizadas. • La carpeta del proyecto actualizada.
--	--

Tabla 8.1 Guión de Implementación para el primer ciclo, IMP1

Objetivo	Guiar al equipo en la implementación e inspección de software para el segundo o ciclos posteriores.	
Criterio de entrada	El equipo ha actualizado la estrategia y el plan de desarrollo. <ul style="list-style-type: none"> • Los documentos de Especificación de Requerimientos de Software, Especificación de Diseño de Software y el glosario de nombres. 	
General	El proceso de implementación genera productos que tienen que: <ul style="list-style-type: none"> • Estar revisados, inspeccionados y a los cuales se le han aplicado pruebas unitarias. • Cubrir completamente las funciones de los documentos Especificación de Requerimientos de Software, Especificación de Diseño de Software y casos de uso. • Estar realizados de acuerdo a los estándares de codificación y de diseño. 	
Paso	Actividades	Descripción
1	Revisión del proceso de implementación	El instructor describe el proceso de implementación incrementado: <ul style="list-style-type: none"> • Los elementos que se deben considerar al incrementar un producto. • Problemas de incremento común. • Cualquier problema con el proceso de implementación anterior, métodos o estándares que se van a corregir en el ciclo actual.
2	Planeando la implementación	El AD dirige al equipo de trabajo a: <ul style="list-style-type: none"> • Definir y planear las tareas de implementación (RESPL y RESCA).
3	Asignación de tareas	El LE ayuda a asignar las tareas entre los integrantes y ha fijar las fechas de terminación.
4	Diseño detallado	Los ingenieros producen el diseño detallado. <ul style="list-style-type: none"> • Hacen la revisión de diseño utilizando métodos de revisión de diseño esmerados. • Completan las formas REGD y REGT.
5	Plan de pruebas unitarias	Los ingenieros producen el plan de pruebas unitarias.
6	Desarrollo de pruebas	Los ingenieros se basan en el guión de pruebas unitarias (PU) para desarrollar los casos de pruebas unitarias, procedimientos de pruebas y datos de pruebas.
7	Inspección de diseño detallado	El ACP dirige al equipo en la inspección del diseño detallado de cada componente (guión INS, formas REPINS y REGD).
8	Código	Los ingenieros producen el código fuente de cada componente. <ul style="list-style-type: none"> • Hacen la revisión de código usando una lista de verificación personal. • Compilan y arreglan el código hasta que compile sin error. • Completan las formas REGD y REGT.
9	Inspección de código	El ACP dirige al equipo en la inspección del código de cada componente (guión INS, formas REPINS y REGD).
10	Pruebas unitarias	Los ingenieros basándose en el guión de pruebas unitarias dirigen dichas pruebas y completan las formas REGD, RESPL y RESCA.
11	Revisión en la calidad de componentes	El ACP revisa los datos de cada componente para determinar si reúne los criterios de calidad del equipo. <ul style="list-style-type: none"> • Si es así, el componente es aceptado para las pruebas de integración. • Si no, el ACP recomienda que: <ul style="list-style-type: none"> • Otra vez el producto sea inspeccionado y que se apliquen las modificaciones pertinentes. • Se descomponga en piezas y sea desarrollado de nuevo.

12	Liberación de componentes	<ul style="list-style-type: none"> • Cuando los componentes son inspeccionados e implementados en forma satisfactoria los ingenieros lo entregan al AA. • El AA registra los componentes en el sistema de administración de configuración.
Criterio de éxito		<ul style="list-style-type: none"> • Componentes terminados, inspeccionados y bajo control de configuración. • Las formas de inspecciones REPINS del diseño y código completadas. • Planes de pruebas unitarias y materiales de apoyo. • Las formas RESPL, RESCA, RESTM, REGD y REGT actualizadas. • La carpeta del proyecto actualizada.

Tabla 8.2 Guión de Implementación para ciclos posteriores, IMPn

Capítulo 9. Pruebas de Integración y del Sistema

Objetivo: Integrar las pruebas y la documentación de las mismas. Indicar los objetivos, las estrategias y planeación de las pruebas.

9.1. Principios de las pruebas

En TSPi, el propósito de las pruebas consiste en evaluar el producto y no en modificarlo. Los defectos debieron haberse detectado y arreglado antes de la fase de pruebas. La calidad de un producto se determina cuando se desarrolla.

9.2. Estrategia de pruebas de TSPi

TSPi tiene como objetivo verificar que los productos a los que se les aplican las pruebas son de alta calidad. Las actividades principales son:

1. Utilizar las partes desarrolladas y a las que se les ha aplicado pruebas unitarias para construir el sistema.
2. Utilizar las pruebas de integración para verificar que el sistema está debidamente construido, que todas las partes se encuentran desarrolladas y que funcionan en conjunto.
3. Validar con las pruebas del sistema que el producto realiza los requerimientos del sistema.

Paralelamente se deben:

1. Identificar módulos o componentes de baja calidad y entregarlos al ACP para su evaluación y depuración.
2. Identificar componentes de baja calidad que estén causando problemas, aún después de que se han depurado; regresarlos al ACP para que trabaje de nuevo con ellos o los reemplace.

Los pasos de las pruebas son construcción, integración, pruebas de integración y pruebas del sistema. La construcción integra las partes del sistema en uno completo y listo para aplicarle pruebas, lo cual se llama sistema de construcción. Las pruebas de integración aseguran que todas las partes se hayan incluido en el sistema y que las interfaces trabajen en conjunto. Finalmente, las pruebas del sistema, validan las funciones y desempeño contra los requerimientos.

9.3. La estrategia de construcción e integración

La estrategia de integración define la propuesta de las pruebas de integración; verifica que todos los componentes estén presentes y que las llamadas y otras interacciones trabajen adecuadamente. La estrategia de integración depende de la estrategia de desarrollo, porque depende del trabajo realizado. Por eso, el plan de implementación es importante, porque asegura que las partes implementadas del sistema sirvan como base para la integración.

La estrategia Big-Bang

Con las partes que están disponibles, se puede proceder de diferentes maneras. La propuesta consiste en reunir todas las partes y verificar que trabajen en conjunto. Esta estrategia es atractiva porque requiere de menos pruebas. Generalmente, en sistema complejos y grandes se requerirá bastante tiempo para diagnosticar y arreglar cada defecto. A menos de que se tengan partes de buena calidad, aplicar esta estrategia no es buena opción.

La estrategia de uno a la vez

Consiste en reunir todas las partes disponibles al mismo tiempo; debido a que así se tendrá un sistema más simple con pocos problemas; por eso esta estrategia se considera muy efectiva. Pero tiene la desventaja que requiere mayor trabajo en el desarrollo de las pruebas.

La estrategia Cluster

Consiste en añadir las partes cercanas. Se examinan las partes disponibles y con características particulares, eso consiste en identificar clases con componentes relacionados e integrarlos. Por ejemplo, las partes correspondientes al manejo de archivos, impresiones, o algunas otras. Esta propuesta puede ser útil cuando una función es necesaria para probar el resto del sistema.

La estrategia del sistema por niveles

Consiste en construir el sistema por niveles. Integra todas las partes del nivel mayor y crea capas sucesivas de niveles menores. Se pueden probar todas las partes nuevas una vez o añadir componentes a la vez. La ventaja de esta estrategia consiste en detectar ampliaciones al sistema desde el inicio. También permite construir el esqueleto del sistema. La desventaja es que requiere de bastantes interfaces ficticias para generar valores de regreso a todas las funciones que no estén disponibles.

Desafortunadamente, ninguna estrategia de integración será la mejor. Se aconseja considerar todas las opciones y seleccionar la que se adapte al proyecto en particular.

9.4. La estrategia de pruebas del sistema

En esta estrategia se pretende responder cuatro preguntas:

1. ¿El sistema desempeña las funciones especificadas?.
2. ¿Reúne los objetivos de calidad?.
3. ¿Operará apropiadamente en condiciones normales?.
4. ¿Operará apropiadamente bajo condiciones adversas?.

Estrategia de pruebas del sistema alternas

Consisten en evaluar cada una de las funciones bajo condiciones de estrés; evaluar la usabilidad y el desempeño.

Una de las estrategias se enfoca a áreas funcionales, la cual consiste en cubrir todos los aspectos de cada área antes de pasar a la siguiente. Por ejemplo, probablemente se apliquen pruebas a procesos numéricos en operaciones normales y adversas, a la usabilidad, al desempeño y a la calidad. Esta estrategia evita que las pruebas se dupliquen, pero no trabaja con el comportamiento completo del sistema.

Otra estrategia consiste en aplicar pruebas a las funciones de nivel más bajo en condiciones normales, anormales y de estrés. Después se deberá pasar al nivel superior y probar con funciones agregadas para verificar que trabaja. De nuevo se aplican las pruebas bajo condiciones normales y de estrés. Se deberán aplicar las mismas pruebas a los niveles superiores hasta cumplir con el sistema completo. La estrategia es importante para los sistemas de baja calidad porque existen funciones que no trabajan desde el inicio. La desventaja es que se requiere bastante tiempo para probar todas las combinaciones de funciones en un sistema grande.

Una última estrategia es el camino inverso. Se inicia con las funciones de nivel superior, y posteriormente se pasa a niveles inferiores, aplicando pruebas bajo condiciones normales y de estrés. El sistema se prueba de acuerdo a la funcionalidad, a casos de uso y guiones de pruebas. Aunque cubre el sistema más rápidamente, sólo trabaja con sistemas de alta calidad. La calidad del producto se determina a partir de los datos de los componentes y de las pruebas de integración y de construcción de esos datos.

9.5. Planeación de las pruebas

Se realiza en diversos momentos de acuerdo a TSPi. El plan de pruebas describe las pruebas que se planean aplicar, el orden de las mismas y los materiales necesarios. Con el plan completo se debe mostrar la manera en la que cada requerimiento será probado y cómo cada escenario o guión cubre las áreas de los requerimientos. También se debe saber que áreas serán probadas y cuales tendrán un amplio cubrimiento. Además se deberá nombrar a cada prueba por anticipado, estimar los defectos encontrados en cada fase, el tiempo total en la detección de defectos y el tiempo de pruebas total. Después se deberán estimar los materiales requeridos.

Al final de la planeación de las pruebas se deberá tener:

- Una lista de todos los pasos que se van a desarrollar en las pruebas.
- Los materiales requeridos para cada una.
- Los resultados que generarán.
- Estimación del tiempo de ejecución libre de defectos, de los defectos que serán encontrados y del tiempo total para cada prueba.
- Estimación del trabajo requerido en el plan de pruebas.

También será necesario tener:

- Lista de los materiales requeridos y cuales se utilizarán en cada prueba.
- Los objetivos de cada prueba.
- Las características de los materiales y su tiempo de desarrollo.
- El responsable de desarrollar cada material y la fecha final de terminación.

9.6. Seguimiento y mediciones en las pruebas

En cada ciclo de desarrollo de TSPi se ejecutará un conjunto completo de pruebas, es necesario tener medidas de pruebas. Además de los datos que se registraron en REGD y REGT, se deben responder las siguientes preguntas:

1. ¿Cuánto tiempo se necesitó para ejecutar la prueba?
2. ¿Cuántos defectos se encontraron?
3. ¿Requiere intervención manual o puede procesarse con otras pruebas?
4. ¿Se revisa a si mismo?

En la forma Registro de Pruebas (REGPR) se deberá indicar:

1. La fecha de ejecución de la prueba.
2. El nombre de quien aplica la prueba.
3. Nombre o número de las pruebas.
4. El producto y configuración que se está probando.
5. La hora de inicio.
6. La hora de terminación.
7. El número de defectos encontrados, de acuerdo a los datos en REGD.
8. Los resultados de las pruebas.
9. La configuración del sistema que se está probando.
10. Herramientas o facilidades utilizadas.
11. Si se requirió la intervención de algún operador y cuánto tiempo.

Se deberá anexar la forma Registro de Pruebas en la carpeta.

Módulos de defectos

Utilice datos de pruebas para evaluar el riesgo que el sistema tendría en una o más partes defectuosas. Ordene los datos de defectos por módulo para encontrar cuales tuvieron uno o la

mayoría de los defectos encontrados en cada prueba. Generalmente, aquellos módulos con la mayoría de los defectos mantendrán los mismos después de las pruebas y de manera general no serán encontrados. Se aconseja que en módulos con esas características se suspendan las pruebas temporalmente, se reinspeccione y arreglen tales componentes defectuosos antes de continuar con las pruebas.

Datos de defectos por módulo

La forma Resumen de Defectos Eliminados (RESDE) lista los nombres y/o módulos de nombres en la columna de la izquierda, con los números de defectos eliminados por fase. También se puede generar una forma RESCA en la cual se indican todas las medidas de calidad para módulos particulares. Con esos datos es fácil identificar módulos defectuosos. Al hacer tal revisión se tiene que buscar en cada fase de defectos eliminados de cada producto en el ciclo de desarrollo; y determinar la fase en la cual iniciaron los problemas del módulo.

Seguimiento en los datos de defectos

Es importante que en las reuniones semanales se revisen los defectos encontrados en la construcción, integración o pruebas del sistema con el equipo completo. Sus datos pueden proporcionar señales para encontrar o prevenir defectos similares en un futuro. Por tales motivos se aconseja responder las siguientes preguntas:

1. ¿Qué pasos del proceso originó que los defectos no fueran detectados?
2. ¿Cómo se pueden cambiar tales pasos para que no se vuelvan a repetir?
3. ¿Cómo se puede modificar el proceso de desarrollo para prevenir los defectos en un futuro?
4. ¿Qué parte del sistema es susceptible de mantener defectos que no serían fácilmente encontrados?
5. ¿Cómo se pueden encontrar y arreglar tales defectos?

Posteriormente, el ACP deberá dirigir una revisión de todas las pruebas de construcción e integración, y de los defectos de las pruebas del sistema con el equipo completo. También alguien deberá buscar y arreglar los defectos no encontrados. Se deberá actualizar el proceso de acuerdo a los cambios identificados.

9.7. Documentación

El número de personas asignadas para las pruebas y la documentación varían dependiendo del contenido funcional y de calidad del producto. Es buena idea, que durante el primer ciclo de desarrollo se asignen más ingenieros en las pruebas y en ciclos posteriores, se incremente el número de ingenieros que se asignen a la documentación. El desarrollo de pruebas debería reducirse en ciclos posteriores y la carga de trabajo de la documentación debería aumentar en cada funcionalidad que se incremente.

La importancia de la documentación

La documentación es parte esencial de cada producto de software. Es más importante que el código del programa. TSPi incluye el trabajo de generación de documentación en la fase de pruebas. En grandes sistemas, debería iniciar antes y continuar durante las pruebas. Es conveniente aplicar pruebas a los documentos de los usuarios como parte de las pruebas del sistema.

Diseño de la documentación

Escribir manuales de usuario útiles es un reto para los ingenieros de software.

Un manual bien diseñado debería organizarse de acuerdo a las necesidades del lector y no a la estructura del producto. En general, la primera sección debería contener las necesidades del usuario y no la estructura del producto. La primera sección explica al usuario cómo empezar. Después se deberá explicar al usuario que puede hacer con el producto. A continuación se presentan una guía útil.

- Utilice un glosario para definir términos que no estén en el diccionario estándar.
- Incluya una sección con mensajes de error y de recuperación de errores.
- Tenga un índice de términos importantes.
- Proporcione una tabla de contenido detallado.

El primer paso consiste en generar un esquema detallado. Inicie con un esquema de alto nivel y continúe con un esquema detallado.

Estilo de escritura

Escriba frases cortas, utilice palabras simples y aplique viñetas cuando sea necesario. Por ejemplo:

Pruebas de desarrollo: El Administrador de Desarrollo dirige el desarrollo de pruebas.
Los integrantes del equipo desempeñan las tareas de desarrollo de pruebas.
Asigne las tareas de desarrollo de pruebas al equipo de pruebas.
Defina procesos y procedimientos requeridos.
Desarrolle los procedimientos y facilidades para las pruebas de integración.

Para mayor información en técnicas de escritura consulte bibliografía relacionada [*The Chicago Manual of Style: The Essential Guide for Writers: The University of Chicago Press., Dupre, Strunk, Tichy, Williams, Zinsser*].

Revisión de la documentación

Se aconseja que uno o más colegas revisen la documentación generada. Si el producto es para clientes o usuarios, aplique pruebas de usuario para asegurarse que la escritura es clara, precisa, completa y entendible. A continuación se presentan los siguientes elementos para las revisiones.

- Organización de documentos. ¿El documento está organizado de acuerdo a lo que el usuario hará o de acuerdo al contenido del producto?. Los documentos del usuario deberían estar organizados de acuerdo a sus necesidades y no a la estructura o contenido del producto.
- Terminología del documento. ¿El documento presume conocimiento que el usuario probablemente no tenga?. Debe definir cualquier palabra en el diccionario.
- Contenido del documento. ¿El documento cubre el material requerido?.
- Precisión. ¿El desarrollo de los métodos y procedimientos es de acuerdo a su descripción?.
- Legibilidad. ¿Es fácil leer el documento? Lea el documento verificando que es fácil decir lo que está escrito.
- Entendimiento. ¿Las personas entienden lo que está escrito? Generalmente, esta pregunta es la más difícil de responder. Las mejores pruebas consisten en solicitar a alguien que no ha utilizado el sistema que lo haga. Observe sus reacciones, registre los problemas y modifique el documento de acuerdo a los problemas identificados.

9.8. Los guiones de pruebas de TSPi

Los guiones PRUEBA1 y PRUEBA_n se muestran en las siguientes tablas, pertenecientes al primero y a ciclos posteriores, respectivamente. En PRUEBA_n se hacen pruebas de regresión.

Criterio de entrada

Requiere que se hayan terminado e inspeccionado las especificaciones de ERS y EDS; que los componentes bajo control de configuración se hayan implementado, inspeccionado y se les hayan aplicado pruebas. Para PRUEBA_n, que se tenga una versión anterior bajo control de configuración del producto.

Desarrollo de pruebas

A continuación se mencionan algunos elementos que se generarán durante el desarrollo de pruebas.

1. Pruebas completamente construidas. Consiste en revisar que el sistema ha sido construido y que todos los componentes están incluidos.
2. Los procedimientos de integración para probar todas las interfaces bajo condiciones normales y de error. Consiste en asegurarse de que el sistema construido está listo para iniciar y ejecutarse, que todos los componentes pueden llamar a otros componentes y que todas las interfaces son reconocidas y funcionan apropiadamente.
3. Los materiales para las pruebas de integración. Se aconseja realizar un procedimiento automatizado simple.
4. Los materiales para las pruebas del sistema. Tales materiales deben probar todas las funciones del sistema bajo condiciones normales y de estrés. Deben revisar cómo el sistema será utilizado (en la instalación, conversión, operación o recuperación). Las pruebas del sistema deben considerar la usabilidad, desempeño, anomalías en fechas, y correctez y precisión matemática. Es importante considerar la usabilidad y desempeño en el primer ciclo, porque aplicar modificaciones en esas partes en los ciclos posteriores es difícil.
5. Definición clara de resultados esperados en cada prueba. Aplique revisión a partir de las propias pruebas. Un caso de prueba de ese tipo de pruebas es aquel que genera resultados actuales correctos e incorrectos, eso permitirá hacer una revisión rápida y conocer donde están los problemas.

Construcción

Después del desarrollo de pruebas y de la planeación, este es el siguiente paso en los guiones de pruebas.

1. Se revisan todos los componentes necesarios para asegurarse que están disponibles y reúnen los requerimientos de dependencia. Un requerimiento de dependencia es una capacidad requerida por el sistema base para apoyar al componente. Como ejemplos se mencionan una base de datos, manejo de errores y dispositivos de soporte.
2. Se revisan los elementos proporcionados para la construcción y se identifica cualquier pieza que haga falta. Se asegura que el sistema se pueda construir sin esas piezas y que se tenga control sobre ellas durante la planeación de la integración y el desarrollo de pruebas.
3. Se revisa la consistencia y completas en la construcción. Se asegura que todas las partes requeridas se hayan incluido y que la construcción de los componentes sea un producto portable para las pruebas de integración.
4. Se construye el producto. En productos pequeños consiste en compilar y ligar los componentes del sistema y registrar el tiempo en la forma Registro de Pruebas (REGPR) y los problemas en las formas REGD del propietario.

Cuando se encuentren defectos en la construcción, decida continuar o entregarlos para reparación. Solicite al ACP que le indique que hacer. Si la reparación es rápida, se puede reconstruir; si no, se tendrá que modificar el calendario para la construcción.

Integración

En esta sección:

1. Se revisa que el producto construido esté completo. Se verifica que todas las partes necesarias están presentes en la construcción.
2. Se ejecutan las pruebas de integración sobre lo que se planeó.

3. Para las pruebas, se registra el nombre, fecha, hora, el sistema que se está probando, las pruebas ejecutadas, los tiempos de ejecución y los defectos encontrados en la forma Registro de Pruebas (REGPR).
4. Si se encuentran defectos, el propietario debe registrar cada uno. Si es posible, los defectos se repararán mientras la prueba de integración continua.

Pruebas del sistema

Siga el plan de pruebas para aplicar estas pruebas. También se deberán registrar todos los defectos en la forma Registro de Defectos (REGD). Si es posible los defectos se repararán mientras la prueba del sistema continua.

Pruebas de regresión

El paso final en las pruebas del sistema, consiste en ejecutar las pruebas de regresión a partir del ciclo de desarrollo anterior. Son un subconjunto de las pruebas del sistema de ciclos anteriores. Después de que las pruebas del sistema y de regresión se han ejecutado satisfactoriamente, el sistema está listo para continuar desarrollándose en el ciclo siguiente. En PRUEBA1 no se aplican esas pruebas. Complete todas las formas necesarias (REGD, REGT y REGPR).

Documentación

Complete la documentación del usuario.

Criterio de éxito

En la fase de Pruebas, se debe tener:

- Una versión del producto terminada, integrada, con pruebas del sistema y de regresión.
- La forma Registro de Pruebas (REGPR) completa.
- Registro de defectos completo en la forma REGD y análisis de cada defecto.
- Documentación del usuario terminada y revisada.
- Las formas RESPL y RESCA del sistema y de sus componentes actualizadas.
- Copia de los datos de pruebas y de formas en la carpeta del proyecto.

A continuación se muestran los guiones de esta fase.

Objetivo	Guiar al equipo en la integración y pruebas de los componentes del producto del primer ciclo de desarrollo	
Criterio de entrada	El equipo tiene una estrategia de desarrollo y un plan <ul style="list-style-type: none"> • Los documentos ERS y EDS terminados e inspeccionados. • Los componentes implementados e inspeccionados mediante pruebas unitarias bajo control de configuración. • Los estudiantes han leído este capítulo. 	
General	El ACP determina si las pruebas continúan al encontrar defectos en la construcción, integración o pruebas del sistema. Cada defecto se registra en la forma REGD y es revisado por el equipo para determinar: <ul style="list-style-type: none"> • Dónde es posible encontrar defectos similares. • Cómo y cuándo encontrar y reparar defectos. • Los cambios del proceso para prevenir defectos similares en un futuro. 	
Paso	Actividades	Descripción
1	Panorama en el proceso de pruebas	El instructor describe el proceso de pruebas de integración y del sistema. <ul style="list-style-type: none"> • La necesidad de tener componentes de calidad antes de las pruebas. • La necesidad de tener estándares de pruebas. • Estrategia para manejar componentes de baja calidad.

2	Desarrollo de pruebas	<p>El AD dirige el desarrollo de las pruebas. El LE ayuda en la asignación de tareas para desarrollar las pruebas entre los integrantes.</p> <ul style="list-style-type: none"> • Se definen procesos y procedimientos requeridos en la construcción. • Se desarrollan procedimientos y facilidades de pruebas de integración y del sistema. • Se miden el tamaño y tiempo de ejecución de cada prueba. • Se revisan los materiales de pruebas y corrigen errores.
3	Construcción	<p>El equipo construye el producto y revisa que esté completo</p> <ul style="list-style-type: none"> • Se verifica que las partes necesarias estén disponibles. • Se construye el producto y se pone a disposición para la prueba de integración. • El propietario registra los defectos en REGD.
4	Integración	<p>El AD dirige al equipo en el desarrollo de las pruebas de integración.</p> <ul style="list-style-type: none"> • Se revisa que el producto y las pruebas de integración estén completas. • Se registran todas las actividades de pruebas en la forma REGPR. • El propietario del producto registra los defectos en REGD.
5	Pruebas del sistema	<p>El AD dirige al equipo en el desarrollo de las pruebas del sistema.</p> <ul style="list-style-type: none"> • Se prueba el producto en condiciones normales y de estrés. • Se prueban el producto en cuanto a instalación, conversión y recuperación. • Se registran todas las actividades de pruebas en la forma REGPR. • El propietario del producto registra los defectos en REGD.
6	Documentación	<p>El AD dirige al equipo en:</p> <ul style="list-style-type: none"> • La generación de la documentación del usuario y en las tareas requeridas. • La asignación de esas tareas como parte de la documentación del equipo. • La revisión del esquema con el equipo de pruebas. • La generación del borrador de los documentos del usuario en el primer ciclo. • La revisión, corrección y generación de la documentación del usuario.
Criterio de éxito		<ul style="list-style-type: none"> • El producto del primer ciclo integrado y probado. • Las formas REGD y REGPR de todas las pruebas. • La documentación del usuario revisada y terminada. • Los datos de tiempo, tamaño y defectos registrados.

Tabla 9.1 Guión de Pruebas de Integración y del Sistema para el primer ciclo, PRUEBA1

Objetivo		Guiar al equipo en la integración y pruebas de los componentes del producto para el segundo ciclo de desarrollo o posteriores.
Criterio de entrada		<p>El equipo tiene una estrategia de desarrollo y un plan</p> <ul style="list-style-type: none"> • Los documentos ERS y EDS terminados e inspeccionados. • Los componentes implementados e inspeccionados mediante pruebas unitarias y versiones anteriores del producto bajo control de configuración.
General		<p>El ACP determina si las pruebas continúan al encontrar defectos en la construcción, integración o pruebas del sistema. Cada defecto se registra en la forma REGD y es revisado por el equipo para determinar:</p> <ul style="list-style-type: none"> • Dónde es posible que defectos similares existan. • Cómo y cuándo encontrar y reparar defectos. • Los cambios del proceso para prevenir defectos similares en un futuro.
Paso	Actividades	Descripción
1	Revisión del proceso de pruebas	<p>El instructor describe problemas con el proceso de integración y pruebas del ciclo anterior, para que se corrijan en este ciclo.</p> <ul style="list-style-type: none"> • Las razones para las pruebas de regresión. • Cómo hacer las pruebas de regresión.

2	Desarrollo de pruebas	<p>El AD dirige el desarrollo de las pruebas. El LE ayuda en la asignación de tareas para desarrollar las pruebas entre los integrantes.</p> <ul style="list-style-type: none"> • Se definen procesos y procedimientos requeridos en la construcción. • Se desarrollan procedimientos y facilidades de pruebas de integración y del sistema. • Se mide el tamaño y tiempo de ejecución de cada prueba. • Se revisan los materiales de pruebas y se corrigen errores.
3	Construcción	<p>El equipo construye el producto y revisa que esté completo.</p> <ul style="list-style-type: none"> • Se verifica que las partes necesarias estén disponibles. • Se construye el producto y se pone a disposición para la prueba de integración. • El propietario registra los defectos en REGD.
4	Integración	<p>El AD dirige al equipo en el desarrollo de las pruebas de integración.</p> <ul style="list-style-type: none"> • Se revisa que el producto y las pruebas de integración estén completas. • Se registran todas las actividades de pruebas en la forma REGPR. • El propietario del producto registra los defectos en REGD.
5	Pruebas del sistema	<p>El AD dirige al equipo en el desarrollo de las pruebas del sistema.</p> <ul style="list-style-type: none"> • Se prueba el producto en condiciones normales y de estrés. • Se prueba el producto en cuanto a instalación, conversión y recuperación. • Se aplican pruebas de regresión al sistema. • Se registran todas las actividades de pruebas en la forma REGPR. • El propietario del producto registra los defectos en REGD.
6	Documentación	<p>El AD dirige al equipo en:</p> <ul style="list-style-type: none"> • La actualización de la documentación del usuario y las tareas. • La asignación de esas tareas como parte de la documentación del equipo. • La revisión del esquema con el equipo de pruebas. • La generación del borrador de los documentos del usuario para ciclos posteriores. • La revisión, corrección y generación de la documentación del usuario.
Criterio de éxito		<ul style="list-style-type: none"> • Un sistema, del segundo o de ciclos posteriores, integrado y probado. • Las formas REGD y REGPR de todas las pruebas. • La documentación del usuario revisada y actualizada. • Los datos de tiempo, tamaño y defectos registrados.

Tabla 9.2 Guión de Pruebas de Integración y del Sistema para ciclos posteriores, PRUEBAn

Capítulo 10. El Postmortem

Objetivo: Describir la necesidad del postmortem , su proceso y utilidad.

El Postmortem es el último paso en el proceso de TSPi. Aquí se revisa que el equipo de trabajo haya terminado todas las tareas y registrado todos los datos requeridos. Proporciona una manera estructurada de aprendizaje y mejora, porque se evalúa el desempeño personal y del equipo. Cada ciclo de desarrollo de TSPi finaliza con un postmortem. El presente capítulo proporciona los guiones PM1 y PMn.

10.1. Propósito de Mejora del Proceso

PSP y TSPi utilizan la forma Propuesta de Mejora al Proceso (PMP). En la misma se registran ideas de mejora; las cuales incluyen las mejores prácticas personales sobre el proceso de trabajo en equipo, mejora en las herramientas, cambios al proceso y cualquier otro aspecto que esté relacionado con la mejora del desempeño en el trabajo.

10.2. Los guiones de TSPi

El guión PM1 permite al instructor describir el proceso del postmortem. En PMn se evalúa la efectividad de los cambios del proceso realizados en fases de postmortem anteriores y el equipo decide cuáles puede continuar utilizando o a cuales puede aplicar modificaciones factibles en ciclos posteriores.

En esta fase se genera el reporte de Postmortem, cada integrante describe las actividades del ciclo, las compara con lo que se planeó y con ciclos anteriores. Se evalúan los datos del proyecto y se realiza una revisión y crítica sobre roles.

Criterio de entrada

El equipo ha terminado y evaluado el producto. Los ingenieros han reunido todos los datos y completado todas las formas requeridas. Todos los integrantes han leído este capítulo.

Datos del proceso de revisión

En el inicio del postmortem el ACP dirige al equipo en la revisión de datos del ciclo, incluyendo las formas RESPL y RESCA del producto y sus componentes. Se examinan los datos generados del trabajo en equipo y de cada uno de sus integrantes. Se identifican las áreas en las cuales el proceso se desarrollo favorablemente y dónde surgieron problemas y existe la necesidad de realizar mejoras. Se compara el desempeño del equipo de acuerdo a los objetivos y planes. Se aconsejan mejoras al proceso mediante la forma PMP.

Revisión de calidad

Como parte de la revisión del proceso se compara el desempeño personal y del equipo de acuerdo al plan de calidad. Se inicia con un análisis sobre los datos referentes a los defectos y se evalúan todos los aspectos relacionados en la generación de un producto con calidad. Si el producto tuvo uno o más defectos en las pruebas del sistema, se deberá evaluar en que parte el proceso tuvo deficiencias para realizar mejoras en el futuro. Se puede guiar mediante las siguientes preguntas:

- ¿Cómo comparó el desempeño actual con el planeado?.
- ¿Qué se puede aprender a partir de esta experiencia?.
- ¿Se debería aplicar un criterio personal o por equipo en ciclos posteriores?.
- ¿Dónde se pueden aplicar mejoras y cuáles son las razones de ello?.

Para postmortem posteriores, evalúe las mejoras realizadas y juzgue la efectividad de las mismas. Guíese mediante las siguientes preguntas:

- ¿Dónde el equipo tuvo deficiencias y dónde no?.
- ¿Cómo comparan su desempeño con lo que otros equipos han realizado?.
- ¿Qué metas u objetivos son factibles de aplicarse en ciclos o proyectos posteriores?.
- ¿Dónde se modificarían los procesos utilizados?.

Utilice la forma PMP para las sugerencias de mejora.

Evaluaciones de roles

El Líder de Equipo es el responsable de guiar esta evaluación. Primero se evalúa ese rol y posteriormente los demás. Respecto a las evaluaciones sobre todos los roles deberá considerar que actividad realizó cada uno, dónde surgieron problemas, dónde se pueden realizar mejoras y cuáles de ellas se podrían aprovechar para el siguiente ciclo o proyecto. Se deberá ser lo más específico posible.

Prepare el reporte del ciclo

En este reporte se describe lo que se produjo, el proceso utilizado y los roles desempeñados; lo que sí se pudo aplicar y lo que no fue conveniente, para ello se pueden hacer sugerencias que puedan aplicarse la siguiente vez. Se deberá describir el desempeño que cada integrante tuvo respecto al rol que desempeñó, de acuerdo a los indicados por TSPi, así como el rol correspondiente al área de desarrollo. Si es posible, se justificarán las conclusiones con datos actuales del equipo y se comparará el desempeño en ciclos anteriores. Tal reporte debe ser breve y objetivo.

El reporte del ciclo deberá tener el siguiente contenido:

- Índice.
- Resumen.
- Reporte de roles, en cuanto a:
 - Liderazgo.
 - Desarrollo.
 - Planeación.
 - Proceso.
 - Calidad.
 - Apoyo o soporte.
- Reporte de ingenieros.

Reporte de roles

El Líder de Equipo será responsable de generar el índice y el resumen, en ese último se describirán fortalezas y debilidades de cada área brevemente.

En el reporte de roles se deberán añadir dos secciones, una en la que se describirá la evaluación sobre el rol desempeñado individualmente y la evaluación sobre el desempeño del equipo de acuerdo a la perspectiva de TSPi sobre cada rol.

El objetivo de este reporte consiste en proporcionar una guía para desempeñar de mejor manera cada rol o que sea utilizado por quien posteriormente tenga que desempeñar cada rol tomando en cuenta las recomendaciones escritas.

El Líder de Equipo deberá analizar el desempeño del equipo desde la perspectiva de liderazgo, tomando en cuenta asuntos de motivación y acuerdos establecidos; áreas donde la guía del instructor pudo haber sido útil. También se deberá hacer referencia a las reuniones de equipo, tomando en cuenta la manera en la que establecieron tales reuniones y se deberá mencionar como poder mejorar esa responsabilidad en el futuro.

El Administrador de Desarrollo deberá comparar el contenido del producto con los requerimientos y evaluar la efectividad de la estrategia de desarrollo; para ello deberá analizar si la estrategia generó los resultados esperados y se deberá identificar que cambios podrían aplicarse. Respecto al diseño y a la implementación, se deberá analizar el reuso aplicado, el desempeño, la compatibilidad, instalación, mantenimiento, seguridad, etc.; para que los resultados originados puedan aplicarse en ciclos de desarrollo o proyectos futuros.

El Administrador de Planeación compara el desempeño del equipo real con lo que se planeó, tomando en cuenta las horas semanales y los valores ganados, y detecta cómo realizar un mejor trabajo en ciclos futuros. También es conveniente que analice el desempeño del equipo en ciclos anteriores o en otros proyectos.

El Administrador de Calidad y Proceso compara los datos actuales de calidad con los objetivos de calidad establecidos para describir el desempeño del equipo. Muestra las tendencias de calidad de los ciclos de desarrollo terminados hasta la fecha actual. En la sección del proceso deberá evaluar la disciplina del mismo, tomando en cuenta hasta que grado los integrantes siguieron el proceso, midiendo el trabajo, utilizando medidas de mejora, y generando formas PMP. Deberá incluir un resumen de todas las formas PMP generadas y un resumen en el que se deberá indicar cómo fueron manejadas. También deberá revisar el seguimiento en los estándares establecidos y en las revisiones aplicadas. Tal análisis deberá incluir una sección de posibles sugerencias y mejoras para el futuro.

El Administrador de Apoyo describe las facilidades de soporte, problemas enfrentados y mejoras en cualquier área. Hace comentarios sobre el procedimiento de control de cambios y la administración de configuración, tomando en cuenta aspectos que intervinieron en los cambios, los impactos de los mismos, y hace recomendaciones para un mejor manejo en ciclos o proyectos futuros. También se refiere al manejo y seguimiento en riesgos y otros asuntos. Respecto al reuso indica el porcentaje realizado por el equipo y cada ingeniero e indica mejoras para los mismos en un futuro.

Reportes de ingenieros

Cada ingeniero debe describir el desempeño personal de acuerdo a lo planeado y la calidad del trabajo que realizó. Es importante indicar cómo mejorar el trabajo para la siguiente vez.

Cada ingeniero deberá entregar sus comentarios al LE. El ACP deberá revisar el reporte completo y sugerir las correcciones necesarias. Cuando el reporte final del ciclo esté correcto deberá entregarse al instructor y a los integrantes del equipo.

Evaluaciones de roles

Consiste en llenar la forma Evaluación del Equipo por Colegas, la cual se encuentra en el Anexo C (forma C.11). En tal forma se evalúa el desempeño del equipo y de cada rol. En la primera sección se evalúa el trabajo requerido y la dificultad de los roles en cada una de las actividades respectivas. Se asigna a cada rubro un porcentaje correspondiente, donde la suma total debe ser igual a cien por ciento. Para los siguientes rubros, en cada aspecto se selecciona el número correspondiente, el rango inicia en 1 y termina en 5, catalogado desde inadecuado hasta superior, conforme avanza la numeración.

El objetivo de tal evaluación consiste en identificar y aplicar mejoras que permitan manejar de mejor manera los roles.

Criterio de éxito

El equipo generó un producto de alta calidad y la documentación necesaria.

- El producto terminado está bajo control de configuración.
- Los datos del proceso han sido evaluados y las formas PMP se completaron y entregaron.
- Las evaluaciones sobre los roles se realizaron y entregaron.

- El reporte del ciclo de desarrollo se terminó, se proporcionó una copia a cada integrante y al instructor.
- Todas las formas de TSPi han sido completadas, tanto para el sistema y sus partes componentes.
- La carpeta del proyecto está actualizada con los datos y el reporte del ciclo correspondiente.

A continuación se presentan los guiones correspondientes.

Propósito		<ul style="list-style-type: none"> • Reunir, analizar y registrar los datos del proyecto. • Evaluar el desempeño del equipo y de cada rol. • Identificar mejoras para el proceso en el segundo ciclo. • Producir el reporte del primer ciclo.
Criterio de entrada		<ul style="list-style-type: none"> • Los ingenieros han terminado y aplicado pruebas al producto. • Han completado todas las formas. • Los estudiantes han leído los capítulos 10 (Postmortem), 16 (Administrándose uno mismo), 17 (Iniciando un equipo) y 18 (Equipo de Trabajo) (estos tres últimos capítulos los puede consultar en la bibliografía correspondiente a TSPi).
General		<p>El reporte del primer ciclo contiene un análisis del proyecto por cada rol.</p> <ul style="list-style-type: none"> • Desempeño completo del equipo, realizado por el Líder de Equipo. • Desempeño planeado contra real, realizado por el Administrador de Planeación. • Diseño y estándares del producto, realizado por el Administrador de Desarrollo. • Administración de cambios y de soporte del producto, realizado por el Administrador de Apoyo. • Calidad del proceso y del producto, realizado por el Administrador de Calidad y Proceso. <p>El reporte del ciclo debería:</p> <ul style="list-style-type: none"> • Utilizar los datos del proceso para apoyar los comentarios de los ingenieros. • Considerar los resultados obtenidos. • Ser corto y conciso.
Paso	Actividades	Descripción
1	Panorama sobre el proceso postmortem	<p>El instructor describe el proceso postmortem</p> <ul style="list-style-type: none"> • La necesidad de tener datos precisos del proceso. • El contenido del reporte del ciclo. • El proceso de evaluación del equipo por colegas y las formas EEC.
2	Revise los datos del proceso	<p>El ACP dirige al equipo en el análisis de los datos del proyecto, en la identificación de problemas y áreas para mejorar, de acuerdo a:</p> <ul style="list-style-type: none"> • Liderazgo, planeación, proceso, calidad y soporte. • Nuevas toma de acciones y responsabilidades en equipo. • Mejoras por parte del instructor. <p>Los ingenieros preparan y entregan las formas PMP con las sugerencias indicadas.</p>
3	Evalúe el desempeño del rol	<p>El LE dirige la evaluación sobre la efectividad de los roles, las actividades del instructor y las facilidades de soporte.</p> <ul style="list-style-type: none"> • Dónde se generó mayor beneficio. • Dónde se pueden hacer mejoras.
4	Prepare el reporte del primer ciclo	<p>El LE dirige al equipo en la generación del reporte del primer ciclo.</p> <ul style="list-style-type: none"> • Se asignan actividades a cada integrante para generar tal reporte. • Se establecen acuerdos para entregar el documento asignado. • Se integra, revisa y corrige el reporte terminado.
5	Prepare las evaluaciones de roles	<p>Cada ingeniero entrega la forma EEC completada</p> <ul style="list-style-type: none"> • En cuanto a dificultad y contribución de cada rol. • Con porcentajes que deberían sumar cien por ciento. • La efectividad de cada rol, calificada desde 1 (inadecuado) hasta 5 (superior).

Criterio de éxito	<ul style="list-style-type: none"> • El ciclo de desarrollo ha generado un producto de alta calidad y la documentación necesaria. • El producto terminado está bajo control de configuración. • Los datos del proceso han sido evaluados y las formas PMP entregadas. • Las evaluaciones del equipo entre colegas se realizaron y entregaron (EEC). • El reporte del primer ciclo se terminó y entregó. • Las formas RESPL y RESCA han sido completadas, tanto para el sistema y sus partes componentes. • La carpeta del proyecto ha sido actualizada.
-------------------	--

Tabla 10.1 Guión de Postmortem para el primer ciclo, PM1

Propósito	<ul style="list-style-type: none"> • Reunir, analizar y registrar los datos del último ciclo. • Evaluar el desempeño del equipo y de cada rol. • Identificar mejoras para el proceso en ciclos posteriores. • Producir el reporte del ciclo. 	
Criterio de entrada	<ul style="list-style-type: none"> • Los ingenieros han terminado y aplicado pruebas al producto. • Han completado todas las formas. 	
General	<p>El reporte del ciclo contiene un análisis del proyecto por cada rol.</p> <ul style="list-style-type: none"> • Desempeño completo del equipo, realizado por el Líder de Equipo. • Desempeño planeado contra real, realizado por el Administrador de Planeación. • Diseño y estándares del producto, realizado por el Administrador de Desarrollo. • Administración de cambios y de soporte en el proyecto, realizado por el Administrador de Apoyo. • Calidad del proceso y del producto, realizado por el Administrador de Calidad y Proceso. <p>El reporte del ciclo debería:</p> <ul style="list-style-type: none"> • Utilizar los datos del proceso para apoyar los comentarios de los ingenieros. • Considerar los resultados obtenidos. • Ser corto y conciso. 	
Paso	Actividades	Descripción
1	Revisión del proceso postmortem	<ul style="list-style-type: none"> • El instructor describe problemas del proceso postmortem del ciclo anterior, que deben ser corregidos en este ciclo.
2	Revise los datos del proceso	<p>El ACP dirige al equipo a:</p> <ul style="list-style-type: none"> • Analizar los datos del proyecto e identificar áreas con problemas. • Evaluar las propuestas de mejora establecidas en las formas PMP de ciclos anteriores. <p>Se identifica dónde es necesario realizar mejoras, de acuerdo a:</p> <ul style="list-style-type: none"> • Liderazgo, planeación, proceso, calidad o soporte. • Toma de acciones y responsabilidades en equipo. • Mejoras por parte del instructor. <p>Se preparan y entregan las formas PMP con las sugerencias indicadas.</p>
3	Evalúe el desempeño del rol	<p>El LE dirige la evaluación sobre la efectividad de los roles, las actividades del instructor y las facilidades de soporte.</p> <ul style="list-style-type: none"> • Dónde generaron mayor beneficio. • Dónde se pueden hacer mejoras.
4	Prepare el reporte del ciclo n	<p>El LE dirige al equipo en la generación del reporte del ciclo n.</p> <ul style="list-style-type: none"> • Se asignan actividades a cada integrante para generar tal reporte. • Se establecen acuerdos para entregar el documento asignado. • Se integra, revisa y corrige el reporte terminado.
5	Prepare las Evaluaciones de Roles	<p>Cada ingeniero entrega la forma EEC completada</p> <ul style="list-style-type: none"> • En cuanto a dificultad y contribución de cada rol. • Con porcentajes que deberían sumar cien por ciento. • La efectividad de cada rol, calificada desde 1 (inadecuado) hasta 5 (superior).

Criterio de éxito	<ul style="list-style-type: none">• El ciclo de desarrollo ha generado un producto de alta calidad y la documentación necesaria.• El producto terminado está bajo control de configuración.• Los datos del proceso han sido evaluados y las formas PMP entregadas.• Las evaluaciones del equipo entre colegas se realizaron y entregaron (EEC).• El reporte del ciclo n se terminó y entregó.• Las formas RESPL y RESCA han sido completadas, tanto para el sistema y sus partes componentes.• La carpeta del proyecto ha sido actualizada.
-------------------	---

Tabla 10.2 Guión de Postmortem para ciclos posteriores, PMn

Capítulo 11. Roles de Equipo de TSPI

Objetivo: Describir los objetivos, destrezas y habilidades y las actividades principales de cada rol.

Con TSPI, todos los integrantes del equipo tienen objetivos específicos y cuantificables definidos por el proceso. Si se desea modificarlos, se deberá hacer lo siguiente: redactar los objetivos que se desean usar, especificar cómo se cuantificarán esos nuevos objetivos, describir porque se han seleccionado en lugar de los aconsejados por TSPI, dar una copia de los objetivos revisados al equipo y al Instructor, y poner una copia en la carpeta del proyecto.

Con TSPI, cada integrante del equipo tiene dos tipos de objetivos: ser un integrante de equipo eficaz y tener sus objetivos propios.

Objetivo común

Este consiste en ser un integrante de equipo eficaz y cooperativo. Todos tienen el objetivo principal de trabajar en cooperación con el equipo entero para generar un producto de alta calidad de acuerdo al calendario establecido para el proyecto. El éxito del equipo depende de que todos sus integrantes contribuyan con su mejor esfuerzo personal apoyando a los demás y trabajando conjuntamente para resolver problemas y desacuerdos.

Las medidas sugeridas son las siguientes:

- Las evaluaciones de la forma Evaluación del Equipo por Colegas (EEC) del equipo completo, son buenas o bastante buenas, si el resultado obtenido es mayor o igual a 3 al referirse al rubro de espíritu de equipo.
- Las evaluaciones de la forma Evaluación del Equipo por Colegas (EEC) para cada rol, son buenas o bastante buenas, si el resultado obtenido es mayor o igual a 3 al referirse al rubro de apoyo y ayuda.

A continuación se mencionan los roles de TSPI y los puntos a tratar en cada uno. También se presentan los guiones correspondientes a cada rol.

11.1. Rol: Líder de Equipo

11.1.1. Objetivos

Objetivo 1: Construir y mantener un equipo eficaz. Las medidas de éxito para este objetivo son las siguientes:

- El equipo conoce el costo estimado, el calendario establecido y los objetivos de calidad
- Las evaluaciones de la forma Evaluación del Equipo por Colegas (EEC) en el rubro efectividad total del equipo es igual o mayor a 3.
- Las estimaciones de los integrantes respecto a la contribución total de este rol, son buenas o bastante buenas si el resultado es mayor o igual a 3.
- Las estimaciones de los integrantes respecto al desempeño del rol, son buenas o bastante buenas si el resultado es mayor o igual a 3.

Objetivo 2: Motivar a todos los integrantes del equipo a comprometerse en el trabajo del proyecto. El Líder de Equipo tiene que establecer tiempos para realizar el trabajo requerido que permita generar productos de calidad. Las medidas para mantener control se indican a continuación:

- Todos los integrantes del equipo trabajan las horas que se comprometieron.
- Todos los integrantes del equipo conocen sus compromisos de valor ganado.
- Todos los integrantes del equipo siguen el proceso de TSPi completando sus formas REGT y REGD, registran el estado semanal en la forma SEMANA, entregan su reporte a tiempo y reportan los resultados del producto en las formas RESPL y RESCA.

Objetivo 3: Resolver todos los conflictos entre los integrantes del equipo. El LE tiene que identificar y resolver rápida y objetivamente los desacuerdos y problemas entre los integrantes. Las medidas sugeridas son :

- Las evaluaciones de los integrantes del equipo respecto a este rol, en las formas EEC, son buenas o bastante buenas, si son mayor o igual a 3 en el rubro de ayuda y apoyo.

Objetivo 4: Mantener al instructor completamente informado sobre el avance del equipo. Una parte importante durante el desarrollo de un proyecto es mantener al administrador informado del estado, avance y problemas en el equipo. Para completar este objetivo se debe:

- Proveer al instructor reportes de estado del proyecto completos y exactos semanalmente.
- Asegurar que el instructor esté oportunamente informado de cualquier problema que requiera de su atención.

Objetivo 5: Ser mediador eficaz en las reuniones del equipo. La medida más eficaz como líder de equipo consiste en analizar la manera en la que cada integrante participa activamente y contribuye en las reuniones de equipo y en otras actividades, porque así pueden sentir una satisfacción propia en el trabajo de equipo y en los resultados obtenidos. Por lo tanto, la medida sugerida para este objetivo consiste en evaluar el ciclo de desarrollo como una experiencia gratificante (mayor o igual a 3) en las formas de evaluación EEC.

11.1.2. Destrezas y habilidades

Las características más importante que un líder de equipo debe tener son las siguientes:

1. Disfruta actuar como líder.
2. Tiene la capacidad de identificar los conflictos clave y tomar decisiones objetivas. Por tanto, el primer paso crucial en la elaboración de decisiones efectivas es identificar claramente los problemas o desacuerdos que se deberán resolver. Después de que se han encontrado y analizado los problemas objetivamente, las respuestas son obvias. Si el líder se propone que el trabajo salga bien, escuchará las opiniones de sus compañeros de equipo.

El LE tiene seguidores

A continuación se mencionan los prerequisites para que un líder tenga seguidores: tiene visión de lo que quiere que se haga, el equipo está motivado para llevar a cabo la visión del líder y el líder está comprometido a ayudar a su equipo en las actividades que vislumbro como guía para realizar el trabajo.

El objetivo del líder es ayudar a que los integrantes del equipo se unan para cumplir los objetivos del equipo y procurar que se realicen. Para esto, el líder debe conocer la capacidad de todos los integrantes y aplicarlas donde y cuando las tareas lo requieran. Para eso debe escuchar las ideas de su colaboradores aunque difieran de su punto de vista.

El líder solicita a su equipo que tengan gran disposición para el desarrollo del trabajo

El trabajo de este rol consiste en que el equipo haga su trabajo adecuadamente. Debe propiciar que el trabajo se haga con calidad cumpliendo con los estándares establecidos.

El líder toma posiciones difíciles

El líder tendrá que manejar disputas ocasionales. En ocasiones, existirán integrantes del equipo con visiones fuertemente opuestas, y a menudo, no siempre serán la mejor respuesta. Por tanto, el líder deberá analizar tales visiones y tomar una decisión sobre las mismas, para aplicarlas o no.

El trabajo del líder es motivar al equipo para que tenga grandes deseos de trabajar y de que exista paz en el equipo.

El líder maneja problemas de personal

Puede ser que un integrante del equipo no tenga el deseo de cooperar en el trabajo, porque no se encuentra en acuerdo con la estrategia del proyecto o porque no se siente motivado para desempeñar su trabajo. La labor del LE es convencer a los integrantes a que trabajen en equipo, sin permitir que los conflictos avancen. Debe analizar el problema y resolverlo, sino deberá acudir con el instructor para que lo apoye en tal problema.

11.1.3. Actividades principales del Líder de Equipo

Actividad 1: Motivar a los integrantes del equipo en el desarrollo de sus tareas. La principal actividad es motivar a los integrantes del equipo a desarrollar de la mejor manera sus habilidades. TSPi fomenta la motivación involucrando a todos los integrantes del equipo en la planeación y administración de sus tareas personales.

A continuación se presentan los conflictos más comunes en un trabajo en equipo: algunos integrantes del equipo no siguen el proceso, algunos integrantes del equipo no realizan el esfuerzo suficiente y para algunos integrantes, es difícil respetar el calendario del proyecto.

No se sigue el proceso

Debe motivarse a los ingenieros para que sigan el proceso o el proyecto fallará. Si algunos de ellos se rehusan a cooperar, a reunir datos o a no reportar los progresos, esa manera de actuar no cambiará sin una fuerte acción correctiva. El problema deberá ser tratado en un inicio por el LE, si él no lo resuelve se canalizará al instructor.

No se realiza el esfuerzo necesario

Este es un problema común entre los estudiantes. Algunos estudiantes tienen cargas de trabajo pesadas y otros tienen empleos u obligaciones familiares. La mayor dificultad es causada por los estudiantes que no se presentan y que pierden interés en el compromiso establecido. La gente que tiene estos problemas nunca ha tenido una disciplina de compromiso, por tanto, la tarea del líder es ayudarlos a desarrollar esta disciplina lo antes posible. El primer paso consiste en comunicar al estudiante que su comportamiento es inaceptable e informar al instructor del problema, explicándole que está trabajando en la solución del mismo. Posteriormente, se establecerá un compromiso con el estudiante. Si este compromiso falla, se deberá acudir al instructor para que el estudiante sea eliminado del equipo.

No se respeta el calendario establecido

Si el equipo está trabajando de acuerdo al tiempo establecido, pero el trabajo no es óptimo, frecuentemente este problema se debe a que se trata de nuevos proyectos. Probablemente, sea el primer proyecto para la mayoría de los integrantes del equipo, y puede ser la primera vez que están aplicando TSPi. Quizás, también se trata de un nuevo equipo que está aprendiendo a trabajar unido. Todo aprendizaje toma tiempo y no hay que desesperarse rápidamente. Un poco de tiempo y guía del instructor ayudarán a superar los problemas.

La mejor manera de respetar el calendario establecido consiste en fijar metas en las reuniones semanales para la siguiente semana.

Actividad 2: Efectuar la reunión semanal de equipo. La segunda actividad de este rol consiste en realizar la reunión de equipo semanal, utilizando el guión SEMANA.

Los principales objetivos de la reunión consisten en comparar el avance del proyecto con las tareas de la semana anterior; asegurarse de que todos están indicando apropiadamente el tiempo requerido, el tamaño y los datos de defectos y que las formas se están completando correctamente. Otro objetivo consiste en que se establezcan objetivos por cada ingeniero y por el equipo completo para la semana siguiente. Esos deben incluir horas semanales, valor ganado, y las tareas específicas que serán realizadas y completadas. Otro objetivo consiste en discutir previamente los riesgos identificados y decidir quién tomará acciones correctivas sobre los mismos, identificar nuevos riesgos y decidir quién mantendrá seguimiento sobre ellos y los manejará adecuadamente. El último objetivo consiste en identificar cualquier problema nuevo y determinar la solución necesaria.

Actividad 3: Reportar el estado semanal. Como Líder de Equipo, la tercera actividad principal consiste en reportar el estado del proyecto al instructor. La información necesaria para este reporte puede recopilarse de las reuniones semanales. Procure realizar las reuniones semanales del equipo antes de la primera clase o laboratorio semanal. Si los integrantes de equipo entregan sus datos al Administrador de Planeación al final de la semana anterior, se tendrá el tiempo suficiente para resumir esos datos antes de la reunión semanal.

El reporte semanal del equipo tiene como propósitos mantener al instructor informado del estado y avance, para que ayude al LE cuando lo necesite y motivar al equipo para que tenga energía y entusiasmo al informarle lo que han realizado y lo que deben hacer para continuar con el plan.

Actividad 4: Ayudar al equipo a asignar tareas. La cuarta actividad del LE consiste en asignar tareas a los integrantes. TSPi presentan los siguientes consejos: hacer una lista de las tareas que faltan asignar; hacer otra lista de los integrantes de equipo que están disponibles; pedir a cada uno que escriba en un pedazo de papel dos o tres tareas que desee realizar; dibujar una tabla en un pizarrón, escribiendo las tareas en los renglones y los nombres de los integrantes en las columnas; marcar en la tabla las tareas seleccionadas por un solo ingeniero; llenar la lista de tareas hasta que todas las asignaciones queden cubiertas y no asignar más tareas a un ingeniero cuando haya acumulado un número considerable.

Actividad 5: Actuar como mediador y administrador de tiempo en las reuniones de equipo. Esta actividad involucra comparar y registrar el avance de las reuniones de acuerdo a la agenda indicada. Esto requiere que en las reuniones de equipo se tenga una agenda con tiempos establecidos. En la primera reunión semanal, el LE debe hacer una agenda para comparar los tiempos actuales contra las actividades indicadas en la agenda. Después, tendrá datos actuales que servirán como guía en la asignación de tiempos para las reuniones posteriores. Se utilizará el guión SEMANA para realizar la agenda semanal.

Como mediador, es responsabilidad del LE colaborar para que las reuniones sean eficientes. Sus responsabilidades son:

- Enfocar la discusión en los temas de la agenda. Cuando un punto esté fuera de la agenda, el líder deberá anotarlo en una libreta o pizarrón para que sea tratado posteriormente.
- Enfocar la reunión a una simple discusión. Como mediador, discretamente se pedirá al grupo que se concentre en un tema. Si otros temas son significativos y no están en la agenda se escribirán en una lista para atenderlos más adelante.
- Evitar que alguien más dirija la discusión. Si alguien toma el mando de la discusión, el LE debe pedirle en su momento que indique sus puntos de vista y se debe asegurar que todos tengan la oportunidad de hablar.

- Asegurar que todos estén escuchando y participando. El LE deberá pedir los puntos de vista de los demás en el momento oportuno.
- Cerrar la discusión. Cuando se cierre la discusión de un tema, el LE se debe asegurar que el secretario (el Administrador de Calidad y Proceso) registre las conclusiones finales. Se debe anotar la decisión final y la razón. Si la acción se realizó se debe registrar quién y cuando la llevó a cabo. Si se decidió aplazarla o rechazarla se debe anotar la razón. También se deberá indicar quién es responsable de proporcionar la información y quién establecerá fecha para la reunión.

Actividad 6: Integrar y completar la carpeta del proyecto. La sexta tarea consiste en actualizar la carpeta del proyecto. Ésta es un registro oficial de todas las actividades del equipo. En ella se incluyen copias de todos los productos de trabajo, los reportes de prueba, los planes, los reportes de inspección, los reportes de las reuniones, los reportes de los estados de equipo semanales, el reporte final para cada ciclo de desarrollo y el conjunto completo de las formas del proyecto completadas.

Actividad 7: Dirigir al equipo en la generación del reporte del ciclo de desarrollo. La tarea final es conducir la preparación del reporte del ciclo. Se deberá guiar a los integrantes del equipo en la generación, revisión y corrección de las partes. Como parte de este reporte, se deberá proveer el esquema y el resumen de apertura del reporte en el que se indique el desempeño completo del equipo. Finalmente, se describirán las áreas en las que tuvo problemas y en las que todo resultó bien, en cuanto a este rol. También se deberá comentar que mejoras se sugieren para el mismo.

Actividad 8: Actuar como un ingeniero de desarrollo. Como previamente se ha indicado, cada integrante del equipo también desempeña el rol de ingeniero de desarrollo.

A continuación se presenta el guión correspondiente a este rol.

Objetivo	El Líder de Equipo dirige el equipo y se asegura que los ingenieros reporten los datos del proceso y completen su trabajo de la manera planeada.
Características del rol	Las características más útiles para los líderes de equipos son las siguientes: <ol style="list-style-type: none"> 1. Disfruta siendo líder y asume naturalmente el rol de liderazgo. 2. Es capaz de identificar los conflictos importantes y tomar decisiones objetivas. 3. Está dispuesto a presionar a la gente para completar tareas difíciles. 4. Respeta a cada uno de los integrantes del equipo y está dispuesto a escuchar sus puntos de vista y ayudarlos a desarrollar sus habilidades.
Objetivos y Medidas	<p>Objetivo del integrante de equipo: Ser un integrante de equipo eficaz y cooperativo.</p> <ul style="list-style-type: none"> • Medidas: Las estimaciones de la forma EEC respecto al espíritu del equipo, a la contribución total de roles y al apoyo y ayuda. <p>Objetivo 1: Construir y mantener un equipo eficaz.</p> <ul style="list-style-type: none"> • Medida 1.1. Desempeño del proyecto contra el costo, calendario y objetivos de calidad. • Medida 1.2. Evaluaciones EEC sobre la efectividad completa del equipo. • Medida 1.3. Evaluaciones EEC sobre la contribución completa del Líder de Equipo. • Medida 1.4. Estimaciones de los integrantes del equipo para evaluar el desempeño del rol. <p>Objetivo 2: Motivar a todos los integrantes del equipo a comprometerse en el trabajo del proyecto.</p> <ul style="list-style-type: none"> • Medida 2.1. Todos los integrantes del equipo trabajaron las horas acordadas. • Medida 2.2. Todos los integrantes del equipo conocieron el valor ganado de sus acuerdos. • Medida 2.3. Todos los integrantes del equipo siguieron el proceso TSPi, registrando todos los datos y completando todas las formas requeridas. <p>Objetivo 3: Resolver todos los conflictos entre los integrantes del equipo.</p> <ul style="list-style-type: none"> • Medida 3. Las valoraciones EEC sobre el rol del Líder de Equipo de los integrantes de equipo respecto al apoyo y ayuda. <p>Objetivo 4: Mantener al instructor completamente informado sobre el avance del equipo.</p> <ul style="list-style-type: none"> • Medida 4.1. Entregar al instructor reportes del estado del proyecto completos y exactos semanalmente.

	<ul style="list-style-type: none"> • Medida 4.2. Se asegura que el instructor esté oportunamente informado de cualquier problema de equipo que pueda requerir de su atención. <p>Objetivo 5: Ser mediador eficaz en las reuniones del equipo.</p> <ul style="list-style-type: none"> • Medida 5. La evaluación EEC como una experiencia gratificante.
Actividades Principales	<ol style="list-style-type: none"> 1. Motivar a los integrantes a desarrollar sus tareas. 2. Cada semana, antes de la primera clase, al inicio de la misma o en la sesión de laboratorio, se deberá efectuar la reunión semanal de equipo para que: <ul style="list-style-type: none"> • Se mantenga un seguimiento de todas las tareas para verificar que han sido cumplidas. • Se verifique que todos los integrantes de equipo han entregado los datos requeridos. • Se verifique que todas las formas necesarias se han completado en el tiempo establecido. • Se verifique el estado del proyecto en cuanto a riesgos y otros asuntos. • Se identifiquen las tareas que se realizarán en la siguiente semana y quiénes serán los responsables. 3. Reportar semanalmente el estado del equipo al instructor. <ul style="list-style-type: none"> • Se muestra la carpeta del proyecto con los datos del equipo. • Se busca asesoramiento del instructor para los ingenieros que fallan en terminar sus tareas o en entregar datos a tiempo. • Se obtiene asesoramiento del instructor para todo el equipo. 4. Ayudar al equipo en la asignación de tareas y en la resolución de otros asuntos. 5. Actuar como mediador y estar pendiente del tiempo utilizado en las reuniones del equipo. 6. Integrar y completar la carpeta del proyecto. 7. Dirigir al equipo en la generación del reporte del ciclo de desarrollo. 8. Actuar como un ingeniero de desarrollo.

Tabla 11.1 Guión del rol Líder de Equipo

11.2. Rol: Administrador de Planeación

11.2.1. Objetivos

Objetivo 1: Generar un plan completo, preciso y detallado para el equipo y cada integrante. Las medidas son:

- El plan está completo cuando cubre todas las tareas que el equipo desempeñará en el ciclo de desarrollo y está documentado en las formas TAREA y CALENDARIO.
- Cada integrante genera las formas TAREA y CALENDARIO para su trabajo.
- El plan del equipo es preciso cuando el número promedio de horas por tarea es igual o menor a cinco. Para tareas en las que participa más de un integrante se tomarán como aceptable que el número de horas sea mayor o igual a cinco y menor o igual a diez.
- El plan preciso se mide de acuerdo al error en las horas totales estimadas en el ciclo comparándolas con el esfuerzo actual requerido.
- Una segunda medida precisa son las horas semanales planeadas del equipo contra las horas semanales actuales.

Objetivo 2: Proporcionar los reportes del estado semanal del equipo. El Administrador de Planeación es responsable de apoyar al equipo en el desarrollo y seguimiento del proyecto. Las medidas consisten en reportar el estado semanal del equipo de manera precisa, proporcionando datos individuales y del equipo de acuerdo a las horas trabajadas y al valor ganado; que cada integrante del equipo actualice las formas TAREA, CALENDARIO y SEMANA y las entregue en

una fecha razonable y que si alguno de los integrantes no reportan a tiempo sus datos, se busque ayuda del LE y del instructor.

11.2.2. Destrezas y habilidades

1. Desarrolla un plan de trabajo bien definido.
2. Tiene interés en procesar datos y en conocer el estado del proyecto registrando resultados semanales.
3. Piensa que la planeación es importante y esta preparado para presionar a los integrantes del equipo en el seguimiento y medición de su trabajo.

11.2.3. Actividades principales del Administrador de Planeación

Actividad 1: Conducir al equipo en la generación del plan para el siguiente ciclo de desarrollo. Antes que el equipo haya producido los requerimientos, TSPi requiere que se haya hecho un plan de desarrollo (actividad más importante de este rol). La razón anterior surge por la necesidad de administrar el trabajo de acuerdo al tiempo disponible. Los pasos esenciales son los siguientes: especificar los productos que se generarán; estimar los tamaños de esos productos; para cada producto, estimar el tiempo requerido en cada paso del proceso y registrar todos los datos en las formas RESTM y TAREA.

Posteriormente, con datos históricos se generarán mejores planes.

Actividad 2: Dirigir al equipo en la generación del calendario, estimando tiempos para el siguiente ciclo de desarrollo. Después que se tiene una lista con horas para cada tarea y tamaños estimados para cada producto, el siguiente paso consiste en asignar fechas. Debe pedirse a cada ingeniero que especifique el número de horas que utilizará en el proyecto semanalmente para el ciclo de desarrollo actual. Debe indicarse a cada integrante que deberán estimar el tiempo directo que utilizarán en las tareas listadas en los guiones de TSPi y en la forma CALENDARIO.

Después que se tienen las horas planeadas para cada integrante del equipo, se generará el plan de equipo en conjunto. Esto proporcionará una base para determinar las etapas del plan. También, es una base para generar los planes individuales de los ingenieros. En este sentido, este es un diseño de alto nivel del plan y provee el contexto para todos los planes de los ingenieros.

Actividad 3. Dirigir al equipo en la generación de un plan balanceado. La generación de un plan balanceado es uno de los pasos más importantes en el proceso de planeación. La razón es que cuando la carga del equipo no está balanceada, algunos ingenieros tienen más trabajo que otros y les toma más tiempo terminar sus tareas y retrasan el proyecto entero. A menos que cada ingeniero genere un plan detallado y el equipo balancee la carga de trabajo total, los proyectos indudablemente tomarán más tiempo del necesario. Esto se puede resolver mediante el balanceo de la carga de trabajo.

Para balancear la carga de trabajo elabore una tabla con los nombres de los ingenieros a la izquierda (en cada renglón) y las semanas correspondientes para el ciclo en curso (en columnas). Entonces, tomé la primera tarea que se elaborará e indique las horas requeridas para que el ingeniero la realice haciendo que el tiempo sea proporcional al tiempo planeado para cada tarea. Se continuará asignando tareas a los ingenieros de esta manera.

Después que se ha balanceado la carga de trabajo, los integrantes del equipo deben registrar esos datos en las formas TAREA y CALENDARIO y generar sus planes personales. Entonces, con esos planes, generarán el plan de equipo compuesto y proporcionarán copias a los demás integrantes de equipo y al instructor.

Actividad 4. Mantener seguimiento del avance del equipo de acuerdo al plan. Es importante porque da a conocer a los integrantes del equipo el avance del proyecto, y les permite tener un entendimiento preciso de lo que ellos deben terminar a tiempo. También brinda el conocimiento requerido para mantener al equipo motivado y preparado para tener éxito.

Cada semana antes del encuentro semanal, el administrador obtiene los datos de la semana anterior de todos los integrantes del equipo. Entonces se registrarán esos datos en las formas TAREA y CALENDARIO del proyecto. Con esos datos se generan las formas de estado del equipo completo sin introducir nuevos datos.

En el seguimiento del estado del proyecto, se aconseja considerar las actividades siguientes:

- Generar las formas RESPL y RESCA de los integrantes de equipo. TSPi automáticamente genera la mayoría de las formas de los ingenieros. Si no se tiene TSPi, debe apoyar a los ingenieros para que completen las formas. Para completarlas, se debe estar seguro que, sin importar cuando los integrantes terminan el trabajo o cualquier componente o tarea del sistema, registren en las formas REGD, REGT, RESTM, TAREA y CALENDARIO los tiempos que ellos invirtieron, cuando terminaron, las horas semanales, los defectos que encontraron, y el tamaño del producto. El Administrador de Planeación deberá tener los datos sobre todas las tareas completadas semanalmente, estando seguro que esos datos estén registrados en las formas TAREA de los ingenieros.
- Después que el Administrador de Planeación tiene los datos individuales, generará las formas de equipo TAREA, CALENDARIO, RESPL y RESCA completas.
- Generar una gráfica del valor ganado contra tiempo mostrando el estado de avance del equipo. Si se está utilizando la herramienta de TSPi, la misma puede generar una gráfica del valor ganado que muestra el estado de avance del proyecto. El valor ganado ayuda a conocer la contribución relativa de las tareas terminadas y el estado de avance del equipo. Esta es la razón por la cual el valor ganado es importante y porque TSPi requiere que todos los ingenieros mantengan los datos personales de valor ganado. Cada semana, los ingenieros desarrollan distintas tareas y completan su trabajo en tiempos no planeados. Cuando el Administrador de Planeación revisa los datos de valor ganado del equipo en la reunión semanal, cada uno puede ver cuales tareas son primero y cuales después. Entonces, ellos pueden balancear sus planes y decidir como recuperar el tiempo perdido. En contra de los beneficios que proporciona el valor ganado puede ocasionalmente rechazarse. Esto puede generar que sea balanceada, otra vez, la carga de trabajo del equipo semanalmente.
- Generar reportes de estado semanales. Cada semana, el AP debe reportar los estados de equipo personales y consolidado al equipo y al instructor. Los datos del equipo muestran el estado del proyecto e identifican donde existen problemas. Esos datos también ayudan al equipo a determinar el riesgo de no poder cumplir con el calendario y a partir de eso, que acciones especiales son necesarias para terminar a tiempo. Los reportes de estado semanales proporcionan los datos necesarios para saber quiénes están haciendo un esfuerzo real y quiénes no. A su vez, ellos pueden ver cuando el equipo está llevando a cabo su plan y cuando necesitan ayuda.
- Generar un breve análisis semanal sobre el desarrollo actual del equipo contra el planeado. El Administrador de Planeación genera un breve análisis del desarrollo del equipo y lo compara con el planeado. Esos breves reportes puede servir como una parte importante de la contribución del Administrador de Planeación para el reporte del ciclo de desarrollo.

Actividad 5. Participar en la generación del reporte del ciclo de desarrollo. El trabajo del Administrador de Planeación bajo la dirección del Líder de Equipo consiste en generar, revisar, corregir y aceptar las secciones asignadas de este reporte. En tales secciones correspondientes, discutirá la planeación y seguimiento del proyecto, explicará cómo fue desarrollado el trabajo y cómo se comparan los resultados con los objetivos y planes del equipo. Finalmente, el AP deberá

comentar cómo efectuó el equipo el trabajo de planeación y seguimiento, cómo podría hacerlo mejor la próxima vez y la razón de eso.

Actividad 6. Actuar como un ingeniero de desarrollo. Como previamente se ha indicado, cada integrante del equipo también desempeña el rol de ingeniero de desarrollo.

A continuación se presenta el guión correspondiente a este rol.

Objetivo	El Administrador de Planeación apoya y guía a los integrantes del equipo en la planeación y seguimiento de su trabajo.
Características del rol	Las características más útiles para los Administradores de Planeación son las siguientes: <ol style="list-style-type: none"> 1. Tiene aptitudes para desarrollar un plan de trabajo. 2. Planea su trabajo. 3. Está interesado en procesar datos. 4. Está preparado para presionar a la gente para seguir y medir su trabajo.
Objetivos y Medidas	Objetivo del integrante de equipo: Ser un integrante de equipo eficaz y cooperativo. <ul style="list-style-type: none"> • Medidas: Las estimaciones de la forma EEC respecto al espíritu del equipo, a la contribución total de roles y al apoyo y ayuda. Objetivo 1: Generar un plan completo, preciso y detallado para el equipo y para cada integrante. <ul style="list-style-type: none"> • Medida 1.1. El plan de equipo cubrió todas las tareas en el ciclo de desarrollo. • Medida 1.2. El plan fue completamente documentado en los formatos TAREA y CALENDARIO. • Medida 1.3. Las horas de tarea promedio fueron menores a 5 y ninguna tarea individual de ingeniero fue mayor a las 10 horas. • Medida 1.4. Las horas semanales y las horas planeadas totales representaron detalladamente los resultados del ciclo actual. Objetivo 2: Proporcionar los reportes del estado del equipo. <ul style="list-style-type: none"> • Medida 2.1. Proporcionó los reportes del estado del equipo completos y precisos semanalmente. • Medida 2.2. Los integrantes del equipo actualizaron sus formas TAREA, CALENDARIO y SEMANA y las entregaron a tiempo. • Medida 2.3. Si uno o más integrantes no reportaron sus datos a tiempo, debió pedir ayuda al Líder de Equipo y al instructor.
Actividades Principales	<ol style="list-style-type: none"> 1. Conducir al equipo en la generación del plan para el siguiente ciclo de desarrollo. <ul style="list-style-type: none"> • Se definen los productos a generar y sus tamaños estimados. • Se especifican las tareas y las horas necesarias para generar los productos. • Se documentan las tareas en la forma TAREA. 2. Dirigir al equipo en la generación del calendario para el siguiente ciclo de desarrollo. <ul style="list-style-type: none"> • Se determinan las horas semanales que cada ingeniero utilizará en el proyecto. • Se registran las horas individuales y de equipo en la forma CALENDARIO. • Se produce la forma CALENDARIO del equipo. 3. Dirigir al equipo en la generación del plan balanceado. <ul style="list-style-type: none"> • Se obtienen los planes detallados de cada ingeniero. • Se identifica el desbalance en la carga de trabajo entre los integrantes del equipo. • Se conduce al equipo en el ajuste de la carga de trabajo para realizar el balance. • Se genera el plan de equipo consolidado. • Se obtienen planes detallados de cada ingeniero. 4. Mantener seguimiento del avance del equipo de acuerdo al plan. <ul style="list-style-type: none"> • Se obtienen los datos semanales de los integrantes del equipo. • Se genera una gráfica del estado del equipo sobre el tiempo y valor ganado semanalmente. • Se genera un análisis semanal del desarrollo actual del equipo contra lo planeado. • Se reportan los estados de equipo personal y consolidado al instructor. 5. Participar en la generación del reporte del ciclo de desarrollo. 6. Actuar como un ingeniero de desarrollo.

Tabla 11.2 Guión del rol Administrador de Planeación

11.3. Rol: Administrador de Desarrollo

11.3.1. Objetivos

Objetivo 1: Generar un producto de alta calidad.

Objetivo 2: Utilizar las destrezas y habilidades de los integrantes del equipo.

11.3.2. Destrezas y habilidades

1. Disfruta en la realización de la construcción de los elementos de software.
2. Desempeña el rol de ingeniero de software y le agrada la experiencia de dirigir el trabajo de diseño y desarrollo para un proyecto.
3. Es un diseñador competente y se siente capaz de dirigir al equipo de diseño, cuyo trabajo es visualizar una estructura ordenada, a partir de conceptos e ideas para generar una sola idea coherente.
4. Tiene disponibilidad para escuchar y capacidad para unir ideas y aportaciones de otras personas.

11.3.3. Actividades principales del Administrador de Desarrollo

Actividad 1. Dirigir al equipo en la generación de la estrategia de desarrollo. Esta estrategia guía el diseño del producto y el trabajo de desarrollo. Se produce para generar una base que permita hacer estimaciones y un plan. En la generación de esta estrategia se deben seguir los pasos mencionados a continuación: establecer los criterios de la estrategia; proponer y evaluar diferentes estrategias; seleccionar una propuesta estratégica; producir un diseño conceptual; asignar funciones para cada ciclo; registrar la asignación en el formato ESTRA; definir cómo subdividir esas funciones en los elementos del producto y decidir cómo integrar las partes que se van a implementar dentro del producto de trabajo.

Actividad 2. Dirigir al equipo en las estimaciones de tamaño y tiempos preliminares. Como parte de la fase de estrategia, debe hacerse un estimado preliminar del tamaño y el tiempo de desarrollo para cada función del producto, tales datos se anotan en el formulario ESTRA. Con esta información, se puede tomar una decisión inicial sobre cuales funciones aplicar y en cual ciclo de desarrollo. El responsable de dirigir la reunión (ACP) documenta las estimaciones en tamaño y tiempo de desarrollo en el formato ESTRA y entrega una copia al Líder del Equipo para anexarla en la carpeta del proyecto.

Actividad 3. Dirigir el desarrollo de la especificación de requerimientos (ERS). La especificación de requerimientos describe las funciones que serán desarrolladas. El objetivo del Administrador de Desarrollo es resolver los conflictos que surjan en el equipo respecto a la definición del problema, para que todos los integrantes entiendan con precisión lo que se pretende construir. Es importante que todos los integrantes participen en el desarrollo e inspección de la ERS para que se termine el trabajo más rápido, para que el resultado sea de buena calidad y para que el equipo entero tenga claro entendimiento del producto que se pretende construir.

Actividad 4. Dirigir al equipo en la generación de un diseño de alto nivel. La siguiente tarea es dirigir al equipo en la generación de un diseño de alto nivel del producto. Se necesita este diseño antes que se genere la especificación de diseño de software (EDS). El equipo deberá debatir la estructura del producto, considerando los caminos alternativos para asignar las funciones a los componentes e identificar las relaciones entre los mismos. Aunque existen distintas maneras de juzgar la calidad de un diseño de alto nivel, una de las mejores es usar las medidas de acoplamiento, fuerza y acoplamiento/fuerza (descritas en el Apéndice A de la bibliografía de TSPi). En cada diseño alternativo deben estimarse esos valores por cada componente. Al comparar los

diseños se encontrará que los componentes diseñados con bajos rangos de acoplamiento y acoplamiento/fuerza son más fáciles de diseñar, más fáciles de modificar y menos propensos al error. De lo contrario, los componentes con altos rangos de acoplamiento y/o alto acoplamiento/fuerza, generalmente tiene más diseños complejos, generan más código, tiene más defectos, y ocupan más tiempo para el desarrollo que en cualquier otro caso.

El paso final del diseño de alto nivel es listar en el formato ESTRA los componentes, tamaños, tiempos de desarrollo probables, y contenidos funcionales. También debe representarse el diseño, mediante algún tipo de diagrama, en el cual se mostrarán los componentes y sus relaciones.

Actividad 5. Dirigir al equipo en la generación de la especificación del diseño (EDS). Los equipos a menudo cometen el error de no comenzar a escribir la EDS hasta que sienten que han completado todo su trabajo de diseño. Cuando comienzan a documentarlo, descubren que mucho del trabajo hecho está incompleto e impreciso. Antes de generar la EDS, debe estar seguro que el ACP ha establecido los estándares de diseño del equipo; si no ha sido así se deberán establecer antes de escribirla.

Actividad 6. Dirigir al equipo en la implementación del producto. Antes de iniciar la implementación se deberá verificar que todos los estándares de implementación requeridos han sido establecidos y que el equipo está de acuerdo con ellos. Si no es así, deberán establecerse dichos estándares antes de comenzar la aplicación. El primer paso de la implementación es definir las tareas que se harán y que el LE asigne esas tareas entre todos los integrantes. Posteriormente, todos seguirán el proceso TSPi para desarrollar las partes asignadas de la implementación.

Actividad 7. Dirigir al equipo en el desarrollo de la construcción, integración y planes de prueba del sistema. Para la fase de prueba deberá dividirse al equipo en dos grupos: uno que haga los planes de pruebas y las pruebas y otro que produzca la documentación para el usuario. Si el AD planea trabajar en la documentación, el equipo de pruebas deberá asignar un integrante que será el AD alterno. Esta persona conducirá la planeación de pruebas y el trabajo de pruebas. De la misma forma, si el AD planea trabajar en las pruebas, los integrantes del equipo de documentación seleccionarán un AD alterno que conduzca su trabajo de documentación. El grupo de pruebas revisará los planes de pruebas desarrollados durante las fases de diseño y requerimientos y los actualizará reflejando cualquier cambio. El grupo de documentación planeará el trabajo de documentación. En ambos casos, todos los integrantes del equipo que hagan cada parte del trabajo, participarán en su planeación. El plan de pruebas que describe TSPi en el capítulo referente a tal tema sigue la estrategia de planeación PSP, la cual consiste en realizar una estrategia de pruebas o propuesta conceptual; estimar o medir los tamaños de los productos que se van a probar; estimar los números y tamaños de los casos de pruebas y otros materiales de apoyo de pruebas, estimar el tiempo para realizar las pruebas y el trabajo de desarrollo de pruebas y asignar ese trabajo entre los integrantes del equipo.

Actividad 8. Dirigir al equipo en el desarrollo de materiales de prueba y ejecución de las pruebas. Siguiendo el plan de pruebas, los integrantes del equipo desarrollarán los guiones de prueba, los datos de prueba y los programas necesarios. Para cada prueba, estimarán el tiempo de ejecución de la prueba libre de defectos y anticiparán los resultados de prueba. En el desarrollo de las pruebas, los ingenieros deberán hacer planes personales para cada tarea, registrando su tiempo, tamaño y datos defectuosos, y documentando los resultados finales usando las formas REGD, REGT y RESTM.

Actividad 9. Dirigir al equipo en la generación de la documentación de los productos del usuario. En la producción de la documentación se seguirá otra vez la estrategia estándar PSP/TSP, la cual consiste en generar el diseño conceptual del documento. Se identificarán las funciones que se completarán en el ciclo de desarrollo que se esté llevando a cabo, y aquellas que se cambiarán o aumentarán en ciclos posteriores. Para minimizar los cambios en la documentación, se recomienda concentrarse en la documentación de las funciones que se completarán en el ciclo de desarrollo actual.

Antes de empezar, el grupo de documentación planeará el trabajo de documentación; se asignará entre los integrantes del equipo, se desarrollará el documento, se revisará el trabajo de cada integrante y se harán las correcciones necesarias. Finalmente, se combinarán las secciones del documento como borrador y se distribuirá al equipo para la revisión. El ACP dirigirá al equipo en la revisión del borrador de la documentación del usuario y tendrá a cada ingeniero corrigiendo su parte. Finalmente, se integrarán las partes corregidas dentro del documento final y se distribuirán al Instructor y al equipo.

Actividad 10. Participar en la generación del reporte del ciclo de desarrollo. Como Administrador de Desarrollo, la tarea final consiste en preparar las secciones correspondientes del reporte de ciclo. En esta tarea se seguirán las órdenes del Líder de Equipo para producir, revisar, corregir y aceptar las partes correspondientes. Se debe definir el trabajo de desarrollo, evaluar cómo se desarrolló y compararlo con los objetivos y estándares del equipo. Finalmente, se comenta cómo se hizo el desarrollo, las mejoras que se podrían aplicar. El AD deberá participar en las revisiones por colegas del equipo.

Actividad 11. Actuar como un ingeniero de desarrollo. Como previamente se ha indicado, cada integrante del equipo también desempeña el rol de ingeniero de desarrollo.

A continuación se presenta el guión correspondiente a este rol.

Objetivo	El Administrador de Desarrollo conduce y guía al equipo en la definición, diseño, desarrollo y pruebas del producto.
Características del rol	Las características más útiles para los administradores de desarrollo son las siguientes: <ol style="list-style-type: none"> 1. Tiene gusto por construir cosas. 2. Desea ser un ingeniero de software y le agrada dirigir el trabajo de diseño y desarrollo para un proyecto. 3. Es un administrador competente y siente que puede conducir al equipo de desarrollo. 4. Está familiarizado con los métodos de diseño. 5. Está dispuesto a escuchar las ideas de diseño de otras gentes y poder objetiva y lógicamente comparar las cualidades de esas ideas de diseño con las propias.
Objetivos y Medidas	Objetivo del integrante de equipo: Ser un integrante de equipo eficaz y cooperativo. <ul style="list-style-type: none"> • Medidas: Las estimaciones de equipo de la forma EEC respecto al espíritu del equipo, la contribución completa, y al apoyo y ayuda. Objetivo 1: Generar un producto de alta calidad. <ul style="list-style-type: none"> • Medida 1.1. El equipo generó un producto útil y completamente documentado que cumplió los requerimientos básicos de la definición del problema. • Medida 1.2. Se mantiene seguimiento en los requerimientos a partir de la definición del problema en la ERS, en la EDS y en la implementación final. • Medida 1.3. El diseño del producto está completamente documentado y cumple los estándares de diseño del equipo. • Medida 1.4. La implementación representa el diseño. • Medida 1.5. El producto cumplió todos los criterios de calidad. • Medida 1.6. El producto cumplió sus objetivos funcionales y de operación. Objetivo 2: Utilizar todas las destrezas y habilidades de los integrantes del equipo. <ul style="list-style-type: none"> • Medida 2.1. Las evaluaciones de las formas EEC sobre el desempeño del rol de Administrador de Desarrollo. • Medida 2.2. Las evaluaciones de las formas EEC sobre el apoyo y ayuda del Administrador de Desarrollo. • Medida 2.3. Las evaluaciones de las formas EEC sobre la calidad del producto.
Actividades Principales	<ol style="list-style-type: none"> 1. Dirigir al equipo en la generación de la estrategia de desarrollo. 2. Dirigir al equipo en las estimaciones de tamaño y tiempo preliminares. 3. Dirigir el desarrollo de la especificación de requerimientos. (ERS) 4. Dirigir al equipo en la generación de un diseño de alto nivel. 5. Dirigir al equipo en la generación de la especificación del diseño(EDS). 6. Dirigir al equipo en la implementación del producto. 7. Dirigir al equipo en el desarrollo de la construcción, integración y planes de prueba del sistema. 8. Dirigir al equipo en el desarrollo de materiales de prueba y ejecución de las pruebas.

9.	Dirigir al equipo en la generación de la documentación de los productos de usuario.
10.	Participar en la generación del reporte del ciclo de desarrollo.
11.	Actuar como un ingeniero de desarrollo.

Tabla 11.3 Guión del rol Administrador de Desarrollo

11.4. Rol: Administrador de Calidad y Proceso

11.4.1. Objetivos

Objetivo 1. Realizar reportes de los datos de TSPi. La medida de éxito para este objetivo es el registro de los datos de TSPi, la documentación de las propuestas de mejora (forma Propuesta de Mejora al Proceso - PMP), la documentación del proceso y estándares. A continuación se muestran los procesos y estándares que se debieron haber realizado: proceso de administración de cambios; proceso de control y seguimiento; estándares de diseño; estándares de codificación; estándares de nombre y glosario; estándares de reuso; estándar de calidad (forma completa RESCA) y estándar de contador de LOC.

Objetivo 2. Seguir TSPi de acuerdo al plan de calidad y generar un producto de calidad. Este objetivo se mide analizando la manera en que los integrantes del equipo utilizaron TSPi para realizar los objetivos del plan de calidad. Otra medida es el desempeño de calidad del equipo de acuerdo al plan de calidad. A continuación se presentan los parámetros que se aplican para tales medidas: pruebas del sistema libres de defectos; defectos/KLOC en la compilación; defectos/KLOC en las pruebas unitarias; defectos/KLOC en la pruebas del sistema; porcentaje de defectos totales eliminados antes de la primera compilación; porcentaje de defectos totales eliminados antes de las pruebas unitarias y porcentaje de defectos totales eliminados antes de las pruebas del sistema.

Objetivo 3. Dirigir y registrar las inspecciones. A continuación se presentan las medidas que se aplican en este objetivo: todas las inspecciones se realizaron de acuerdo al proceso de inspección, en el cual se aplica el guión INS y se utiliza el Reporte de Inspección (REPINS); todas las inspecciones del equipo se realizaron de acuerdo a un plan de calidad del equipo; se completaron los Reportes de Inspección en todas las inspecciones y se proporcionaron copias de las formas REPINS y de las reuniones al LE para anexarlas en la carpeta.

Objetivo 4. Registrar todas las reuniones e incluirlas en la carpeta. En este objetivo se mide el porcentaje de las reuniones semanales registradas en la carpeta (así como de reuniones extras).

11.4.2. Destrezas y habilidades

1. Tiene interés en generar un producto de calidad. TSPi se enfoca en la calidad desde el inicio del proceso; por eso se aplican tareas y tiempo de desarrollo desde el inicio. Aunque el tiempo se reduce en la compilación y pruebas. Cumplir con los objetivos de calidad consiste en cumplir con los costos y el calendario establecido.
2. Tiene interés en el proceso y en las medidas del mismo. Para eso deberá conocer los guiones, comprenderlos y aplicarlos.
3. Tiene experiencia con métodos de inspección y revisión.
4. Tiene habilidad para realizar revisiones constructivas.

11.4.3. Actividades principales del Administrador de Calidad y Proceso

Actividad 1. Dirigir al equipo en la generación y seguimiento del plan de calidad. En la fase de desarrollo el ACP dirige al equipo en la generación del plan de calidad, mediante los estándares y objetivos a aplicar y esta pendiente del seguimiento en el trabajo.

Actividad 2. Informar los problemas de calidad al LE, al equipo y al instructor. Para tal actividad debe analizar los datos y determinar si el trabajo realizado reúne los criterios de calidad. Otra responsabilidad consiste en manejar las revisiones de defectos de pruebas. Es responsable de que cada defecto encontrado se analice en las reuniones semanales y que se registre cualquier observación.

Actividad 3. Dirigir al equipo en la definición y documentación de procesos y en la mejora de los mismos. Debe ayudar al equipo a definir y documentar sus procesos, lo cual se realiza en la fase de planeación. Por eso, se recomienda incluir cualquier actividad de trabajo en el plan de desarrollo. Otra parte importante en la definición de un proceso consiste en establecer el proceso de propuesta de mejoras, utilizando la forma PMP. Tales propuestas se pueden realizar semanalmente.

Actividad 4. Establecer y mantener los estándares de desarrollo del equipo. Se deben establecer y mantener los estándares. Ejemplos de estándares son los de codificación, diseño, documentación, nombres, manejo de errores, interfaces, mensajes, pantalla, reporte y LOC. También es responsable de documentar los estándares, revisarlos y aprobarlos. Si el ACP desarrolla los estándares, los ingenieros del equipo deben revisarlos. Se debe cerciorar de la aplicación de los mismos. Y se aconseja revisar el estado de los estándares en las reuniones semanales.

Actividad 5. Revisar y aprobar todos los productos antes de entregarlos a la Mesa de Control de Configuración. Lleva a cabo el aseguramiento de calidad al identificar los resultados de los productos que se generan. Para eso se deben revisar los datos del proceso, lo cual consiste en evaluar las tasas de tiempo de desarrollo, niveles de defectos y revisar el producto mismo. Al aprobar un producto, se certifica que los datos del proceso estén integrados en el producto y que este último haya sido desarrollado bajo un proceso, para anexarlo como línea base.

Actividad 6. Dirigir las inspecciones del equipo. Para esta actividad se debe utilizar el guión INS. Se debe basar en las siguientes actividades: se define cuántos ingenieros deben estar en el equipo de inspección; se asegura que el autor del producto esté preparado para la inspección; en la primera inspección, se describe el proceso de inspección (considerando el producto que será inspeccionado, el tiempo y lugar); se asegura de que todos los ingenieros hayan terminado la preparación de la inspección, y que los datos de tiempo y defectos se registren en la forma REPINS; se verifica que los revisores hayan permanecido tiempo suficiente en la detección de defectos; se identifican los defectos; al finalizar la inspección, se verifica que el equipo haya encontrado la mayoría de los defectos (consulte el Apéndice C de la bibliografía de TSPi); se asegura que todos los datos estén registrados en las reuniones y los defectos mayores en la forma REGD y al final de la inspección, se asegura que los datos de la misma se registren en la forma INS y se entregue al LE para anexarla en la carpeta.

Actividad 7. Registrar las reuniones del equipo. Se registran las acciones a aplicar, el responsable de eso y la fecha.

Actividad 8. Participar en la generación del reporte del ciclo de desarrollo. Se analiza el contenido desde una perspectiva de calidad y proceso, que tan precisos y completos son los datos indicados, el seguimiento del proceso y el cumplimiento en los objetivos de calidad establecidos por el equipo. También participa en las evaluaciones EEC.

Actividad 9. Actuar como un ingeniero de desarrollo. Como previamente se ha indicado, cada integrante del equipo también desempeña el rol de ingeniero de desarrollo.

A continuación se presenta el guión correspondiente a este rol.

Objetivo	El Administrador de Calidad y Proceso apoya al equipo en la definición de las necesidades del proceso, en elaborar el plan de calidad y en mantener seguimiento en el proceso y en la calidad del producto.
Características del rol	Las características más útiles para los administradores de calidad y proceso son las siguientes: <ol style="list-style-type: none"> 1. Está consciente de la calidad que debe existir en el software que se va a desarrollar. 2. Está interesado en el proceso y en las medidas del mismo. 3. Tiene experiencia en métodos de revisión e inspección. 4. Tiene capacidad para revisar el trabajo de otras personas sin tomar posturas personales.
Objetivos y Medidas	Objetivo del integrante de equipo: Ser un integrante de equipo eficaz y cooperativo. <ul style="list-style-type: none"> • Medidas: Las estimaciones de la forma EEC respecto al espíritu del equipo, a la contribución total de roles y al apoyo y ayuda. Objetivo 1: Realizar reportes de los datos de TSPi. <ul style="list-style-type: none"> • Medida 1.1. El equipo generó y utilizó los datos requeridos por TSPi. Objetivo 2: Seguir TSPi y generar un producto de calidad. <ul style="list-style-type: none"> • Medida 2.1. Que tan bien el equipo siguió el TSPi. • Medida 2.2. Que tan bien el desempeño cumplió con la calidad, de acuerdo al plan de calidad. • Medida 2.3. El nivel de información sobre problemas de calidad al instructor y al Líder de Equipo. • Medida 2.4. El nivel de calidad del rol sin tomar en cuenta asuntos personales con integrantes del equipo. Objetivo 3: Dirigir y registrar las inspecciones. <ul style="list-style-type: none"> • Medida 3.1. Todas las inspecciones son dirigidas por el guión INS y los estándares de calidad del equipo. • Medida 3.2. En todas las inspecciones se completan las formas REPINS y los defectos mayores se registran en las formas REGD. Objetivo 4 : Registrar todas las reuniones e incluirlas en la carpeta. <ul style="list-style-type: none"> • Medida 4. Se miden los reportes de las reuniones de acuerdo con las que se incluyeron en la carpeta.
Actividades Principales	<ol style="list-style-type: none"> 1. Dirigir al equipo en la generación y seguimiento del plan de calidad. 2. Informar al equipo, al LE y al instructor los problemas de calidad. 3. Dirigir al equipo en la definición y documentación de procesos y en la mejora de los mismos. 4. Establecer y mantener los estándares de desarrollo del equipo. 5. Revisar y aprobar todos los productos antes de entregarlos a la Mesa de Control de Configuración (MCC). 6. Dirigir las inspecciones del equipo. 7. Registrar las reuniones del equipo. 8. Participar en la generación del reporte del ciclo de desarrollo. 9. Actuar como un ingeniero de desarrollo.

Tabla 11.4 Guión del rol Administrador de Calidad y Proceso

11.5. Rol: Administrador de Apoyo

11.5.1. Objetivos

Objetivo 1. Facilitar al equipo las herramientas y métodos de apoyo para realizar su trabajo. También es responsable de indicar y ayudar al equipo el uso de las herramientas y los métodos. Las medidas son las siguientes: el equipo tiene herramientas y métodos portables (por ejemplo, los ingenieros tiene un sistema de administración de configuración, un sistema de control y seguimiento, un ambiente de desarrollo; etc.); el Administrador de Apoyo ha revisado las solicitudes de cambios a las herramientas y las ha modificado en caso necesario.

Objetivo 2. No aplicar ningún cambio no autorizado en la línea base. Las medidas son las siguientes: todos los elementos del producto final están bajo control de configuración; todos los cambios a los productos bajo control de configuración son aprobados por la Mesa de Control de Configuración; cuando se realizan cambios al código, se reflejan en la documentación del diseño de la línea base.

Objetivo 3. Reportar semanalmente todos los asuntos y riesgos del equipo de acuerdo al sistema de seguimiento de riesgos. La medida es la siguiente: el porcentaje de todos los asuntos y riesgos se registran y se les aplica seguimiento mediante el sistema de seguimiento de riesgos.

Objetivo 4. Establecer objetivos de reuso para el ciclo de desarrollo. Las medidas son las siguientes: el Administrador de Apoyo es responsable de colaborar con el Administrador de Calidad y Proceso en la definición de los estándares de reuso; el AA ha establecido y aplicado mantenimiento a la lista de partes de reuso disponibles; se miden los porcentajes de reuso y de reuso nuevo del equipo y se les aplica seguimiento; se aplicó el reuso durante el primer ciclo y se incrementó el grado de reuso en cada ciclo de desarrollo.

11.5.2. Destrezas y habilidades

1. Está interesado en herramientas y métodos.
2. Es un usuario competente y apoya al equipo en sus necesidades.
3. Tiene experiencia en diferentes herramientas de apoyo y en sistemas.
4. Está familiarizado con las herramientas que se van a usar en cada proyecto.

11.5.3. Actividades principales del Administrador de Apoyo

Actividad 1. Dirigir al equipo en la identificación de necesidades que deberá cubrir y ser responsable de obtener las herramientas y facilidades. Es conveniente tomar en cuenta las sugerencias de los integrantes del equipo porque es posible que tengan ideas o que sus ayudas puedan ser útiles. Posiblemente, pueden saber cómo obtener ciertas herramientas o que puedan apoyar el uso de las mismas.

Actividad 2. Trabajar con la Mesa de Control de Configuración (MCC) y administrar el sistema de control de cambios. Colabora con la Mesa de Control de Configuración para establecer el procedimiento para la administración de cambios, otro integrante es el Administrador de Desarrollo.

Actividad 3. Administrar el sistema de la administración de la configuración, para lo cual deberá: generar el plan de la administración de la configuración (el cual deberá ser revisado y aprobado por el equipo); establecer un procedimiento para almacenar y mantener seguimiento en todos los elementos de los productos de línea base y sus versiones; establecer la manera en la que los ingenieros puedan entregar y recuperar los elementos del producto a partir del sistema de la administración de la configuración y administrar el sistema para el manejo de cambios a los elementos de los productos de la línea base.

Para mayor información consulte el Apéndice B de la bibliografía de TSPI.

Actividad 4. Mantener el glosario del sistema, porque este documento incluirá secciones para los nombres de los módulos, variables globales, nombres de las partes de reuso, etc. También será utilizado como referencia para mensajes estándar, interfaces, condiciones de error y muchos otros elementos del sistema.

Actividad 5. Mantener el sistema de seguimiento de riesgos y asuntos del equipo. Es responsabilidad del AA mantener seguimiento en los riesgos y asuntos del equipo, registrar cada

riesgo o asunto para darle seguimiento y que no se olviden. La información necesaria para el registro es: asunto y número asignado, momento en el que el asunto se identificó, responsable de manejarlo, fecha en la que será tratado y fecha de cierre del asunto.

Respecto a riesgos se deberá registrar la misma información, añadir el impacto del mismo y el impacto anticipado.

Actividad 6: Fomentar el reuso en el equipo. Se puede ahorrar gran tiempo y esfuerzo tomando las oportunidades de reuso. Es importante que este asunto se trate en el diseño y en la implementación. El AA deberá: asegurarse de que los estándares de reuso sean utilizados por el equipo de trabajo; fomentar entre los integrantes que cada uno colabore durante el diseño y la implementación en el reuso; estar informado de las partes de reuso desarrolladas; mantener una lista de las partes de reuso y sus especificaciones; mantener un repositorio bajo control de configuración de las partes de reuso; dar seguimiento del reuso y del progreso y estado semanal en las reuniones.

Actividad 7. Participar en la generación del reporte del ciclo. La tarea final consiste en preparar las secciones del reporte del ciclo, para eso trabajará con el Líder de Equipo para generar, revisar, corregir y entregar las partes asignadas del reporte. En la sección correspondiente indicará el apoyo que brindó al equipo y los beneficios y desventajas del mismo, la efectividad del proceso de la administración de la configuración para controlar los productos de trabajo y el nivel de reuso. Comentaré cómo se llevó a cabo el plan de la administración de la configuración y que tan bien utilizó los principios de reuso para mejorar la productividad del equipo. Indicará cómo realizar mejoras en todos esos aspectos y participará en las revisiones de la forma EEC.

Actividad 8. Actuar como un ingeniero de desarrollo. Como previamente se ha indicado, cada integrante del equipo también desempeña el rol de ingeniero de desarrollo.

A continuación se presenta el guión correspondiente a este rol.

Objetivo	El Administrador de Apoyo colabora con el equipo en la determinación, obtención y mantenimiento de las herramientas necesarias para cumplir con las necesidades administrativas y aplicar la tecnología definida.
Características del rol	Las características más útiles para los administradores de apoyo son las siguientes: <ol style="list-style-type: none"> 1. Está interesado en herramientas y en métodos. 2. Puede apoyar al equipo en las necesidades del mismo. 3. Tiene experiencia en el manejo de herramientas y sistemas de apoyo. 4. Está familiarizado con las herramientas que se utilizarán en el proyecto.
Objetivos y Medidas	<p>Objetivo del integrante de equipo: Ser un integrante de equipo eficaz y cooperativo.</p> <ul style="list-style-type: none"> • Medidas: Las estimaciones de la forma EEC respecto al espíritu del equipo, a la contribución total de roles y al apoyo y ayuda. <p>Objetivo 1: Tener herramientas y métodos para el desarrollo del trabajo.</p> <ul style="list-style-type: none"> • Medida 1.1. El equipo tiene un sistema de administración de cambios, un sistema de seguimiento de asuntos, un sistema de administración de configuración y un ambiente de desarrollo. • Medida 1.2: El equipo utilizó las herramientas <p>Objetivo 2: No aplicar ningún cambio no autorizado en la línea base.</p> <ul style="list-style-type: none"> • Medida 2.1. Todos los elementos del producto final están bajo control de Configuración. • Medida 2.2. Todos los cambios pasan por MCC. • Medida 2.3. Si se realizaron cambios en el código, se encuentran en la documentación del diseño de la línea base. <p>Objetivo 3: Reportar semanalmente todos los asuntos y riesgos del equipo en el Reporte de Asuntos y Riesgos (RAR).</p> <ul style="list-style-type: none"> • Medida 3. El porcentaje de asuntos y riesgos registrados y a los que se les ha dado seguimiento. <p>Objetivo 4 : Establecer objetivos de reuso para el ciclo de desarrollo.</p> <ul style="list-style-type: none"> • Medida 4.1 El equipo tiene una lista de partes de reuso.

	<ul style="list-style-type: none"> • Medida 4.2 Se tiene seguimiento sobre los porcentajes de reuso (nuevo como anterior). • Medida 4.3. El equipo aplicó reuso en el primer ciclo de desarrollo. • Medida 4.4. El nivel de reuso se incrementó en cada ciclo.
Actividades Principales	<ol style="list-style-type: none"> 1. Dirigir al equipo a determinar las necesidades de apoyo y a obtener las herramientas y facilidades. 2. Colaborar con la MCC y administrar el sistema de control de cambios, revisando los cambios en los productos controlados, evaluando el impacto y beneficio de cada cambio y recomendando al equipo que cambios hacer. 3. Administrar el sistema de administración de configuración, manteniendo una copia protegida de todos los elementos controlados, haciendo los cambios aprobados sólo a la versión controlada y manteniendo copias de los elementos y versiones. 4. Mantener el glosario del sistema. 5. Mantener el sistema de seguimiento de riesgos y asuntos del equipo. 6. Fomentar el reuso en el equipo. 7. Participar en la generación del reporte del ciclo. 8. Actuar como un ingeniero de desarrollo.

Tabla 11.5 Guión del rol Administrador de Apoyo

Parte II

Aplicación de TSPi al curso de Ingeniería de Software

Capítulo 12. Aplicación de TSPi

Durante la aplicación del proceso se utilizaron los guiones de TSPi, se completaron las formas propuestas y se realizaron los procedimientos indicados. Este capítulo integra diversas secciones que se decidieron incluir para presentar la experiencia de la aplicación de TSPi al grupo de Ingeniería de Software de la Universidad. El contenido de cada una utilizó como base los hechos que se suscitaron durante el transcurso del semestre, los cuales se fueron integrando en la documentación correspondiente. En cada sección se indica que documentación se utilizó como base para su contenido.

12.1. Antecedentes académicos

Uno de los objetivos de la encuesta aplicada a diez alumnos del grupo de Ingeniería de Software, consistió en obtener información acerca de las materias que habían cursado hasta el sexto semestre de la Licenciatura en Ciencias de la Computación. Cabe mencionar, que aparte de las materias obligatorias estaban las de carácter optativo. Cada una presentó un número irregular en el total de alumnos que las habían cursado. Uno de los motivos de estos resultados se debe a que en la licenciatura no existía el acuerdo de cursar las materias en serie. Por lo tanto, un alumno podía decidir cursar asignaturas del semestre actual o posteriores. Así como también, existía la posibilidad de que la materia no hubiera sido acreditada.

La siguiente tabla presenta el conjunto de materias que han permitido establecer el antecedente académico del grupo. También se indica el número del semestre en el que se debía cursar cada una. Las materias optativas pueden ser cursadas a partir del quinto semestre.

Materia	Semestre
Matemáticas Discretas	1o.
Introducción a las Ciencias de la Computación I	2o.
Introducción a las Ciencias de la Computación II	3o.
Probabilidad y Estadística	3o.
Análisis de Algoritmos	4o.
Análisis Lógico	4o.
Arquitectura de Computadoras	5o.
Diseño de Sistemas Digitales	5o.
Teoría de la Computación	5o.
Inteligencia Artificial	6o.
Lenguajes de Programación	6o.
Sistemas Operativos	6o.
Redes de Computadoras	7o.
Bases de Datos I	7o.
Redes Neuronales	Optativa
Diseño y Programación Orientada a Objetos	Optativa
Programación Funcional y Lógica	Optativa

Tabla 12.1 Materias y los semestres en que se deben cursar

La siguiente gráfica representa el número de alumnos que cursaron cada materia, las cuales se identifican por una letra que posteriormente se define en la tabla Claves de las materias.

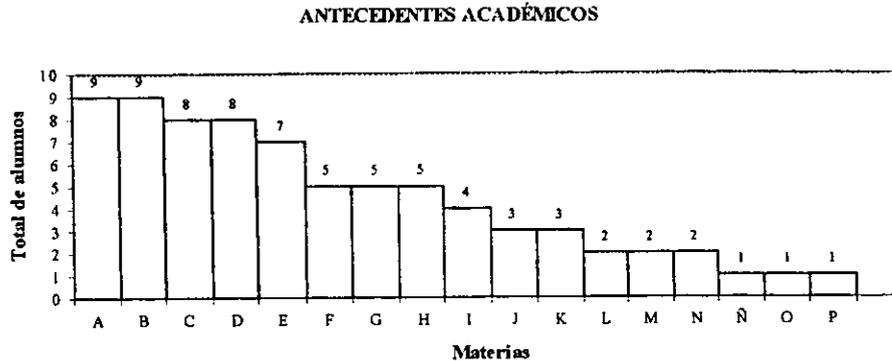


Figura 12.1 Número de alumnos que cursaron las materias

Materias	Clave
Introducción a las Ciencias de la Computación I	A
Introducción a las Ciencias de la Computación II	B
Análisis de Algoritmos	C
Arquitectura de Computadoras	D
Inteligencia Artificial	E
Lenguajes de Programación	F
Matemáticas Discretas	G
Análisis Lógico	H
Redes de Computadoras	I
Diseño de Sistemas Digitales	J
Teoría de la Computación	K
Bases de Datos I	L
Redes Neuronales	M
Sistemas Operativos	N
Diseño y Programación Orientada a Objetos	Ñ
Probabilidad y Estadística	O
Programación Funcional y Lógica	P

Tabla 12.2 Claves de las Materias

Como lo muestra la gráfica, la mayor parte de las materias tuvieron un número irregular de alumnos inscritos; probablemente por las razones mencionadas en un inicio. Ellos ya deberían de haber cursado las materias que se representan con las letras A, B, C, D, G, H, J y K obligatoriamente. Además la gráfica refleja que hubo alumnos que ya habían cursado materias del séptimo semestre; por lo tanto, se deduce que los alumnos eran irregulares y su preparación académica se evaluó como dispareja. Por eso, al momento de desarrollar cada una de las fases, realizar todas aquellas actividades indicadas y generar los productos requeridos los problemas en el desarrollo del proceso aumentaron porque no habían cursado y acreditado las materias que preceden a Ingeniería de Software.

El desarrollo del curso se realizó bajo el gran riesgo de que los alumnos no habían sido entrenados bajo PSP, el cual es un requisito para cursar TSPi. Cuando se hacía referencia a temas de esa bibliografía, el Instructor dedicaba parte de la clase a exponer la información necesaria y a proporcionar la bibliografía requerida.

12.2. Método de trabajo

El método de trabajo que se aplicó en el curso, entre los equipos y el Instructor se expone a continuación.

Al inicio, el Instructor entregó a los alumnos el plan de trabajo de la materia, hecho en base al que se propone en el libro de TSPi. Al inicio del curso el plan se distribuyó en tres ciclos, pero debido a los problemas que surgieron durante el desarrollo, se redujo a dos ciclos.

Las siguientes tablas describen los dos planes de trabajo realizados.

Objetivo		Guía para el equipo de desarrollo de un producto de software.
Criterio de entrada		<ul style="list-style-type: none"> El instructor guía y apoya a uno o más equipos de 5 estudiantes. Los estudiantes tienen el entrenamiento en PSP. El instructor tiene materiales, facilidades y recursos para apoyar a los equipos. El instructor ha descrito el objetivo general del producto.
General		<p>El proceso de TSPi está diseñado para:</p> <ol style="list-style-type: none"> Desarrollar el producto de software de tamaño pequeño o mediano en 2 o 3 ciclos de desarrollo. Desarrollar un producto más pequeño en cada ciclo. Producir los elementos del producto tales como documento de requerimientos, especificación de diseño, plan de pruebas, etc., en cada ciclo. <p>Siga los guiones que apliquen en su proyecto y modo de operación.</p>
Semana	Paso	Actividades
1 10-14 Abril	Revisión	<ul style="list-style-type: none"> Introducción al curso y revisión de PSP. Se leen los capítulos 1 y 2 del libro y el Apéndice A.
2 24-28 Abril	LAN1	<ul style="list-style-type: none"> Se revisan los objetivos del curso y se asignan los equipos de estudiantes y los roles. Se lee el capítulo 3 del libro, el Apéndice B y uno de los capítulos del 11 al 15.
	ESTRA1	<ul style="list-style-type: none"> Se produce el diseño conceptual, se establece la estrategia de desarrollo, se hacen estimaciones del tamaño y se evalúan riesgos. Se lee el capítulo 4.
3 01-05 Mayo	PLAN1	<ul style="list-style-type: none"> Se producen los planes del equipo y de los ingenieros para el ciclo 1. Se lee el capítulo 5 y el Apéndice C.
4 08-12 Mayo	REQ1	<ul style="list-style-type: none"> Se definen e inspeccionan los requerimientos para el ciclo 1. Se produce el plan de pruebas del sistema y el material de apoyo. Se lee el capítulo 6 y las secciones de pruebas del capítulo 9.
5 15-19 Mayo	DIS1	<ul style="list-style-type: none"> Se produce e inspecciona el diseño de alto nivel para el ciclo 1. Se produce el plan de pruebas de integración y el material de apoyo. Se lee el capítulo 7.
6 22-26 Mayo	IMP1	<ul style="list-style-type: none"> Se implementa e inspecciona el ciclo 1. Se produce el plan de pruebas unitarias y el material de apoyo. Se lee el capítulo 8.
7 29 Mayo-02 Junio	PRUEBAS1	<ul style="list-style-type: none"> Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 1. Se produce la documentación para el usuario del ciclo 1. Se lee el capítulo 9.
8 05-09 Junio	PM1	<ul style="list-style-type: none"> Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 1. Se producen las evaluaciones de roles y de equipos para el ciclo 1. Se leen los capítulos 10, 16, 17 y 18.
	LAN2	<ul style="list-style-type: none"> Se reorganizan equipos y roles para el ciclo 2. Se leen los capítulos del 11 al 15.
	ESTRA2, PLAN2	<ul style="list-style-type: none"> Se produce la estrategia y el plan para el ciclo 2. Se evalúan riesgos.

9 12-16 Junio	REQ2	<ul style="list-style-type: none"> Se ajustan los requerimientos y el plan de pruebas para el ciclo 2.
	DIS2	<ul style="list-style-type: none"> Se produce e inspecciona el diseño de alto nivel para el ciclo 2. Se actualiza el plan de pruebas de integración para el ciclo 2.
10 19-23 Junio	IMP2	<ul style="list-style-type: none"> Se implementa e inspecciona el ciclo 2, se produce el plan de pruebas unitarias.
	PRUEBAS2	<ul style="list-style-type: none"> Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 2. Se produce la documentación de usuario para el ciclo 2.
11 26-30 Junio	PM2	<ul style="list-style-type: none"> Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 2. Se producen las evaluaciones de roles y de equipos para el ciclo 2.
	LAN3	<ul style="list-style-type: none"> Se reorganizan los equipos y roles para el ciclo 3.
	ESTRA3, PLAN3	<ul style="list-style-type: none"> Se produce la estrategia y los planes para el ciclo 3. Se evalúan riesgos.
12 03-07 Julio	REQ3	<ul style="list-style-type: none"> Se ajustan los requerimientos y el plan de pruebas del sistema para el ciclo 3
	DIS3	<ul style="list-style-type: none"> Se produce e inspecciona el diseño de alto nivel para el ciclo 3. Se actualiza el plan de pruebas de integración para el ciclo 3.
13 10-14 Julio	IMP3	<ul style="list-style-type: none"> Se implementa e inspecciona el ciclo 3, se produce el plan de pruebas unitarias.
	PRUEBAS3	<ul style="list-style-type: none"> Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 3.
14 17-21 Julio	PRUEBAS3	<ul style="list-style-type: none"> Se produce la documentación de usuario para el producto final. Se revisa y actualiza el manual de usuario con respecto a su utilidad y exactitud.
15 19-18 Agosto	PM3	<ul style="list-style-type: none"> Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 3. Se producen las evaluaciones de roles y de equipos para el ciclo 3. Se revisan los productos generados y el proceso utilizado. Se identifican las lecciones aprendidas y propuestas para la mejora del proceso.
Criterio de éxito		<ul style="list-style-type: none"> Haber terminado un producto o parte del mismo con documentación para el usuario. La carpeta del proyecto completa y actualizada. Evaluaciones de equipos y los reportes de ciclos documentados.

Tabla 12.3 Guión inicial de Desarrollo de TSPi del Instructor, DES

El guión que a continuación se muestra representa el plan de trabajo final.

Objetivo		Guía para el equipo de desarrollo de un producto de software
Criterio de entrada		<ul style="list-style-type: none"> El instructor guía y apoya a uno o más equipos de 5 estudiantes. Los estudiantes tienen el entrenamiento en PSP. El instructor tiene materiales, facilidades y recursos para apoyar a los equipos. El instructor ha guiado a los alumnos en la definición de los requerimientos generales del producto.
General		<p>El proceso de TSPi está diseñado para:</p> <ol style="list-style-type: none"> Desarrollar el producto de software de tamaño pequeño o mediano en 2 o 3 ciclos de desarrollo. Desarrollar un producto más pequeño en cada ciclo. Producir los elementos del producto, tales como documento de requerimientos, especificación de diseño, plan de pruebas, etc., en cada ciclo. <p>Siga los guiones que apliquen en su proyecto y modo de operación.</p>
Semana	Paso	Actividades

1 10-14 Abril	Revisión	<ul style="list-style-type: none"> • Introducción al curso y revisión de PSP. • Se leen los capítulos 1 y 2 del libro y el Apéndice A, o se prepara el documento de requerimientos validado por el cliente
2 24-28 Abril	LAN1	<ul style="list-style-type: none"> • Se revisan los objetivos del curso y se asignan los equipos de estudiantes y los roles. • Se lee el capítulo 3 del libro, el Apéndice B y uno de los capítulos del 11 al 15.
	ESTRA1	<ul style="list-style-type: none"> • Se produce el diseño conceptual, se establece la estrategia de desarrollo, se hacen estimaciones del tamaño y se evalúan riesgos. • Se lee el capítulo 4.
3 01-05 Mayo	PLAN1	<ul style="list-style-type: none"> • Se producen los planes del equipo y de los ingenieros para el ciclo 1. • Se lee el capítulo 5 y el Apéndice C.
4 08-12 Mayo	PLAN1	<ul style="list-style-type: none"> • Se producen los planes del equipo y de los ingenieros para el ciclo 1.
5 15-19 Mayo	REQ1	<ul style="list-style-type: none"> • Se definen e inspeccionan los requerimientos para el ciclo 1. • Se produce el plan de pruebas del sistema y el material de apoyo. • Se lee el capítulo 6 y las secciones de pruebas del capítulo 9.
6 22-26 Mayo	DIS1	<ul style="list-style-type: none"> • Se produce e inspecciona el diseño de alto nivel para el ciclo 1. • Se produce el plan de pruebas de integración y el material de apoyo. • Se lee el capítulo 7.
7 29 Mayo-02 Junio	IMP1	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 1. • Se produce el plan de pruebas unitarias y el material de apoyo. • Se lee el capítulo 8.
8 05-09 Junio	IMP1	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 1. • Se produce el plan de pruebas unitarias y el material de apoyo.
9 12-16 Junio	PRUEBAS1	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 1. • Se produce la documentación para el usuario para el ciclo 1. • Se lee el capítulo 9.
10 19-23 Junio	PM1	<ul style="list-style-type: none"> • Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 1. • Se producen las evaluaciones de roles y de equipos para el ciclo 1. • Se leen los capítulos 10, 16, 17 y 18.
11 26-30 Junio	LAN2	<ul style="list-style-type: none"> • Se reorganizan equipos y roles para el ciclo 2. • Se leen los capítulos del 11 al 15.
	ESTRA2 PLAN2	<ul style="list-style-type: none"> • Se produce la estrategia y el plan para el ciclo 2. • Se evalúan riesgos.
12 03-07 Julio	REQ2	<ul style="list-style-type: none"> • Se ajustan los requerimientos y el plan de pruebas para el ciclo 2.
	DIS2	<ul style="list-style-type: none"> • Se produce e inspecciona el diseño de alto nivel para el ciclo 2. • Se actualiza el plan de pruebas de integración para el ciclo 2.
13 10-14 Julio	IMP2	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 2, se produce el plan de pruebas unitarias.
	PRUEBAS2	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistema para el ciclo 2. • Se produce la documentación de usuario para el ciclo 2.
14 17-21 Julio	IMP2	<ul style="list-style-type: none"> • Se implementa e inspecciona el ciclo 2, se produce el plan de pruebas unitarias.
	PRUEBAS2	<ul style="list-style-type: none"> • Se aplican las pruebas unitarias, de integración y del sistemas para el ciclo 2. • Se produce la documentación de usuario para el ciclo 2.
15 14-18 Agosto	PM2	<ul style="list-style-type: none"> • Se dirige la evaluación postmortem y se escribe el reporte final del ciclo 2. • Se producen las evaluaciones de roles y de equipos para el ciclo 2. • Se revisan los productos generados y el proceso utilizado. • Se identifican las lecciones aprendidas y propuestas para la mejora del proceso.

Criterio de éxito	<ul style="list-style-type: none"> • Haber terminado un producto o parte del mismo con documentación para el usuario. • La carpeta del proyecto completa y actualizada. • Evaluaciones de equipos y los reportes de ciclos documentados.
-------------------	---

Tabla 12.4 Guión final de Desarrollo de TSPi del Instructor, DES

En un principio, los días viernes de cada semana, durante el horario de clases (de 10.00 a 11.00 a. m.) se reunía el Instructor con los líderes, quiénes le informaban acerca de los avances, asuntos especiales y problemas acontecidos en cada equipo. Posteriormente, por petición de los alumnos y para no perder horas de clase, tal reunión se llevó a cabo los días lunes, semanalmente, con un horario de 15:00 a 16:00 p.m., el cual era mayor o menor, dependiendo de las necesidades que surgieran en tales juntas.

El Instructor aconsejó a cada equipo realizar una reunión mínima por semana para establecer acuerdos y mantener un control de las actividades a desempeñar.

En un principio se acordó que la entrega de los productos a generar de cada fase debía realizarse durante la reunión con los líderes para la revisión pertinente por parte del Instructor. Posteriormente, la entrega de documentos se realizaba en las reuniones o fuera del horario de clase. Esta última medida se debió a los problemas que tuvieron los alumnos con el proceso y, por eso, surgió la necesidad de fijar reuniones extras con el fin de tener asesorías fuera del horario de clase. Cada reunión con los líderes era guiada mediante la agenda que realizaba el Instructor. Las actividades eran las siguientes:

1. Revisión de la minuta de la reunión anterior. Se realizaba un informe de cada reunión, en la cual el Instructor entregaba los documentos ya revisados o en ese momento se realizaba la revisión, si no eran extensos. Indicaba comentarios y sugerencias a los documentos entregados. Los equipos informaban acerca de los acontecimientos ocurridos. Se establecían acuerdos entre el Instructor y los alumnos y se mantenía un control sobre los documentos o temas pendientes por revisar de la reunión anterior.
2. Revisión de la agenda.
3. Revisión de los documentos pendientes de la reunión anterior. De tales documentos se realizaba un seguimiento en la minuta.
4. Asuntos especiales o problemas. Los alumnos externaban los problemas técnicos o de equipo ocurridos en la semana anterior, principalmente si tenían impacto en las actividades de cada uno. Se establecían acuerdos de reuniones u otros eventos y fechas de entrega de productos.
5. Acciones a tomar. Se refería a las posibles soluciones para resolver o prevenir problemas.
6. Revisión de las formas RESPL, RESCA, RESTM, CALENDARIO, TAREA, SEMANA, REGT y RAR de cada semana.
7. Definición de temas y fecha de la siguiente reunión. Se indicaban los productos a generar durante el transcurso de la semana y los temas relacionados con los mismos.

Al finalizar el segundo ciclo, las revisiones del producto final de cada equipo se extendió una semana más de lo previsto en el plan de trabajo, dando por concluido el semestre el día 21 de Agosto del 2000.

12.3. Reportes semanales de las actividades desempeñadas por los equipos y el Instructor

En esta sección se enumeran las semanas de trabajo del curso, el período que cada una abarcó, las minutas que se realizaron y las fechas de las mismas.

El día 10 de Abril de 2000 inició el curso de Ingeniería de Software en la Facultad de Ciencias de la UNAM; el cual fue impartido al sexto semestre de la Licenciatura en Ciencias de la Computación, por la Dra. Hanna Oktaba.

El objetivo consistió en aplicar el proceso TSPi, cuyo autor es Watts S. Humphrey, para desarrollar proyectos de software mediante el trabajo en equipo.

Después de haber formado los equipos entre los alumnos asistentes se establecieron ciertas necesidades a cumplir. En cada clase el Instructor explicaba los temas establecidos en el guión original, además de otros que se consideraban necesarios impartir para generar los productos de cada fase y tener una mejor comprensión del curso.

Se acordó entre el Instructor y los líderes realizar reuniones, y de esa manera tener un continuo seguimiento y control sobre los productos y el desempeño de cada equipo. Cabe mencionar que durante las reuniones era prescindible la presencia de cada uno de los líderes, si alguno de ellos no podía asistir era necesario que otro integrante del equipo lo sustituyera en las juntas.

En un principio, el Instructor planeó realizar tres ciclos para el desarrollo del proyecto, de acuerdo a lo establecido por TSPi. Debido a los constantes días festivos que originaban la suspensión de clases y a los problemas de los alumnos con el proceso, surgió la necesidad de realizar modificaciones de fechas al guión de actividades. Por tal razón los tres ciclos que se iban a desarrollar, se redujeron a dos.

A continuación se presentan las actividades realizadas cada semana a partir del inicio del curso, se indican los temas impartidos por el Instructor en cada clase, los productos generados por los equipos y los sucesos ocurridos en cada reunión con los líderes; en las cuales se mantenía un seguimiento sobre las actividades realizadas por cada equipo, el desempeño de sus integrantes y la generación de productos requeridos.

Semana no. 1 (10 a 16 de Abril de 2000)

En la primera semana se trataron temas relacionados con la Administración de Proyectos, el Diseño de TSPi y sus Principios, así como una introducción breve sobre las características básicas (establecimientos de métricas, evaluaciones de roles y del equipo, disciplina del proceso y consejos para resolver problemas en equipos).

Se formaron tres equipos entre los integrantes del grupo, asignando en cada uno los cinco roles definidos por TSPi. En dos de los equipos, un rol fue desempeñado por dos personas, designándose un responsable y un suplente. Estos roles fueron: Administrador de Planeación y Administrador de Desarrollo. También cada equipo definió el sistema a desarrollar de acuerdo al Apéndice A de la bibliografía de TSPi (en el Anexo B, en la Tabla B.1, llamada Integrantes de los equipos en los dos ciclos, puede consultar cómo se integraron los equipos y el sistema que cada uno decidió desarrollar).

Cabe mencionar que desde el inicio del curso, en la Universidad se mantenía cierta inestabilidad política, por lo tanto, existía el riesgo de no llevar a cabo el curso de acuerdo a la agenda establecida en el guión original.

Semana no. 2 (17 a 23 de Abril de 2000)

Esta semana abarcó el período vacacional correspondiente a Semana Santa, el Instructor indicó a los alumnos leer los Capítulos 1, 2, 3 y 4; el Apéndice A y el capítulo correspondiente a cada rol.

Semana no. 3 (24 a 30 de Abril de 2000)

Minuta no. 1 (24 de Abril de 2000)

Se realizó la primera reunión del Instructor con los líderes de cada equipo. En ella se revisaron los documentos Objetivos de los equipos e individuales y los Criterios de estrategia y la Estrategia.

De acuerdo a la revisión de los documentos anteriormente mencionados, se hicieron algunos comentarios y sugerencias sobre los mismos. También a los tres equipos se les sugirió establecer un estándar de documentos, esto consistía en indicar el nombre del documento y del equipo, la fecha de elaboración, el autor o redactor y el número total de páginas.

La siguiente tabla muestra las observaciones hechas sobre los documentos.

Documento	Equipo Dione	Equipo Octavos	Equipo Phobos D.S.
Objetivos del equipo e individuales	Documento aprobado. <i>Sugerencia:</i> Ampliar los objetivos de acuerdo a TSPi.	Documento aprobado.	Acordó entregar en la siguiente reunión.
Criterios de estrategia y la Estrategia	1. Corregir el criterio de estrategia. 2. Analizar las estimaciones referentes al número de líneas de código y a las horas asignadas.	1. Analizar de nuevo las líneas de código (la estimación fue menor de acuerdo al criterio del Instructor). 2. Unir en un solo documento los Criterios de estrategia con la Estrategia.	1. Modificar la descripción de funcionalidades. 2. Realizar estimaciones de totales en líneas de código y en horas. 3. Los criterios de estrategia no se entregaron.

Tabla 12.5 Documentos de la tercera semana

El Instructor propuso un horario para el uso de la Sala de Cómputo de la Facultad de Ciencias.

Semana no. 4 (01 a 07 de Mayo de 2000)

Debido a los días festivos, durante esta semana sólo hubo una clase. El Instructor explicó los temas referentes al Plan de Desarrollo y al Plan de Calidad.

Semana no. 5 (08 a 14 de Mayo de 2000)

Minuta no. 2 (08 de Mayo de 2000)

Durante la segunda reunión con los líderes se revisaron los pendientes de la reunión anterior. Uno de ellos consistía en verificar la existencia de las carpetas de cada equipo y los documentos producidos hasta el momento. Sólo faltaba organizarlas adecuadamente para poder anexar tales documentos.

En la Tabla 12.5, se hace un seguimiento de los documentos pendientes por corregir y entregar, así como de los que quedaron pendientes por revisar. En la tabla 12.6, se muestra un seguimiento de los documentos de la quinta semana y de otros más.

Documentos	Equipo Dione	Equipo Octavos	Equipo Phobos D. S.
Objetivos del equipo e individuales	Documento aprobado con las correcciones sugeridas.	Documento aprobado.	Documento no entregado.
Criterios de estrategia y Estrategia	Documento entregado para revisión.	Documento entregado para revisión.	Documento no entregado.
Formas de registro de tiempo REGT hasta la fecha presente	Entregadas para verificar que se mantenía un control de registro sobre el tiempo trabajado.	Entregadas para verificar que se mantenía un control de registro sobre el tiempo trabajado.	Entregadas para verificar que se mantenía un control de registro sobre el tiempo trabajado.

Documentos	Equipo Dione	Equipo Octavos	Equipo Phobos D. S.
Plan de riesgos	Documento entregado para revisión. <i>Sugerencia:</i> Aplicar el estándar de documentos.	Documento entregado para revisión.	Documento entregado para revisión. <i>Sugerencia:</i> Aplicar el estándar de documentos.
Plan de Configuración	No entregado. Se indicó realizarlo y definir la Mesa de Control de Configuración (MCC). <i>Sugerencia:</i> Realizar modificaciones en la lista de productos de configuración.	<i>Sugerencia:</i> Identificar la lista de productos de configuración.	<i>Sugerencia:</i> Identificar la lista de productos de configuración.

Tabla 12.6 Documentos de la quinta semana

Los tres equipos opinaron que existían problemas al desarrollar la fase de planeación y al utilizar las formas correspondientes (siendo éstas: RESPL, RESTM, CALENDARIO, TAREA, SEMANA y RESCA). Por lo tanto, se tomó la decisión de dedicar una semana más a esa fase y recibir asesorías extras, independientes del horario de clases.

Los alumnos identificaron como un riesgo el hecho de que el autor del libro se refiriera de manera continua a temas sobre Personal Software Process y A Discipline for Software Engineering, cuando ellos no habían tenido ningún curso sobre los mismos.

Al iniciar el curso, se les informó a los alumnos la dirección en Internet de la herramienta TSPi, se les recomendó verificar si existían ventajas al utilizarla, algunos de los integrantes encargados de desempeñar esta tarea encontraron ciertas dificultades.

Se estableció un horario para el uso de la Sala de Cómputo; fijándose para los días martes, a partir de las 14:00 a 16.00 horas, en la Sala 4.

Minuta no. 3 (12 de Mayo de 2000)

Durante esta semana se realizó la segunda reunión con líderes, se presenta un seguimiento de documentos en la siguiente tabla.

Documento	Equipo Dione	Equipo Octavos	Equipo Phobos D.S.
Objetivos del equipo e individuales	Documento entregado.	Documento entregado.	Documento entregado para revisión.
Criterios de estrategia y Estrategia	Documento entregado en la reunión anterior para revisión.	Documento entregado en la reunión anterior para revisión.	Documento entregado para revisión.
Plan de riesgos	Documento entregado para revisión con la sugerencia hecha en la reunión anterior.	Documento entregado en la reunión anterior para revisión.	Documento entregado para revisión con la sugerencia hecha en la reunión anterior.
Plan de configuración	Documento entregado para revisión.	Documento entregado para revisión.	Documento entregado para revisión.

Tabla 12.7 Documentos de la segunda reunión en la quinta semana

El equipo Octavos decidió utilizar algunas de las formas de la herramienta TSPi y realizar otras manualmente de acuerdo al libro. Este equipo continuó con el Plan de riesgos definido anteriormente. Entregaron la mayoría de las formas para revisión, tomando en cuenta REGT de esta semana. No entregaron SEMANA individual y por equipo de esta semana.

El equipo Dione decidió utilizar las formas de acuerdo al libro, por lo tanto, no usar la herramienta TSPI. También continuaron con el Plan de riesgos definido anteriormente. Entregaron la mayoría de las formas para revisión, tomando en cuenta REGT de esta semana. No entregaron SEMANA individual y por equipo de esta semana.

El equipo Phobos D. S. también decidió utilizar las formas de TSPI. Sí entregaron todas las formas para revisión, contando con SEMANA individual, SEMANA por equipo y REGT de esta semana.

Semana no. 6 (15 a 21 de Mayo de 2000)

Minuta no. 4 (19 de Mayo de 2000)

El Instructor se reunió con los equipos Dione y Phobos D. S. para señalarles ciertos comentarios sobre los documentos entregados referentes a la fase de Planeación y Estrategia. Los documentos anteriormente mencionados del equipo Phobos D. S. requerían de una revisión detallada por parte de sus integrantes, quienes tuvieron que realizar de nuevo toda la documentación referente a esa fase, y por lo tanto, se atrasaron una semana.

Los temas de las clases en esta semana se refirieron a la fase de Requerimientos, al Plan de Pruebas del Sistema y a las inspecciones. Se explicó a los alumnos cómo desarrollar el contenido de los documentos ERS y del Plan de Pruebas del Sistema, detallando las diferentes estrategias que se pueden utilizar y el método para realizar las inspecciones.

El Instructor verificó que entre los integrantes de cada equipo realmente se hubiesen realizado revisiones personales y en equipo de los documentos desarrollados. Para ello recomendó registrar cada revisión en las formas REGD y mantener un seguimiento de control de tiempo mediante la forma REGT.

Los equipos realizaron una primera versión de ERS y del Plan de Pruebas del Sistema.

La siguiente tabla muestra un seguimiento sobre los documentos desarrollados en esta semana.

Documento	Equipo Dione	Equipo Octavos	Equipo Phobos D.S.
<i>Especificación de Requerimientos de Software (ERS)</i>	Documento inspeccionado por el equipo.	Documento desarrollado, pero no ha sido inspeccionado por el equipo.	Documento desarrollado, pero no ha sido inspeccionado por el equipo.
<i>Plan de Pruebas del Sistema (PPS)</i>	Documento desarrollado, pero no ha sido inspeccionado por el equipo.	Documento parcialmente terminado.	Documento no desarrollado, sólo existe un bosquejo del mismo.

Tabla 12.8 Documentos de la sexta semana

Para la siguiente semana se definieron los acuerdos que a continuación se mencionan:

1. Entregar ERS y PPS con las inspecciones realizadas por cada equipo.
2. Entregar las formas REPINS, REGD, REGT, CAMC, REPC, RESPL, RESCA, RESTM, SEMANA y CALENDARIO.

Como las reuniones semanales entre el Instructor y los líderes se realizaban durante el horario de clases, los días viernes de cada semana, los alumnos solicitaron establecer otra fecha, la cual les permitiera incrementar los conocimientos sobre TSPI, asistiendo a más horas de clases y utilizando las reuniones para tratar todos aquellos asuntos concernientes a los equipos y a las fases del proceso. A partir del 29 de Mayo de 2000 los días lunes de cada semana se realizaban tales reuniones, con un horario de 15.00 a 16.00 p.m.

Semana no. 7 (22 a 28 de Mayo de 2000)

Los temas de las clases en esta semana se refirieron a la fase de Diseño de Alto Nivel (DAN), UML¹ y Tarjetas CRC². El Instructor consideró importante explicar los dos últimos temas para generar los productos necesarios en esta fase, los cuales fueron los diagramas de clases y de interacción.

Como se acordó durante la semana anterior, los alumnos entregaron al Instructor los documentos ERS y PPS para revisión, quién a su vez se reunió con cada equipo para señalar algunas observaciones.

Semana no. 8 (29 de Mayo a 04 de Junio)**Minuta no. 5 (29 de Mayo)**

En esta semana el Instructor explicó algunos antecedentes sobre la Ingeniería de Software y temas relacionados con el diseño, esos fueron: Arquitectura por capas, El Patrón MVC (Model View Controller) y Patrones de Diseño (Design Patterns). Indicó a los equipos utilizar la arquitectura por capas.

Con respecto a TSPi, el tema fue la fase de Implementación y el guión correspondiente. Se explicaron las actividades de tal guión, el cual establece una planeación de las tareas necesarias utilizando las formas RESPL y RESCA. También incluyó el desarrollo del diseño detallado (DBN). En esta parte el Instructor explicó cada uno de los componentes a realizar. Otra actividad involucró la realización del Plan de Pruebas Unitarias y el proceso para realizar la inspección debida. También se pidió a cada equipo establecer estándares de codificación, los cuales tenían que ser aplicados por cada una de las personas responsables al momento de desarrollar el código fuente. Se explicó a los alumnos el proceso para inspeccionar y revisar el código, corregirlo y compilarlo, utilizando la forma necesaria en la cual debieran quedar registradas tales actividades. Por último, el guión se refirió a la revisión de calidad de los componentes.

También durante esta semana se explicó el tema Administración de Configuración de Software.

El día lunes se realizó la primera reunión con líderes en el nuevo horario. Durante la cual sólo el equipo Dione entregó las formas REPINS, REGT, REGD, RAR, CAMC, CALENDARIO, RESPL, RESCA, RESTM, TAREA y SEMANA.

Cada equipo entregó la primera propuesta de diseño en el documento EDS, el cual incluía diseño de alto nivel, diagramas de clase, diagramas de interacción y las inspecciones correspondientes, así como también el Plan de Pruebas de Integración.

Durante el transcurso de esta semana el Instructor se reunió con los integrantes de cada equipo haciendo las siguientes observaciones sobre ERS y PPS:

Documentos	Equipo Dione	Equipo Octavos	Equipo Phobos D.S.
Especificación de Requerimientos de Software (ERS) y Plan de Pruebas del Sistema (PPS)	Documentos aprobados, los cuales fueron bastante claros y precisos.	Documentos aprobados con pequeñas correcciones.	Documentos aprobados con pequeñas correcciones.

Tabla 12.9 Documentos de la octava semana

El Instructor se reunió con cada equipo para comentar ciertas correcciones sobre los documentos de diseño. Hasta esta fecha no surgían problemas entre los equipos de acuerdo a lo que cada uno reportaba. Se acordó entregar todas las formas al finalizar el primer ciclo.

¹ UML (Unified Modeling Language), es un lenguaje de modelado gráfico para visualizar, especificar, construir y documentar los entregables de un sistema de software.

² Tarjetas CRC (Class Responsibility Collaboration), representan responsabilidades (funciones) de una clase e identifican la relación con otra(s) clase(s), para el desempeño de las mismas.

Semana no. 9 (05 a 11 de Junio)

Minuta no. 6 (05 de Junio)

En esta semana durante las clases se explicaron los temas referentes a los principios de programación en pareja y las reglas para el estándar de codificación (de acuerdo a PSP). Cada uno de los equipos explicaron la manera en la cual estuvieron desarrollando los algoritmos. Los equipos Phobos y Octavos se refirieron al conteo de líneas lógicas y a la complejidad de McCabe. El equipo Dione, se refirió a la manera de llevar el conteo de las líneas de código añadidas, eliminadas o modificadas entre dos versiones de programas.

El Instructor aconsejó que al desarrollar los diferentes productos del diseño detallado, no se incluyeran variables auxiliares como atributos de clase, no se utilizarán valores de una clase como parámetros de métodos y que la inicialización de objetos se hiciera mediante constructores con parámetros.

Durante la reunión con los líderes se hicieron comentarios sobre los documentos revisados. A continuación se muestra un seguimiento de los mismos:

Con el equipo Dione se aclararon ciertos detalles sobre DAN. Se acordó que en el transcurso de esta semana entregarían tal documento con las correcciones pertinentes, así como DBN para la revisión del Instructor (a este documento ya se le habían realizado las revisiones personales y por equipo).

Con el equipo Phobos D. S. se acordó que entregarían para el día 06 de Junio los siguientes documentos: DAN, Plan de Pruebas del Sistema y el avance sobre DBN (los tres documentos estuvieron en revisión personal durante estas fechas).

El equipo Octavos entregó DAN. Tenían un avance sobre los productos de DBN, a los cuales ya se les estaban aplicando las revisiones personales.

Se comunicó al Instructor un problema importante que estaba afectando el desempeño de un equipo; tres de los integrantes trabajaban en la tarde, dado que acordaron reunirse el fin de semana pasado no pudieron entregar el trabajo correspondiente, puesto que no sabían cómo realizarlo. Esta circunstancia afectó los acuerdos establecidos entre los integrantes asistentes y perjudicó la consistencia del trabajo que se debía desarrollar.

Para solucionar tal problema, el Instructor recomendó lo siguiente: que se trabajaran las funcionalidades de quienes sí estaban realizando sus tareas, que sólo eran tres personas; que se reorganizara el equipo para el siguiente ciclo y también, que se hiciera una reunión entre el equipo para encontrar una solución adecuada.

Otro impedimento durante la fase de implementación del software era el que ocasionaba la Sala de Cómputo. No se permitía que los alumnos trabajaran en equipo, es decir, solamente una persona podía usar una computadora. Otro problema eran las diversas plataformas de Unix en las estaciones de trabajo, porque no tenían el mismo compilador.

Semana no. 10 (12 a 18 de Junio)

Durante esta semana no se realizó la reunión con líderes. En clases se explicaron los diferentes ciclos de vida que se pueden aplicar al momento de desarrollar sistemas, mencionando entre ellos los modelos de Cascada, Espiral y Transformación.

El Instructor explicó cómo realizar el Manual de usuario, desarrollando cada sección de la tabla de contenido especificada; y cómo realizar el proceso de documentación de tal manual.

Los integrantes de cada equipo realizaron el documento DBN del módulo que les correspondía y el código respectivo, al cual se aplicaron las pruebas unitarias de caja transparente (donde se aconsejó que los parámetros que se utilizarán fueran muy concretos).

El Instructor se reunió con cada equipo para detallar ciertas observaciones sobre DBN y el código. Después de corregir los defectos se entregó cada módulo al ACP. Posteriormente, se integraron como línea base a la MCC. Se aplicaron las Pruebas de Integración (definidas en el plan que lleva el mismo nombre), registrando los defectos encontrados en la forma REGD. Y por último, se aplicó la prueba del sistema mediante la estrategia establecida en el Plan de Pruebas del Sistema.

Al desarrollar el código, algunos alumnos tuvieron problemas de programación en C++ porque durante un año no estuvieron practicando, debido a la suspensión de actividades en la Universidad.

Todavía en esta semana persistía el problema referente a la Sala de Cómputo de la Facultad, donde sólo se permitía que un alumno trabajara en una estación.

Semana no. 11 (19 a 25 de Junio)

Minuta no. 7 (19 de Junio)

Durante esta semana el Instructor explicó la fase de Postmortem para que los equipos entregaran el reporte correspondiente al primer ciclo del proyecto.

Durante la reunión con líderes, sólo estuvo presente el equipo Dione. El equipo Octavos avisó que no podría asistir y el equipo Phobos D. S. no se comunicó con el Instructor.

El equipo que asistió indicó que tuvieron problemas al momento de codificar por el tiempo que se mantuvieron sin clases normales, pero se apoyaron en material bibliográfico para solucionar tal problema.

En cuanto a los documentos por revisar en la prueba de aceptación, comentaron que sí realizaron pruebas unitarias, el documento ERS se aprobó para la versión inicial, a PPS no se le hicieron cambios y a DAN sí, los cuales ya se encontraban documentados en la forma debida (CAMC). Posteriormente, se aprobó la siguiente versión de DAN. No se realizaron modificaciones al Plan de Pruebas de Integración. Para esta fecha cada integrante del equipo tenía la última versión del código fuente.

El día 20 de junio de 2000 se realizó la primera revisión (también llamada prueba de aceptación por parte del cliente) a los sistemas y a la documentación, consistiendo en los documentos siguientes: ERS, DAN, DBN, Planes de Pruebas del Sistema, código impreso y Manual de usuario. En esa revisión los equipos Phobos D.S. y Dione desarrollaron un sistema que generalmente cumplió con las funciones especificadas en el Plan de Estrategia del primer ciclo. Aunque existieron algunos errores, a continuación se mencionan algunos de ellos:

Equipo Phobos D. S.:

Error: Inconsistencia en los mensajes y en la acción posterior del sistema.

Acción: Al desplegar un mensaje indicando "*Oprima la tecla [Enter] para continuar*", y al oprimir cualquier otra tecla también continuaba la acción, eso significó que no se encontraba validada para la tecla Enter solamente.

Equipo Dione:

Error: Carencia de claridad al momento de redactar el Manual de usuario. El ejemplo que se presenta está escrito como se redactó en ese documento.

Acción:

"De los cinco archivos que copió, compile los tres archivos siguientes:

pantalla.cpp loc.cpp y main.cpp

Con la siguiente instrucción:

Si utiliza el editor de emacs.

M -x compile

g++ pantalla.cpp loc.cpp main.cpp -o <nombre del archivo ejecutable>"

Equipo Octavos:

En cuanto al sistema del equipo Octavos, los problemas detectados en la revisión fueron mayores. Debido a la causa que se mencionó anteriormente (en la semana no. 9).

Algunos integrantes tuvieron que desarrollar módulos de otras personas. El documento ERS no contenía nombres claros en los requerimientos, no se identificaba el código de acuerdo a los casos de prueba, había inconsistencia en algunos nombres del DBN. Lo que se había especificado en los documentos no era consistente con el funcionamiento del sistema.

Durante esta semana se realizaron evaluaciones sobre las formas utilizadas, el proceso, inspecciones y revisiones, y pruebas. Utilizando los datos de las formas RESPL y RESCA de cada equipo se evaluó la planeación, calidad y defectos, tanto planeados como reales.

Se analizó la utilidad de las formas aplicadas y se indicaron los documentos en los cuales se utilizaron. De acuerdo con la opinión de los alumnos se identificaron formas poco útiles o redundantes. El Instructor les aconsejó utilizar la forma PMP, a través de la cual se permite especificar mejoras que pudieran ser útiles en el siguiente ciclo.

También los equipos analizaron el proceso, las inspecciones y revisiones (los análisis mencionados los puede consultar en la sección de Anexo B).

En cuanto a las pruebas aplicadas, los equipos no tuvieron problemas en la comprensión y aplicación de los tres tipos de pruebas (unitarias, integración y del sistema). Los equipos Dione y Octavos se auxiliaron con el depurador del compilador de C para aplicar las pruebas unitarias.

El equipo Octavos abandonó la herramienta TSPi porque no permitía registrar debidamente ciertos datos, los eliminaba. Por lo tanto, continuaron trabajando por completo con las formas manuales.

Mediante la aplicación de la forma EEC los alumnos evaluaron el proceso TSPi y los roles (lo puede consultar en la sección 12.6 Evaluación de los equipos).

El Instructor sugirió que se incluyera en el segundo ciclo las funcionalidades que se habían asignado al segundo y tercer ciclo, durante la fase de Estrategia del primer ciclo. Recomendó a los alumnos la necesidad de intercambiar roles, para adquirir experiencia y conocimiento en las responsabilidades de roles diferentes.

Aún persistían los problemas en la Sala de Cómputo.

Semana no. 12 (26 de Junio a 02 de Julio)

Minuta no. 8 (26 de Junio)

A partir de esta semana se inició el segundo ciclo, y con ello se explicaron las modificaciones en cada fase, iniciando con Lanzamiento. Se discutieron los problemas y las causas del ciclo anterior, para lo cual los equipos propusieron mejoras (las cuales se describen en la sección de Anexo B, en la Tabla B.2, nombrada Problemas y Mejoras durante el primer ciclo).

Otro tema de clase se refirió a la fase de Estrategia. El Instructor propuso que para el Plan de Administración de Configuración se generará la Tabla de Identificación de productos y la Tabla de Seguimiento o Rastreo. También se solicitó a los alumnos modificar el Procedimiento de Control de Cambios de acuerdo a la experiencia del primer ciclo. Otra de las clases se refirió al Capítulo 16, cuyo tema es Cómo conducirse en un equipo de trabajo. También en esta semana se trató la fase de Planeación. Los tres equipos opinaron que un problema importante fue la carencia de tiempo para realizar las actividades debidamente y obtener un producto de mejor calidad. Hicieron los siguientes comentarios sobre las formas utilizadas:

El equipo Dione comentó: *"Al principio surgieron problemas al momento de llenar las formas"*.

El equipo Octavos comentó: *"Algunas formas son redundantes (refiriéndose a CALENDARIO y TAREA)"*.

Se efectuó la reunión con líderes, en la cual sólo asistieron los equipos Dione y Octavos, quienes entregaron el reporte Postmortem correspondiente al primer ciclo. Los integrantes del equipo Phobos D. S. avisaron que no podrían estar presentes en la reunión, pero sí entregaron tal reporte.

El Instructor recomendó que las siguientes sugerencias a los documentos se tomarán en cuenta para el segundo ciclo.

Al equipo Dione: que los requerimientos quedaran relacionados con el DAN; que en el Plan de Pruebas de Integración se indicara que clases se integrarían entre si, en qué orden y mediante cual interfaz se realizaría la prueba; que se explicara cada algoritmo y que el código estuviera documentado de acuerdo al DBN y que se redujera la complejidad de la función Main.

Al equipo Octavos: que en ERS asignaran a cada requerimiento nombres más precisos e identificaran las funcionalidades mediante etiquetas enumeradas; que existiera mayor consistencia en los documentos, esto significa, que lo que se especificaba en cada documento estuviera de acuerdo con el funcionamiento del sistema; que se acordaran las entradas y salidas en el Plan de Pruebas del Sistema para que concordara con lo especificado en ERS; que se identificaran claramente los diagramas en el DAN; que hubiese más consistencia en algunos nombres del DBN y que se identificara el código de acuerdo a los casos de pruebas y al plan de pruebas de integración.

A partir de esta fecha los documentos ya realizados ingresaron a la MCC.

Hasta la fecha persistían los problemas en la Sala de Cómputo, para lo cual se decidió hablar directamente con la persona responsable de tal asunto.

Semana no. 13 (03 a 09 de Julio)

Minuta no. 9 (03 de Julio)

Durante esta semana el Instructor indicó los procedimientos necesarios para generar los productos correspondientes a las fases de Requerimiento y Diseño del segundo ciclo. De nuevo se abarcó el tema referente a los objetivos de las inspecciones. También se explicó a los alumnos sobre el DAN, haciendo referencia al Lenguaje de Modelado Unificado (UML) y al Proceso de Desarrollo de Software Unificado, en este último abarcando el tema de los estereotipos. Otra de las clases consistió en el Código de Ética de Ingeniería de Software.

Los alumnos asistieron a la conferencia denominada Métricas de McCabe para aseguramiento de Calidad y Pruebas. Este tema les permitió ampliar más los conocimientos sobre la complejidad ciclomática de McCabe. Debido a que el sistema Analizador de Programas requería dicho cálculo, principalmente, sirvió de apoyo a los equipos Phobos D. S. y Octavos.

Durante la reunión con los líderes se indicaron los nuevos roles de cada equipo. El equipo Phobos D. S. no realizó cambio alguno. El equipo Octavos intercambió un rol, el responsable de AP en el ciclo anterior adquirió el rol de suplente en el segundo ciclo, y quién tenía ese rol anteriormente, adquirió el rol de responsable. En el equipo Dione se intercambiaron por completo los roles (indicado en la sección de Anexo B, en la Tabla B.1, nombrada Integrantes de los equipos en los dos ciclos).

En esta reunión sólo estuvieron presentes los equipos Dione y Phobos D. S., el equipo Octavos avisó que no iba a poder asistir, pero sí entregaron los documentos requeridos antes de la junta.

El Instructor realizó los siguientes comentarios sobre los documentos entregados en esta reunión.

Equipo Phobos: no entregaron los objetivos y la nueva asignación de roles; entregaron la estrategia con sus estimaciones; entregaron la Tabla de Identificación de Productos y la Tabla de Seguimiento, aunque faltaron algunos datos; entregaron las formas RESPL y TAREA, faltaron RESTM y RESCA; decidieron utilizar las formas CALENDARIO ó SEMANA y no entregaron la forma RAR, para la especificación de riesgos.

Equipo Dione: entregaron la nueva asignación de roles, los objetivos y la estrategia; entregaron la Tabla de Identificación de Productos y la Tabla de Seguimiento, entregaron las formas RESTM, TAREA (por equipo e individual), CALENDARIO (por equipo, individual y por semana) utilizadas a partir del segundo ciclo; entregaron la especificación de riesgos, en la forma RAR y les faltó terminar de completar los datos de las formas RESPL y RESCA.

Los comentarios referentes al equipo Octavos no se registraron puesto que no asistieron a la junta y entregaron anteriormente los documentos al Instructor.

Semana no. 14 (10 a 16 de Julio)

Minuta no. 10 (10 de Julio)

Durante esta semana el Instructor explicó el guión de la fase de Implementación para el segundo ciclo. También se dedicó tiempo al tema de Patrones, el Instructor explicó el patrón Modelo Vista Controlador (MVC). Otro tema de gran interés fue el de CMM³, del cual se explicaron los niveles 1 y 2.

En la reunión semanal con los líderes, solamente asistieron los equipos Dione y Octavos, entregaron los siguientes documentos:

El equipo Dione entregó ERS y Plan de Pruebas del Sistema con sus respectivas inspecciones; las formas CALENDARIO (por equipo e individual) y TAREA (individual); DAN y Plan de Pruebas de Integración con sus respectivas inspecciones y las formas RESPL y RESCA de la semana no.13. Las formas RESPL y RESCA de esta semana no se entregaron porque les faltaba actualizar algunos datos de tiempo y tamaño.

El equipo Octavos entregó ERS y Plan de Pruebas del Sistema con sus respectivas inspecciones; DAN y Plan de Pruebas de Integración con sus respectivas inspecciones.

El equipo acordó entregar las formas TAREA, RESPL y RESCA pendientes, y también se comprometieron a entregar tales formas con los datos de esta semana.

No se realizó un seguimiento de los documentos del equipo Phobos D. S. porque no asistió.

Los alumnos reconocieron como alto riesgo la posibilidad de toparse con cierta dificultad para terminar la implementación de cada sistema, porque el período que abarcó del 10 al 23 de Julio del presente año fue destinado para la realización de exámenes.

Semana no. 15 (17 a 23 de Julio)

Minuta no. 11 (10 de Julio)

Durante esta semana los temas de clase trataron sobre las áreas claves de CMM niveles 2 y 3, llamadas también KPA's⁴.

Durante la reunión semanal con los líderes, los equipos realizaron los siguientes comentarios acerca del panorama de trabajo para su equipo:

- El equipo Dione no contempló como problema entregar el producto terminado al regresar de vacaciones.
- El equipo Octavos planeó terminar la implementación individual y en vacaciones realizar la integración completa del sistema y aplicar las pruebas requeridas.
- El equipo Phobos, D.S. comentó que se estaban atrasando en las actividades de esta materia por las demás actividades escolares.

A continuación se brinda un seguimiento de los documentos que se acordaron entregar durante esta semana.

- El equipo Dione entregó las formas CALENDARIO y TAREA actualizadas con datos de esta semana; DBN y las pruebas unitarias correspondientes.
- El equipo Octavos entregó el nombre del sistema, el cual se llamó JALSY; DBN con cuatro nuevas funcionalidades y las pruebas unitarias correspondientes. Aún les faltaba actualizar las formas RESPL, RESCA y TAREA.

³ Modelo de Madurez de Capacidad (Capability Maturity Model)

⁴ Key Process Area, Área Claves del Proceso, de cada nivel de CMM.

- El equipo Phobos D.S. entregó ERS, DAN, PPS; las formas RESPL, SEMANA y TAREA; pero faltó que actualizaran otras, porque no se tenían los datos de dos integrantes. El DBN y el Plan de Pruebas de Integración no se entregaron este día, pero el LE acordó entregarlos en el transcurso de la semana. No se entregó RESCA porque el ACP no había asistido desde el jueves 13 de Julio de 2000.

Debido al período de vacaciones, el Instructor y los integrantes de cada equipo acordaron reunirse el mes de Agosto para realizar la entrega y revisión de los sistemas, y dar por terminado el semestre. También se definieron los acuerdos siguientes: se planearon realizar dos reuniones el día 14 de Agosto de 2000, la primera sería durante el horario de clase y la segunda en el horario de reunión con líderes; se planeó realizar en Agosto el Postmortem del ciclo 2; la entrega de los sistemas se decidiría el día lunes 14 de Agosto; se acordó entregar las formas generales actualizadas de los ciclos primero y segundo; los documentos, las carpetas y el código fuente en la versión final.

Semana no. 16 (14 a 20 de Agosto)

Después de un período de vacaciones se reanudó el semestre. El día 14 de Agosto de 2000 el Instructor se reunió con los alumnos para reportar el estado de sus sistemas. A los equipos Dione y Phobos D. S. les faltaba realizar la integración de los módulos. Al equipo Octavos se les permitió que el sistema funcionará bajo el ambiente Windows.

Semana no. 17 (21 a 27 de Agosto)

A pesar de que esta semana no se contempló dentro del guión original, el Instructor permitió a los equipos que durante la misma pudieran finalizar sus actividades y efectuar las revisiones.

El 21 de Agosto se realizó la revisión al sistema del equipo Octavos, la cual resultó sin ningún defecto.

El día 22 del presente mes se realizó la revisión al equipo Phobos D. S., tal sistema resultó sin defectos, aunque el Instructor hizo algunas sugerencias para reestructurar el Manual del usuario.

El día 23 del presente mes se realizó el reporte de Postmortem correspondiente al segundo ciclo del proyecto. Se revisaron las formas RESPL y RESCA y se realizó la segunda evaluación de los roles.

Este día se revisó el sistema del equipo Dione, en el cual se encontró cierta inconsistencia en el mensaje de Bienvenida, también se encontró deficiencia en una de las funcionalidades correspondientes al contador, el cual no se actualizaba de acuerdo a los nuevos requerimientos del segundo ciclo.

Esta fue la última semana de trabajo en este curso, la cual concluye con el buen desarrollo de los sistemas, sin dejar de mencionar pequeñas anomalías, las cuales no entorpecieron tal criterio.

12.3.2. Conclusiones de los Antecedentes Académicos, Método de Trabajo y Reportes Semanales

La realización de los reportes semanales a partir del inicio del curso permitieron mantener control y seguimiento sobre las actividades desarrolladas por el Instructor y los alumnos.

Un aspecto que causó confusión en los alumnos fueron las referencias que hacía TSPi a temas de Personal Software Process y A Discipline for Software Engineering, puesto que el autor estableció como requisito que las personas que aplicaran TSPi, anteriormente deberían de haber aplicado y utilizado la bibliografía mencionada. Para solucionar tal requisito, el Instructor cubría los temas referentes cuando era necesario y proporcionaba la documentación requerida.

Para el desarrollo de software es necesario y básico establecer un plan de actividades desde el inicio de un proyecto. Por tal motivo, TSPi utiliza un guión de actividades en el cual se desglosa el

período de trabajo de un equipo en semanas, indicando en cada una todas las tareas a desarrollar y aquellos temas necesarios que apoyarán tales tareas.

Como se mencionó al inicio de esta sección, el desarrollo del proyecto se desglosó en tres ciclos. Debido al desconocimiento e inexperiencia de los alumnos sobre el proceso para el desarrollo de software, el Instructor se vio en la necesidad de reducir los ciclos de trabajo. El número de días festivos ocasionó que las clases se interrumpieran, y por consiguiente, el retraso de los alumnos en cuanto a los procedimientos para el desempeño de las actividades requeridas.

En la fase de Estrategia fue necesario realizar de nuevo las estimaciones en tamaño de los productos. Planeación fue la fase que causó mayor conflicto, debido a que los documentos requeridos debían realizarse en una semana, y el método para completar las formas no era fácilmente entendible. Se propuso a los alumnos utilizar la herramienta proporcionada por TSPi, algunos de ellos encontraron defectos en su uso, pero aún así decidieron combinar la generación automática de formas con la realización manual de las mismas; aunque más tarde decidieron no seguir utilizando la herramienta. Los equipos solicitaron ampliar el tiempo dedicado a la planeación a dos semanas; ese fue otro de los motivos para reducir el número de ciclos que se tenían planeados al inicio del curso. Uno de los equipos tuvo bastantes conflictos al realizar los documentos de esa fase, y por ello, se retrasaron aún más.

Al finalizar el primer ciclo, el Instructor solicitó a los alumnos su opinión sobre la utilidad de las formas. La mayoría de ellas se catalogaron útiles, y otras pocas no. Las de uso óptimo fueron CAMC, REPC, REPINS, REGD y RESTM. Con respecto a las formas TAREA, CALENDARIO y SEMANA, cada equipo era libre de decidir cual(es) utilizar porque son formas equivalentes entre sí. El tiempo de las actividades realizadas se indicaba en la forma REGT, pero los alumnos opinaron que sería mejor sólo registrar las actividades relevantes. Las formas concernientes a resúmenes de defectos, siendo esas, RESDI y RESDE, no se utilizaron. Otras formas que se consideraron redundantes fueron RESTDES y RESTAREA.

Para las fases posteriores (Requerimientos, Diseño, Implementación y Pruebas de Integración y del Sistema) los problemas fueron menores. Sin embargo no dejaron de darse ciertos conflictos en cuanto a la generación de los productos.

Como se mencionó anteriormente, la mayoría de los alumnos nunca antes habían utilizado algún proceso para desarrollar software. Fue conveniente que en las clases el Instructor impartiera algunos temas actuales sobre técnicas y procesos para generar los productos de las fases anteriormente mencionadas, aunque TSPi no los incluyera.

Otro aspecto básico para obtener una buena calidad de producto era la aplicación de inspecciones, revisiones y pruebas como lo especifica TSPi.

Durante la primera revisión de aceptación, dos de los equipos realizaron un producto generalmente aceptable, excepto con algunas inconsistencias. El otro equipo tuvo mayores dificultades, una de las causas fue no haber aplicado revisiones e inspecciones de la manera adecuada.

La práctica continua de cierta actividad permite adquirir mayor destreza y conocimiento sobre la misma. Al llevar a cabo el segundo ciclo, los tres equipos concluyeron que el desempeño de roles en cada fase fue más fácil que en el ciclo anterior. Ya tenían conocimientos y un panorama sobre las actividades y tareas que debían realizar, y los productos que debían generar. Pero lo más importante fue la experiencia adquirida durante el primer ciclo, la cual les permitió evitar defectos que anteriormente habían realizado.

En general, durante la revisión de aceptación del segundo ciclo, la calidad del sistema fue mejor calificada, el Instructor recalcó ciertas sugerencias importantes para que fueran tomadas en cuenta. Aunque uno de los equipos tuvo problemas en cierta funcionalidad, cabe mencionar que se mejoró bastante el desarrollo del proyecto en este último ciclo.

A pesar del número considerado de días festivos, la inestabilidad política en la Universidad, los problemas para aplicar TSPi, la reducción de tres ciclos a dos, los problemas internos de cada equipo y la necesidad de asignar tiempo a otras actividades, se concluye que la aplicación de TSPi se realizó de manera aceptable, gracias al empeño de los alumnos y bajo la dirección y gran apoyo de la Dra. Hanna Oktaba.

12.4. Aspectos Sociales, Técnicos y del Proceso de los equipos

Mediante la encuesta que se aplicó al grupo, las respuestas de cada pregunta se clasificaron en aspectos sociales, técnicos y del proceso.

Los aspectos sociales involucraron la opinión de los alumnos respecto a la interacción del trabajo en grupo. En los aspectos técnicos se tomaron en cuenta las opiniones sobre las fases del ciclo de vida de un proyecto, la necesidad de utilizar el libro de TSPi y las clases impartidas por el Instructor, así como los conocimientos adquiridos en las áreas de Ingeniería de Software (IS) y técnicas de documentación, cuyos temas fueron impartidos por el Instructor. Por último, se presentan los aspectos del proceso, los cuales se refieren a los antecedentes académicos de los alumnos, a las necesidades de impartir temas extras, se hace referencia a dificultades surgidas y las razones de ello, y por último, se presentan las sugerencias hechas por los alumnos, para aplicarlas en cursos posteriores.

12.4.1. Aspectos sociales

Los resultados que se obtienen en un equipo son de mayor calidad debido a que un grupo de personas unen un conjunto de conocimientos que posteriormente comparten entre sí; modifican ciertas formas de desempeño al momento de realizar determinadas actividades y logran una mayor eficiencia.

La interacción en equipo permitió a cada uno de sus integrantes adquirir conocimientos sobre la organización de equipos, conocer el beneficio que proporciona realizar un plan de trabajo y mantener control sobre cada una de las actividades; realizar un plan de horas de trabajo individualmente y por equipo; ser responsables para cumplir con compromisos en fechas y horarios establecidos; valorar la necesidad de cooperación por parte de cada integrante, compartir un horario similar y dedicar el tiempo necesario al trabajo en equipo. Puesto que cada integrante es una pieza fundamental, el incumplimiento de alguno les permitió darse cuenta que afectaba la productividad del equipo completo.

Personalmente aprendieron a colaborar entre sí, siendo más tolerantes en determinadas actitudes, sabiendo aceptar nuevas ideas y no tratando de imponer las propias; administrando y resolviendo los problemas que surgieron; aceptando críticas sobre el trabajo personal; aprendiendo a tratar a los compañeros de grupo en cuestión de trabajo, a conocer rasgos de su personalidad, sabiendo lo que les disgusta o les agrada, pero también aprendieron a conocerse más cada uno.

A la mayoría de los alumnos que se les aplicó la encuesta (seis) les agradó colaborar en equipo; una cantidad mínima (dos) opinó lo contrario, por los problemas que surgieron al colaborar con sus compañeros y a otra cantidad mínima (dos), a pesar de las dificultades, sí le gustaría repetir la experiencia

Por lo anteriormente mencionado, se deduce que el trabajo desarrollado por un conjunto de personas origina beneficios mayores comparado con el trabajo individual. Es importante mencionar que también se compartió y se conoció el entusiasmo y la disposición de algunas personas, lo cual fue de gran apoyo para obtener en conjunto un trabajo productivo, y fue un punto clave para seguir adelante pese a todos los problemas surgidos.

12.4.2. Aspectos técnicos

Anteriormente los alumnos de este curso no habían utilizado ningún proceso para desarrollar software, sin embargo, algunas personas ya habían aplicado algunas técnicas de requerimientos y diseño (precondiciones y postcondiciones).

La mayoría de los alumnos no tenía conocimientos sobre las fases desarrolladas en TSPi. Sin embargo, algunos de ellos ya habían desarrollado dos de esas fases (Implementación y Diseño). La siguiente gráfica representa lo descrito anteriormente.

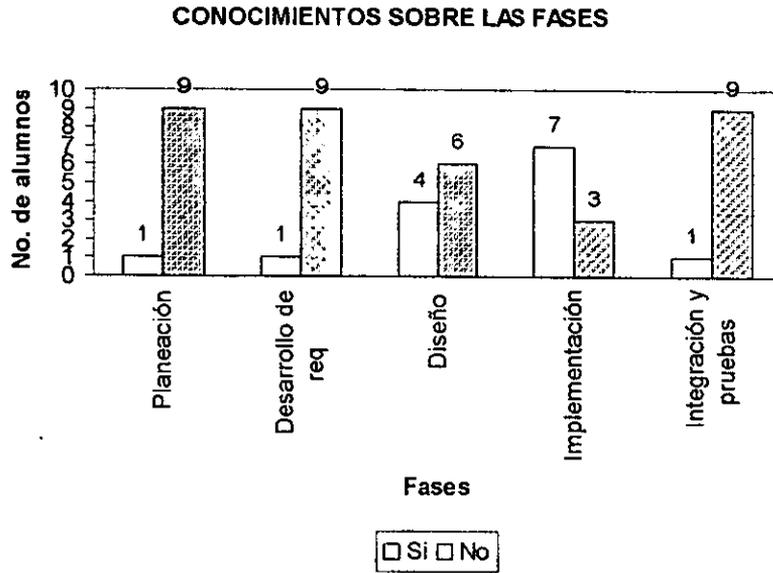


Figura 12.2 Representación del conocimiento de los alumnos sobre las fases

A pesar de que un número mínimo de estudiantes (dos) opinó que no era necesario tener conocimientos previos sobre las técnicas de modelado de requerimientos y diseño, y que era suficiente con las explicaciones que el instructor impartía en la clase, la mayoría de ellos (ocho) comentó que sí sería benéfico.

Para el segundo ciclo, los alumnos ya habían adquirido mayor destreza y tuvieron menor dificultad para desarrollar cada una de las actividades, debido a que ya se tenía la experiencia del primer ciclo y un conocimiento más amplio sobre las tareas que se debían realizar.

Durante el curso, principalmente en el primer ciclo, la mayoría de los alumnos se apoyaron de igual manera tanto en el libro de TSPi como en las clases impartidas por el instructor. A ellos no les causó mayor problema el nivel de inglés, aunque otros compañeros encontraron algunas dificultades en modos de expresión por parte del autor. Para un número menor de alumnos (dos) el nivel de inglés fue difícil, por tal razón utilizaron en un porcentaje menor el libro.

A través del curso fue necesario aplicar ciertas áreas de la Ingeniería de Software para desarrollar las tareas y actividades de cada fase. Esas áreas fueron métricas, técnicas de documentación, roles, fases, inspección y revisión y administración de configuración. La mayoría de ellas ya estaban implícitas en TSPi, pero otras no, por lo tanto, fue necesario que el instructor impartiera tales temas en clase.

A continuación se muestran dos gráficas, la primera brinda información acerca de la puntuación asignada por cada alumno sobre los conocimientos adquiridos en cada área (se estableció el rango desde 1 -más bajo- hasta 5 -más alto).

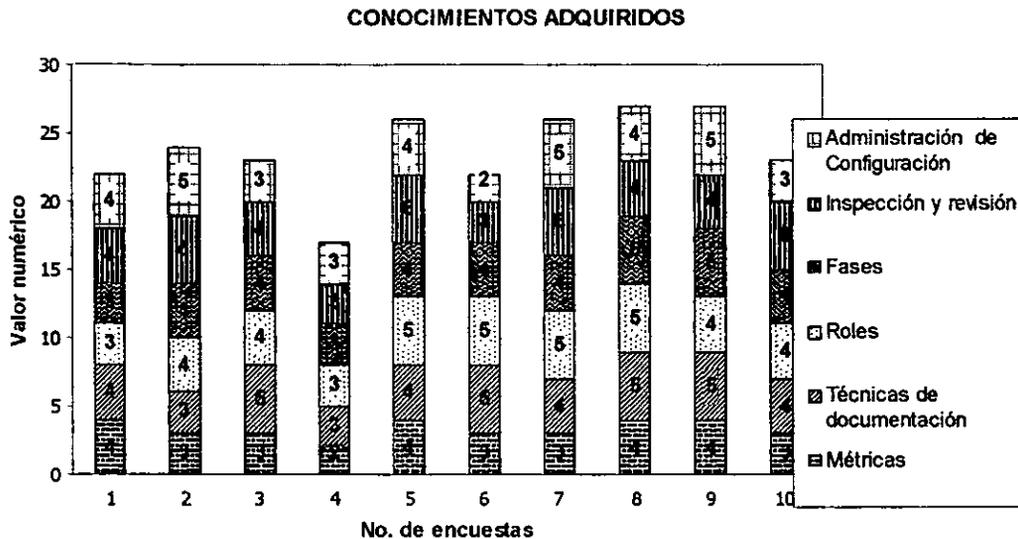


Figura 12.3 Concentrado numérico sobre conocimientos adquiridos en cada área

La puntuación máxima por alumno serían 30 puntos si hubiesen adquirido un alto rango de conocimientos en cada área. La puntuación mayor fue de 27 puntos, lo cual es bastante aceptable y la calificación menor fue de 17 puntos, la cual no se considera óptima.

La segunda gráfica muestra un resumen de evaluación total por área.

Las áreas de técnicas de documentación, roles, e inspección y revisión ocupan el nivel mayor. Posteriormente, administración de configuración se encuentra en tercer nivel y, por último, métricas.

La evaluación óptima para cada área hubiese sido de 60 puntos, porque se pidió a los alumnos que evaluaran cada una desde 1 (más bajo) hasta 5 (más alto). Por lo tanto, se deduce que en las áreas de técnicas de documentación, roles, inspección y revisión, y fases los alumnos adquirieron mayores conocimientos, posiblemente por la continuidad de su aplicación, pero eso no significa que las áreas restantes no requieran aplicarse de igual manera, puesto que su importancia es fundamental desde el inicio de un proyecto, en cada ciclo y hasta el final. Mediante la aplicación de métricas se puede lograr un producto de gran calidad, y con la administración de configuración es posible mantener un control riguroso sobre cada uno de los productos generados en cada fase.

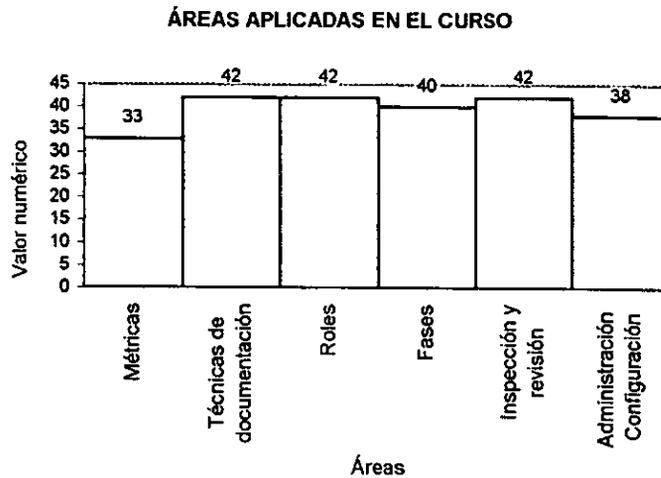


Figura 12.4 Áreas de Ingeniería de Software aplicadas en el curso

12.4.3. Aspectos del proceso

Los antecedentes académicos de los alumnos que cursaron la materia de Ingeniería de Software se representa en la Figura 12.1. Como se observará no existía una cercanía de igualdad entre quienes habían cursado las materias de la licenciatura en Ciencias de la Computación. Posiblemente, ello se debió a que los alumnos cursaron la materia, pero no la acreditaron; dado que las materias no se encontraban seriadas, por lo tanto, si un alumno no acreditaba determinada asignatura, podía cursar las demás sin ninguna restricción.

Al inicio del curso, antes de aplicar TSPi, el instructor debería tener conocimiento sobre los antecedentes académicos y la disposición de tiempo de los alumnos, debido a que es necesario realizar un plan de trabajo con un horario bastante disponible para la materia.

Dado que la gran parte de los alumnos no tenían conocimientos sobre las fases propuestas por Humphrey, opinaron que sería conveniente realizar una introducción previa sobre TSPi para conocer los conceptos manejados, facilitar el desempeño de cada actividad y realizar los tres ciclos sugeridos por el autor del libro; aunque también se presentó la alternativa de desarrollar dos ciclos. Se consideró factible realizar un curso introductorio que presupone Humphrey sobre PSP.

Respecto a las fases, se concluyó que Implementación (70%) es la que ha sido mayormente aplicada en cursos anteriores, posteriormente se encuentra la fase de diseño, con un porcentaje bajo (40%). Sobre las fases restantes, un número mínimo de alumnos tenían conocimientos sobre las mismas (10%) (ver Figura 12.2 Representación del conocimiento de los alumnos sobre las fases). Por ello, se consideró conveniente dedicar más tiempo a cada fase, especialmente a Planeación, Diseño e Implementación (considerando dos semanas para cada una) para aclarar dudas o imprevistos que no se hubieran tratado en clase.

Como se muestra en la gráfica anteriormente mencionada, la carencia de una base sólida respecto a las fases, la falta de experiencia y de conocimientos previos ocasionaron la necesidad de haberles dedicado mayor tiempo que lo planeado, así como haber tenido problemas al desarrollar ciertas actividades.

Principalmente, se encontraron más dificultades en Planeación, fase a partir de la cual se estructura un plan de actividades y de tiempo; al no comprender claramente el objetivo de cada actividad surgieron problemas y un retraso en cuanto a fechas asignadas. En ocasiones hubo temas que ya se habían comprendido, necesarios para generar ciertos productos, y los cuales fueron la base para generar otros más, al momento de generar los productos nuevos, los alumnos se cercioraban de que el tema no había sido claramente comprendido, de que los productos anteriores no habían sido generados correctamente y, por lo tanto, de que surgían problemas al momento de estar realizando los nuevos productos.

Durante el primer ciclo no se tenían conocimientos claros sobre los objetivos de cada fase, pero después de haberlo concluido y de haber iniciado el ciclo siguiente los alumnos ya pudieron comprender y comprobar la razón de desarrollo de cada actividad.

A pesar de que poco más de la mitad de los alumnos estuvieron en contacto con algún lenguaje de programación en los seis meses anteriores al curso, para algunos de ellos no fue así y eso ocasionó problemas al momento de realizar la implementación, porque encontraron dificultades en el desarrollo de algunas funcionalidades.

El desarrollo del curso fue afectado por las siguientes razones:

El tiempo: fue uno de los factores más importantes para realizar un buen trabajo en equipo, la necesidad de dedicar bastante tiempo en TSPi se debe a que el trabajo es realizado en conjunto por los integrantes de un equipo, quienes deben estar interactuando constantemente, ya sea para la toma de decisiones, para la generación de productos en cada fase y otras actividades más. Debido a que los alumnos tenían que cumplir con otras materias y algunos de ellos también dedicaban tiempo a otras actividades (laborales) el tiempo asignado a este curso no era suficiente.

Equipo de cómputo: en el laboratorio de cómputo no existía disponibilidad para su uso y el equipo no estaba en condiciones óptimas para el desarrollo de los sistemas.

Temas relacionados: en cada fase se debían generar determinados productos, para ello fue necesario que el instructor impartiera temas adicionales, por mencionar un ejemplo: las técnicas de análisis. Como la mayoría de los alumnos no tenían conocimientos en esa área, eso causó cierto retraso en cuanto a tiempo y generó problemas al momento de desarrollar, por primera vez, los documentos de requerimientos.

Colaboración en equipo: de nuevo la falta de disciplina estuvo presente entre los alumnos al momento de realizar trabajo en equipo, lo cual ocasionó un inadecuado plan de actividades para las reuniones del equipo.

Es importante indicar el siguiente comentario de uno de los alumnos de acuerdo a su visión del equipo al que perteneció: "*Un equipo en no muy buen estado*".

Otros: al iniciar el semestre, existieron demasiados días festivos combinándolos con la pausa necesaria de labores en la Universidad.

Idioma: se argumentó que el nivel de inglés también causó cierta dificultad (en grado menor). La respuesta a la pregunta ¿Tu nivel de inglés causó dificultad en asimilar el material del libro? se representa a continuación.

EVALUACIÓN DEL NIVEL DE INGLÉS

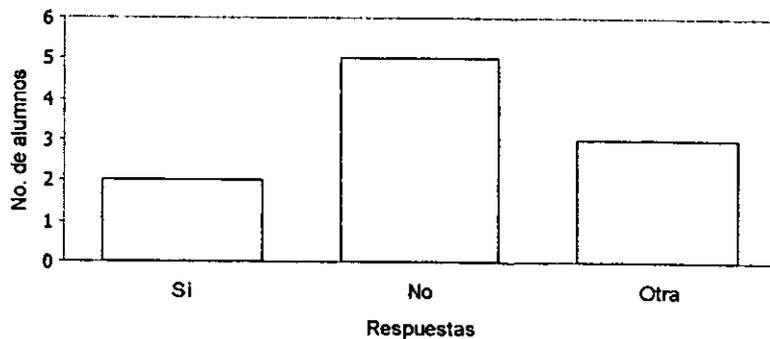


Figura 12.5 Evaluación del nivel de inglés para asimilar el material bibliográfico

La columna de la gráfica con la respuesta Otra, incluye los siguientes comentarios:

- a) "Un poco, principalmente por algunos modos de expresarse, y no tanto, por la sintaxis."
- b) "Algunas veces, pero no fue mucho problema."
- c) "Más o menos."

Al momento de aplicar TSPi y de obtener una comprensión más clara sobre este proceso, los alumnos se percataron de que sería conveniente realizar ciertas modificaciones.

A las formas utilizadas: la mayoría de los alumnos opinó que algunas eran redundantes (principalmente la forma SEMANA) y que lo mejor sería eliminarlas. Que el llenado absorbía demasiado tiempo y que les gustaría utilizar sólo aquellas que permitieran, de manera eficiente, recolectar información útil para el desarrollo del sistema.

A los ciclos: también se sugirió solamente aplicar dos ciclos y no tres.

A la documentación: una de las personas consideró que al documentar en exceso no se obtienen beneficios en el desarrollo de un sistema, por el lado contrario, otra de ellas comentó que por su parte no eliminaría nada al proceso.

Añadir información: se consideró necesario incrementar la información sobre el tema de liderazgo, por la gran necesidad que existía de forjar un buen ambiente de trabajo entre los integrantes de cada equipo.

12.5. Reporte Postmortem

Al finalizar los dos ciclos de trabajo, cada equipo realizó el reporte Postmortem. Mediante este documento se reflejaron los acontecimientos internos de cada equipo, respecto a la manera de trabajar, alcances y problemas durante los dos ciclos. Cada integrante evaluó por si mismo el propio rol desempeñado (de acuerdo a los indicados por TSPi) y el rol de ingeniero, el cual se desempeñaba al momento de realizar la implementación de cada sistema. En esta sección se presenta la información recabada mediante los reportes postmortem de cada equipo.

Comentarios y sugerencias para algunos roles

El rol Líder de Equipo intervino durante el proyecto en el ámbito administrativo y técnico. Debía tener comunicación directa con el equipo de trabajo que realizaba las actividades relacionadas con el desarrollo del sistema, en un aspecto técnico. Era responsable de la unión e integración del equipo; la toma de decisiones relacionadas con el grupo de ingenieros de desarrollo y de que cada actividad fuera realizada adecuadamente por el responsable. Por lo tanto, se aconsejó que un LE procure interactuar el mayor tiempo posible con los integrantes del equipo para conocer su forma de pensar, lo cual le permitirá tomar mejores decisiones y motivar a cada uno de ellos.

El Administrador de Planeación fue catalogado como uno de los más importantes, dado que su responsabilidad primordial era realizar una adecuada planeación mediante una organización efectiva de actividades en un tiempo preciso y a partir de los cuales fuera posible desarrollar un proyecto óptimo durante el ciclo de vida del mismo.

Durante el primer ciclo, se consideró que el aporte del rol Administrador de Apoyo fue mínimo; pero en el segundo ciclo el apoyo fue mayor debido a que se incrementó el desarrollo de funcionalidades en cada sistema.

Los integrantes de un equipo ya habían interactuado anteriormente en este tipo de trabajo. Cuando este es el caso, la integración es más fácil, puesto que ya se tienen conocimientos sobre el desempeño de los compañeros.

Organización de trabajo

Al inicio del curso el Instructor recomendó establecer fechas de reunión para discutir las actividades, fijar acuerdos y manejar asuntos pendientes. Uno de los equipos se anticipó a lo antes dispuesto, y para entonces ya había fijado tales fechas. El procedimiento de trabajo consistió en iniciar cada reunión revisando los puntos que se iban a tratar y los que faltaban por cumplir, que hubieran quedado pendientes en la reunión anterior. Cada reunión era dirigida por el integrante (o integrantes) responsable(s) de las actividades que en ese momento debían realizarse.

En otro equipo, durante el primer ciclo, se generaron problemas de organización y comunicación, al no haber establecido un horario de trabajo, lo cual originó problemas y una falta de organización al tener que realizar las formas y la documentación requerida. A pesar de ello, la buena participación y colaboración adecuada y oportuna de cada integrante les permitió entregar un producto de buena calidad durante el primer ciclo.

En dos de los equipos, un rol era desempeñado por dos personas (responsable y suplente). En uno de esos equipos se asignó suplente de Administrador de Planeación, en general, el trabajo desempeñado fue de buena calidad, pero algunos de sus integrantes trabajaban, entonces no se les hizo una adecuada planeación de trabajo de acuerdo al tiempo disponible, lo cual ocasionó problemas entre los integrantes. En el otro equipo se asignó suplente de Administrador de Apoyo, no habiendo mayores problemas entre las integrantes.

Como se mencionó anteriormente, los equipos formados fueron tres. En dos de ellos surgieron ciertos problemas que a continuación se mencionarán, pero en el otro equipo los obstáculos fueron menores, siendo de aspecto técnico. Hubo realmente una buena organización y comunicación, probablemente por la compatibilidad de horarios entre sus integrantes.

Problemas externos a TSPi

Principalmente, la carencia de tiempo fue una causa que afectó a dos de los equipos. A pesar de que algunos alumnos ya habían experimentado este tipo de trabajo, los problemas no fueron excluidos. Algunas personas tenían que distribuir su tiempo entre la Universidad y el trabajo externo, aparte de las demás materias que estaban cursando. Debido a que algunos integrantes no pudieron cumplir a tiempo con las tareas asignadas, otras personas tenía que realizarlas. Por lo tanto, así provocaban un retraso a los demás compañeros en cuanto a la generación de documentos.

Otra causa detectada fue la falta de puntualidad, la cual fue disminuyendo, aunque no se logró un avance perfecto. La ausencia de algunas personas en las reuniones ocasionaba que los productos a generar no se realizaran tomando en cuenta los acuerdos establecidos.

Durante el primer ciclo los problemas de organización y comunicación en uno de los equipos se trataron hasta el final del ciclo. Por lo contrario, en el otro equipo se trataban de resolver en el mismo momento.

Problemas en las fases

En general, los documentos desarrollados en las fases de Lanzamiento y Estrategia estuvieron correctos. Aunque en uno de los equipos las asignaciones de trabajo establecidas en la estrategia fueron incorrectas. Las dificultades mayores empezaron durante la fase de Planeación. El poco tiempo que se dedicó originó que la comprensión de conceptos no fuera clara, lo cual dificultó la realización del plan de trabajo del equipo y de cada integrante. Al principio, fue complicado para el Administrador de Planeación generar el plan de trabajo completo, porque no se había entendido claramente la manera en que debía realizarse. También el Administrador de Desarrollo encontró ciertas dificultades al tener que decidir sobre las actividades que debía desarrollar cada integrante. En uno de los equipos, en ocasiones sucedió que cada administrador había establecido un plan de trabajo diferente al que se le había asignado (tal consecuencia se debió a que el AD no tenía un horario compatible con los demás integrantes). Eso ocasionaba, que el equipo tuviera que discutir la manera en que debían realizarse las actividades, el tiempo que debería asignarse a cada tarea y el ciclo en el cual se desarrollaría.

En otro equipo, donde el rol de Administrador de Planeación era desempeñado por el responsable y el suplente, el responsable no realizó un adecuado plan de trabajo para las personas que laboraban tiempo extra independiente de las actividades universitarias, y eso originó problemas de desempeño. Fue el suplente quien se responsabilizó del llenado de las formas RESPL, RESCA y TAREA.

Otra de las causas que originaron problemas en el inicio de aprendizaje de este proceso, fue la versión beta de la herramienta TSPI, lo cual provocó dificultades en la generación y llenado de formas, así como en la consolidación de datos; añadiendo a ello la falta de comunicación entre los integrantes.

Las vacaciones de Semana Santa y las diferentes plataformas de trabajo que había en el laboratorio interrumpieron las fases de Diseño e Implementación, lo cual provocó retraso en las fechas de entrega. Durante la fase de Implementación surgieron varias dificultades. Debido a causas externas, antes de que iniciara el curso, los alumnos estuvieron un tiempo considerable sin asistir a la Universidad, eso originó olvido en la programación; otra dificultad fue la complejidad para realizar la implementación de algunas funcionalidades, como por ejemplo, calcular la intensidad de ligas internas. Tales obstáculos se resolvieron mediante la comunicación con los demás compañeros del equipo, a través de la cual se pudieron compartir conocimientos y estrategias para desarrollar algunas funcionalidades similares, y así poder construir las de manera más rápida y eficiente.

Otra causa fue que las estaciones de trabajo no tenían el mismo compilador, por lo tanto, surgieron algunos errores de sistema, no del código desarrollado. No haber seguido el estándar de nombres establecido durante el diseño de bajo nivel, provocó que los documentos generados no fueran consistentes con el código implementado.

A lo largo del curso, existieron dificultades de trabajo en la Sala de Cómputo de la Facultad. En el primer ciclo, uno de los equipos transportó el sistema a Windows, utilizando el ambiente de Borland C++. Posteriormente, en el segundo ciclo, de nuevo implementaron el sistema en ambiente Windows y lo trasladaron a la plataforma Unix.

Otro equipo, durante el segundo ciclo, al realizar la implementación en vacaciones, encontraron problemas relacionados con el software. Tuvieron la necesidad de instalar Visual C++, en el cual

no tenían experiencia, por lo tanto, tardaron tiempo en comprenderlo y debido a que no era posible visualizar la ejecución del programa decidieron trabajar en Ada.

No tener un equipo adecuado de trabajo y las facilidades para el desarrollo del mismo, causó gran pérdida de tiempo para la elaboración de los productos.

Aunque es un error mínimo, pero de vital importancia para generar productos de calidad, las faltas de ortografía estuvieron presentes en ciertos documentos, porque el ACP no realizaba revisiones profundas de los mismos.

Como aspecto técnico, se menciona que uno de los equipos utilizó el depurador del sistema para observar la ejecución del programa durante las pruebas de caja negra y caja blanca.

Al momento de ingresar un producto como línea base a la Mesa de Control de Configuración se notificaba tal noticia a todos los miembros del equipo.

Segundo ciclo

Para realizar el segundo ciclo, los tres equipos concluyeron que ya tenían conocimientos más claros y generales de lo que se debía cumplir, para entonces la comprensión de conceptos fue más fácil. La falta de puntualidad aún persistía (aunque en menor grado que en el ciclo anterior) y pocas veces se lograba tener la asistencia completa.

Las actividades de la fase de Planeación mejoraron notablemente en esta segunda etapa, principalmente en la administración del tiempo y en las tareas que se tenían que realizar. Todavía se cometieron algunos errores del ciclo anterior, pero se concluyó que era necesario realizar una práctica continua para eliminarlos.

El intercambio de roles fue mínimo. Tal razón agilizó el desarrollo de actividades, obteniendo mejor calidad en los productos, evitando errores pasados algunas veces y otras, repitiendo errores ya realizados. Se establecieron nuevas técnicas de organización y compromisos en las fechas de entrega.

En uno de los equipos si se realizó el cambio de roles, lo cual fue de gran utilidad para sus integrantes, puesto que les permitió intercambiar y obtener experiencia en nuevas responsabilidades. Este equipo utilizó el método de trabajo en parejas. A pesar de que dos personas realizaban la planeación y el llenado de formas, en ocasiones el tiempo era insuficiente por la cantidad de trabajo extra. De esa manera, también generaron el documento de requerimientos y realizaron la implementación, ellas concluyeron que así se logró una mejor distribución de trabajo y se mejoró la comunicación entre el equipo. Consideraron útil y práctica la creación de la tabla de seguimiento o rastreo, puesto que les permitía realizar consultas eficaces de las funcionalidades y documentos en que se encontraban.

En este equipo no surgieron grandes problemas, ya que cada integrante cumplía con las responsabilidades asignadas, a pesar de haber desempeñado un rol distinto al del primer ciclo.

También en el segundo ciclo se facilitó la implementación porque se redujeron los problemas para realizar la programación y se aplicó reutilización en el código. Realmente los módulos reutilizados fueron pocos, pero eso permitió que las nuevas funcionalidades fueran realizadas con menos dificultades. La experiencia adquirida anteriormente permitió a los alumnos saber cómo debían elaborar los documentos y cómo desarrollar la programación de los módulos para integrarlos más fácilmente.

Las revisiones e inspecciones se realizaron de mejor manera durante este ciclo. Eran moderadas por el Administrador de Calidad y Proceso y registradas en las formas REPINS, RESPL y RESCA. Cada equipo decidió que formas utilizar y prescindir del uso de otras.

El tiempo dedicado a cada actividad por mínima que fuera debía registrarse en la forma REGT. Durante el primer ciclo, ese registro fue detallado, pero para el segundo ciclo, los alumnos

acordaron sólo registrar el tiempo dedicado a las actividades de mayor importancia. También se utilizaron las formas TAREA y CALENDARIO para tales registros.

Desempeño

En ciertas ocasiones el desempeño de algunos integrantes era menor, debido a compromisos extras. El poco tiempo que podían dedicar a las actividades del curso y la incompatibilidad de horarios para realizar las inspecciones fueron las mayores dificultades. Aunque la cooperación y la puesta en marcha de determinadas estrategias, como realizar reuniones los fines de semana (cuando era necesario) permitieron generar el producto requerido. Por lo tanto, se puede decir que cada integrante desempeñó de la mejor manera posible sus roles, logrando un buen apoyo por parte de los demás. Aunque el acoplamiento no fue perfecto, los problemas y errores del primer ciclo sirvieron para desarrollar mejor las actividades en el segundo ciclo.

Los alumnos concluyeron que durante el trabajo en equipo no se deben descuidar tres aspectos básicos e importantes: la puntualidad en las reuniones, la comunicación constante de cada ingeniero sobre el trabajo asignado y la discusión y solución de problemas.

Experiencia adquirida por parte de los alumnos

A través del curso, las experiencias obtenidas de acuerdo al criterio de los alumnos fueron las siguientes:

- Conocimiento adquirido para plantear los requerimientos, mediante casos de uso y escenarios.
- Desarrollo del diseño de alto nivel a través de un esquema para generar la idea central del sistema a partir de la cual se inicia la codificación.
- Identificación de los errores de los equipos para generar un producto de buena calidad, gracias a las revisiones realizadas por el Instructor.
- Establecimiento de formas de comunicación en el equipo de trabajo; conocimiento de la responsabilidad que se adquiere al comprometerse con los demás y la necesidad del apoyo mutuo, los cuales son necesarios para que un equipo salga adelante.
- Desempeño en cada rol, lo cual permitió aprender a escuchar a las demás personas; adquirir responsabilidades de trabajo y compromisos de entrega con uno mismo y con los demás compañeros.

Propuesta de mejoras para la aplicación de TSPi en cursos posteriores

- Asignar dos semanas de clases para las fases grandes, y una semana para las que se consideren de menor tamaño.
- Establecer desde el principio dos ciclos y, si es posible, otro más.
- Tener un curso previo de PSP y de A Discipline for Software Engineering. Se consideró que no haber tenido tal curso afectó el desempeño del equipo.
- Informar sobre las actividades desarrolladas por cada integrante a todo el equipo para mantener una buena comunicación sobre el trabajo realizado.
- Discutir sobre los avances y problemas surgidos para que en equipo se busquen soluciones.
- Utilizar la forma REGT para detallar el registro de tiempo dedicado a actividades básicamente importantes.
- Prescindir del uso de algunas formas, entre ellas SEMANA. El llenado de las mismas ocasionaba conflictos y discusiones en uno de los equipos, y también bastante pérdida de tiempo. Como el responsable de esa tarea era el AP, en ocasiones dedicaba mayor tiempo a esa tarea que a la planeación.
- Entregar las formas RESPL y RESCA hasta el final de cada ciclo y no semanalmente.
- Mantener siempre firmes la actitud y constancia de cada integrante para lograr un buen desempeño de las habilidades de cada uno, y así conjuntar los esfuerzos de todo el equipo.

- Hacer una planeación oportuna de tiempo para las actividades requeridas, una asignación adecuada a cada integrante del equipo y una especificación de actividades que puedan realizarse individualmente.
- Terminar cada ciclo con toda la documentación bien revisada y corregida.
- Adquirir compromiso de puntualidad.
- Entregar un trabajo realmente completo, de acuerdo a las expectativas del cliente.

Como se indicó al inicio del curso, los cinco roles establecidos por TSPi debían ser desempeñados en cada equipo, así como también el rol de ingeniero de desarrollo. En los equipos integrados algunas personas se dedicaban a realizar la documentación necesaria y otras, a desarrollar el código, debido a la facilidad que tenían para programar. Aunque para las personas que sólo se dedicaban a la documentación, en ocasiones era necesario que comprendieran determinados algoritmos para desarrollar algún documento, como por ejemplo, el documento de Pruebas Unitarias.

No haber cursado alguna materia con un proceso de trabajo disciplinado para la elaboración de productos de software causó cierto desbalance en la manera de trabajar de los alumnos. Se consideró de vital importancia dedicar bastante tiempo al desarrollo de cada fase, puesto que cada una de ellas es básicamente importante para obtener buenos resultados.

El segundo ciclo se deduce que fue prácticamente más fácil. Los equipos tenían una idea más clara de las actividades que tenían que desempeñar por la experiencia adquirida en el ciclo anterior, y en algunos casos el tiempo disponible era aprovechado de mejor manera para elaborar los productos de ese ciclo. Aún así, llegaron a la conclusión de que "no es nada fácil planear".

El hecho de que los alumnos tenían que cumplir con otras materias y otros compromisos, a pesar de tener la voluntad de realizar el mejor desempeño no les permitía depurar los productos como se hubiera deseado.

12.6. Evaluación de los equipos

Al finalizar cada ciclo, mediante la forma Evaluación del Equipo por Colegas (EEC), los alumnos evaluaron el trabajo realizado por ellos mismos y por sus compañeros de equipo. Se consideraron varios criterios proporcionados por TSPi, los cuales se referían al trabajo realizado en equipo y a los roles propuestos.

En la evaluación del primer ciclo, se obtuvieron 17 formas, equivalentes al total de alumnos del curso. Y en el segundo ciclo, sólo se obtuvieron 13 formas. Debido a que dos equipos estuvieron formados por seis personas, durante el primer ciclo se asignaron responsable y suplente para los roles de AP y AD. En el segundo ciclo, tales asignaciones solamente se hicieron para AP. Las conclusiones hechas al final de cada característica se realizaron en forma general, y no por cada equipo. Aún así es importante mencionar que las encuestas enumeradas del 1 al 6, pertenecen al equipo Octavos; del 7 al 11, al equipo Phobos D. S. y del 12 al 17, al equipo Dione; todas ellas correspondientes al primer ciclo. En el segundo ciclo las encuestas enumeradas del 1 al 6, pertenecen al equipo Octavos; del 7 al 9, al equipo Phobos D. S. y del 10 al 13, al equipo Dione.

Por cada característica evaluada, se muestran las tablas con el concentrado de puntos obtenidos; las representaciones gráficas de tales resultados correspondientes a los dos ciclos las puede consultar en el Anexo B.

En la forma EEC, la primera evaluación consistió en aplicar una estimación en porcentaje sobre el trabajo requerido y la dificultad encontrada por cada rol para desempeñar las responsabilidades correspondientes. Al final, la suma total tenía que corresponder a cien por ciento.

Iniciamos con la información referente al trabajo requerido y la dificultad encontrada durante el primer ciclo.

En el primer ciclo, no se tomó en cuenta el porcentaje asignado a los roles suplentes. Porque al sumarlo con el porcentaje del rol responsable, se obtenía un resultado mayor comparado con los demás roles. Aunque sí está contenido en el concentrado de puntos, en cada tabla; pero no, en las gráficas.

En el primer ciclo, el rol que requirió mayor trabajo y dificultad para desempeñar sus actividades fue AP. Posteriormente, ACP se posicionó en segundo lugar. A su vez, AA requirió mayor trabajo y menor dificultad. AD fue evaluado con menor trabajo y dificultad. Aunque LE requirió poco trabajo, encontró mayor dificultad para el desempeño de sus actividades.

Rol	No. de formas																	Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
LE	20	15	15	10	15	15	20	15	10	10	10	20	20	20	20	20	17	272
AD	15	15	15	10	15	15	20	25	30	20	20	10	15	15	10	15	14	279
AD												10	15	15	10	10	16	76
AP	15	15	20	20	12	25	20	25	25	20	25	30	20	25	20	23	20	360
AP	15	15	20	20	17	15												102
ACP	20	20	15	20	16	15	20	20	20	20	25	20	15	15	20	17	17	315
AA	15	20	15	20	20	15	20	15	15	30	20	10	15	10	20	15	16	291
Total	100	100	100	100	95	100	1695											

Tabla 12.10 Evaluación del trabajo requerido durante el primer ciclo

Rol	No. de formas																	Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	
LE	20	15	10	10	17	Más o menos	20	15	20	20	10	20	20	30	20	20	17	284
AD	15	15	10	10	16	Más o menos	20	30	20	20	20	10	15	10	20	11	15	257
AD												10	15	10	10	10	14	69
AP	15	18	20	20	18	Difícil	20	30	20	20	30	30	25	30	20	25	20	361
AP	15	18	20	20	18	Difícil												91
ACP	20	17	10	20	15	Regular	20	15	20	20	20	20	15	15	20	20	18	285
AA	15	17	20	20	16	Difícil	20	10	20	20	20	10	10	5	10	14	14	241
Total	100	100	90	100	100		100	98	1588									

Tabla 12.11 Evaluación de dificultad en cada rol durante el primer ciclo

Nota.

En la encuesta no. 6, el alumno no evaluó mediante asignación de porcentajes, aún así se decidió incluirlo en el concentrado de datos.

En el segundo ciclo, debido a que los dos equipos formados por seis integrantes, decidieron asignar un suplente a AP, si se tomó en cuenta el porcentaje correspondiente y se sumó al valor del rol responsable. El trabajo requerido y la dificultad para AP, otra vez, ocupó el lugar principal. En esta iteración, LE requirió mayor trabajo, aunque la dificultad se mantuvo en la misma posición que en el ciclo anterior. ACP tuvo mayor dificultad y requirió de menos trabajo. Los roles AA y AD se situaron, cada uno en los mismo lugares en cada característica, ocupando así las últimas posiciones.

Rol	No. de formas													Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	
LE	20	15		18	20	15	40	30	20	17	20	20	30	265
AD	10	10		17	15	15	10	20	20	16	10	15	10	168
AP	15	20		16	15	15	20	15	20	19	25	20	20	220
AP	15	20		17	15	20				19	25	15	20	166
ACP	20	10	50	17	15	15	15	15	20	13	10	20	10	230
AA	20	15		17	20	20	15	20	20	13	10	10	10	190
Total	100	90	50	102	100	100	100	100	100	97	100	100	100	1239

Tabla 12.12 Evaluación del trabajo requerido durante el segundo ciclo

Rol	No. de formas													Total
	1	2	3	4	5	6	7	8	9	10	11	12	13	
LE	20	15		18	10	15	10	20	12			25	20	165
AD	10	10		17	10	15	20	20	22			20	10	154
AP	15	20		16	25	20	40	20	22			15	30	223
AP	15	20		17	15	20						15	30	132
ACP	20	10	40	17	15	15	15	20	22			15	5	194
AA	20	15		17	25	15	15	20	22			10	5	164
Total	100	90	40	102	100	100	100	100	100			100	100	1032

Tabla 12.13 Evaluación de dificultad en cada rol durante el segundo ciclo

Resumen de trabajo requerido y dificultad encontrada

AP fue el rol que requirió mayor trabajo y encontró mayor dificultad durante los dos ciclos. ACP fue el rol sucesor en los resultados obtenidos en tales aspectos, también durante los dos ciclos. Para AA, durante el primer ciclo, el trabajo requerido fue mayor que la dificultad encontrada; y en el segundo ciclo, el trabajo requerido y las dificultades fueron menores. Otro rol, que tuvo menores puntuaciones en esas características, durante los dos ciclos fue AD, aunque el trabajo requerido y la dificultad encontrada fueron mayores durante el primer ciclo. Respecto a LE, se originó una variación. Durante el primer ciclo, la evaluación de los alumnos en cuanto al trabajo requerido por ese rol produjo el valor menor, pero la dificultad fue mayor, posicionándose en un tercer lugar. Para el segundo ciclo, el trabajo que ese rol requirió incrementó bastante, hasta posicionarse en un segundo lugar y la dificultad se mantuvo en el mismo lugar del ciclo anterior.

Otros criterios como el trabajo en equipo, la calidad del proceso y el producto, contribución, apoyo y ayuda, y desempeño, también se evaluaron en la forma EEC. El método de evaluación aplicado por los alumnos consistió en asignar a cada criterio y rol (de acuerdo a la característica evaluada), un valor de 1 hasta 5, de menor a mayor, respectivamente. Si a cada uno se les hubiese asignado en todas las formas el valor 5, la puntuación máxima y óptima hubiese sido 85, para el primer ciclo (17 formas por 5 puntos) y 65, para el segundo (13 formas por 5 puntos). Para aquellos roles desempeñados por dos personas, la puntuación máxima debió haber sido 115 puntos, a partir de 85 más 30 (éstos últimos, debido a los seis integrantes de dos equipos).

En el segundo ciclo, la puntuación máxima para los roles desempeñados por dos personas debió haber sido 115 puntos. Los dos equipos coincidieron que al rol AP se le asignara responsable y suplente. Como sólo se generaron 13 formas, los 115 puntos se obtendrían a partir de la

puntuación máxima, la cual era 65 más 50 puntos, obtenidos a partir de los diez integrantes de esos dos equipos.

A continuación se presentan las tablas que contienen los concentrados de valores correspondientes a cada característica. En cada tabla se incluye la columna de Promedio para cada criterio o rol. Tal valor es representado por un entero y dos decimales, para mostrar mayor precisión en los resultados obtenidos. También cada total se representa en porcentaje, el cual fue la base para realizar las conclusiones requeridas.

Los porcentajes que contienen las tablas se representan mediante gráficas, las cuales se pueden consultar en el Anexo B.

Criterios	No. de firmas																	Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17			
Espíritu de equipo	3	3	4	4	4	4	3	5	4	4	3	5	5	5	5	5	5	71	4.18	84
Efectividad total	3	4	4	4	5	3	5	4	5	5	3	5	5	5	5	4	5	74	4.35	87
Experiencia adquirida	4	3	5	4	4	5	3	5	4	4	3	4	4	5	5	4	5	71	4.18	84
Productividad de equipo	4	4	4	4	5	3	4	4	4	5	3	5	5	5	5	5		69	4.06	81
Calidad del proceso	3	4	4	4	5	3	3	4	4	4	3	5	4	5	5	3	4	67	3.94	79
Calidad del producto	3	4	4	4	4	4	4	4	5	4	2	4	5	5	5	3	4	68	4.00	80

Tabla 12.14 Evaluación de criterios durante el primer ciclo

Criterios	No. de firmas													Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13			
Espíritu de equipo	3	4	4	5	4	5	5	4	3	5	5	5	5	57	4.38	88
Efectividad total	4	4	5	5	5	4	4	4	4	5	4	5	5	58	4.46	89
Experiencia adquirida	4	4	4	5	5	4	5	4	5	5	4	5	5	59	4.54	91
Productividad de equipo	3	5	5	5	5	4	5	3	4	5	5	4	4	57	4.38	88
Calidad del proceso	4	5	4	5	5	4	4	3	5	4	4	5	5	57	4.38	88
Calidad del producto	4	5	5	5	5	4	4	5	5	4	4	4	5	59	4.54	91

Tabla 12.15 Evaluación de criterios durante el segundo ciclo

Resumen de criterios

En general, los promedios obtenidos en Criterios, durante los dos ciclos resultaron bastante aceptables. En ocasiones, la interacción de personas en un equipo puede generar ciertos problemas, originados probablemente, por la manera de trabajar, el tiempo disponible y la personalidad de cada uno, por mencionar algunos motivos. A pesar de ello, la buena voluntad de cooperación e integración de los integrantes de los equipos, aún con la falta de experiencia y conocimientos en la aplicación de procesos para el desarrollo de software, permitieron que cada producto requerido fuera generado de la mejor manera posible. Fue así, como los criterios de *efectividad* y *espíritu de equipo* fueron palpables en el curso y generaron una *productividad* óptima. Mediante la aplicación de TSPi, que es un proceso de trabajo en equipo el cual se desarrolla a través de iteraciones (en este caso sólo se desarrollaron dos ciclos), permitió a los alumnos adquirir conocimientos en el área de Ingeniería de Software y reforzarlos e incrementarlos en el segundo ciclo.

A pesar de haberse situado en los últimos lugares durante el primer ciclo, la *calidad del producto* aumentó bastante en el ciclo siguiente. El criterio que se posicionó en el último lugar, durante los

dos ciclos, fue *calidad del proceso*, a pesar de eso fue evaluado con mejor puntuación en el segundo ciclo.

La contribución que cada uno de los roles aportó durante la aplicación del proceso se representa a continuación.

Rol	No. de firmas																	Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17			
LE	3	3	3	5	4	3	1	3	3	3	1	5	5	5	5	5	4	61	3.59	72
AD	3	3	3	4	3	3	4	5	4	4	3	4	4	4	5	5	5	66	3.88	79
AD												4	4	4	5	4	4	25	4.17	
AP	4	4	5	5	5	5	4	4	4	4	3	4	5	5	5	5	5	76	4.47	90
AP	4	4	5	5	5	5												28	4.67	
ACP	4	4	4	5	4	4	4	5	4	4	3	5	5	5	5	5	5	75	4.41	88
AA	4	4	5	5	5	5	4	5	5	5	3	4	4	4	5	5	4	76	4.47	89

Tabla 12.16 Evaluación sobre la contribución de cada rol durante el primer ciclo

Rol	No. de firmas													Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13			
LE	4	5	5	5	5	3		4	3	5	5	5	5	54	4.15	83
AD	3	4	3	5	3	2	4	4	4	5	4	5	5	51	3.92	78
AP	4	5	5	5	5	4	4	3	4	5	5	5	5	59	4.54	93
AP	4	5	5	5	5	5				5	5	4	5	48	4.80	
ACP	3	4	3	5	4	3	5	3	4	5	3	4	4	50	3.85	77
AA	4	5	5	5	5	5	5	3	4	5	3	4	4	57	4.38	88

Tabla 12.17 Evaluación sobre la contribución de cada rol durante el segundo ciclo

El apoyo y ayuda proporcionados por cada rol y el desempeño de cada uno fueron características que también se evaluaron en los dos ciclos. A continuación se presentan las tablas correspondientes.

Rol	No. de firmas																	Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17			
LE	3	3	3	4	4	4	1	3	4	3	1	5	5	5	5	5	5	63	3.71	74
AD	2	3	3	4	4	3	4	5	4	4	3	5	4	5	5	5	5	68	4.00	83
AD												5	4	5	5	4	5	28	4.67	
AP	4	4	5	5	5	4	4	5	4	4	3	5	5	5	5	5	5	77	4.53	91
AP	4	4	5	5	5	5												28	4.67	
ACP	4	4	4	5	5	4	4	5	4	4	3	5	5	5	5	5	5	76	4.47	89
AA	4	4	5	5	5	5	4	5	5	4	3	5	4	4	5	5	5	77	4.53	91

Tabla 12.18 Evaluación del apoyo y ayuda durante el primer ciclo

Rol	No. de formas													Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13			
LE	4	5	5	4	5	3		3	3	5	5	5	5	52	400	80
AD	2	4	3	4	3	2	4	4	4	5	4	4	5	48	369	74
AP	4	5	5	4	5	4	4	4	4	5	4	5	5	58	446	91
AP	4	5	5	4	5	5				5	4	5	5	47	470	
ACP	3	4	3	4	4	3	5	4	4	4	4	4	4	50	385	77
AA	4	5	5	4	5	5	5	4	4	4	4	4	4	57	438	88

Tabla 12.19 Evaluación del apoyo y ayuda durante el segundo ciclo

El desempeño que cada uno de los roles tuvo durante los ciclos, fue otra característica evaluada.

Rol	No. de formas																	Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17			
LE	3	3	3	5	4	3	1	3	3	3	2	5	5	5	5	5	4	62	365	73
AD	3	3	3	4	4	3	3	5	4	4	4	4	4	5	5	5	5	68	400	82
AD												4	4	5	5	4	4	26	433	
AP	4	4	5	5	5	5	3	5	4	4	4	5	5	5	5	5	5	78	459	92
AP	4	4	5	5	5	5												28	467	
ACP	4	4	4	5	5	4	4	5	4	4	4	5	5	5	5	5	5	77	453	91
AA	4	4	5	5	5	5	5	5	5	4	4	4	4	5	5	5	4	78	459	92

Tabla 12.20 Desempeño por rol durante el primer ciclo

Rol	No. de formas													Total	Promedio	Porcentaje
	1	2	3	4	5	6	7	8	9	10	11	12	13			
LE	4	5	5	5	4	3		3	4	5	5	5	5	53	408	82
AD	2	4	3	5	3	2	4	4	4	5	4	5	5	50	385	77
AP	4	5	5	5	5	4	4	4	4	4	4	5	5	58	446	93
AP	4	5	5	5	5	5				5	5	5	5	49	490	
ACP	3	4	4	5	4	3	5	4	4	5	3	5	4	53	408	82
AA	4	5	5	5	5	5	5	5	5	5	3	5	4	61	469	94

Tabla 12.21 Desempeño por rol durante el segundo ciclo

Conclusión de trabajo requerido, contribución, apoyo y ayuda, y desempeño

En casi todas las características evaluadas, AP fue el rol cuya puntuación siempre se mantuvo en primer lugar, con excepción del desempeño que tuvo en el segundo ciclo, el cual fue rebasado mínimamente por AA.

El rol mencionado anteriormente, fue el sucesor de AP, en cuanto a la contribución, apoyo y ayuda, y el desempeño realizados. Aunque no tuvo gran dificultad en el trabajo que requirió realizar durante los dos ciclos, posicionándose entre los últimos lugares.

La contribución, apoyo y ayuda, y desempeño de ACP durante el primer ciclo obtuvo valores más altos que en el segundo ciclo, aunque el trabajo requerido y la dificultad para realizar las actividades requeridas no variaron mucho durante los dos ciclos.

Debido a la falta de conocimientos para generar el producto de cada fase, AD tuvo mayor dificultad en el primer ciclo para realizar las actividades asignadas. Aunque disminuyó en el siguiente ciclo, por los conocimientos adquiridos anteriormente. Durante el primer ciclo, los responsables de este rol fueron evaluados aceptablemente en cuanto al apoyo y ayuda, y desempeño; pero en el segundo ciclo tales aspectos y además la contribución de tal rol se posicionaron en los últimos lugares, concluyendo así con una mala posición.

En el primer ciclo, el trabajo requerido por LE obtuvo el valor menor y se evaluó la dificultad para desempeñar las actividades de manera regular (o término medio), pero en el segundo ciclo la dificultad se mantuvo, otra vez, en término medio detectando la necesidad de realizar mayor trabajo que en el ciclo anterior, posicionándolo entre los dos primeros lugares del trabajo requerido. Algo similar sucedió con la contribución, apoyo y ayuda, y desempeño; tales características fueron evaluadas entre los últimos lugares, durante el primer ciclo, pero en el segundo obtuvieron una posición regular.

12.7. Evaluación final

Se concretaron las siguientes conclusiones de acuerdo a la información proporcionada mediante los documentos siguientes: Reportes Semanales; Reporte Postmortem; Aspectos Sociales, Técnicos y del Proceso y Evaluación de los equipos (mediante la forma EEC).

Necesidades

Al extraer la información que la encuesta recabó y conocer la irregularidad de conocimientos académicos entre los alumnos se consideró conveniente que el Instructor, antes de iniciar el curso conozca los antecedentes académicos de los alumnos para evaluar la necesidad de realizar una introducción sobre TSPi y brindar información sobre los temas de PSP y A Discipline for Software Engineering requeridos como prerequisite.

Aunque en TSPi se incluyen áreas de la Ingeniería de Software, fue necesario que el Instructor impartiera temas relacionados con métricas, roles, fases, inspección y revisión, administración de configuración, técnicas de documentación y procesos para generar los productos de las fases mencionadas. También sería benéfico tener conocimientos previos sobre las técnicas de modelado de requerimientos y diseño, para la agilización y mejor comprensión de los productos generados en las fases relacionadas.

Experiencia y conocimientos adquiridos

La aplicación de un proceso disciplinado para desarrollar software permitió a los alumnos adquirir nuevos conocimientos, los cuales se mencionan a continuación: aprendieron a plantear los requerimientos de un sistema mediante casos de uso; realizaron el diseño de alto nivel para generar el esquema que se utilizaría como base para realizar la codificación; aprendieron a establecer la comunicación en un equipo de trabajo; adquirieron consciencia de la responsabilidad que se genera al comprometerse con los demás integrantes; conocieron la necesidad del apoyo mutuo y la importancia de aplicar revisiones, las cuales les permitió conocer sus errores y generar un producto de buena calidad; en las áreas de técnicas de documentación, roles, e inspecciones y revisiones los alumnos adquirieron el 70% de conocimientos; 67% en las fases; los porcentajes menores fueron en administración de configuración, evaluada en 63% y 55% para métricas.

Experiencia y conocimientos en el segundo ciclo

La aplicación de un proceso disciplinado para desarrollar software permitió a los alumnos adquirir nuevos conocimientos y reforzarlos en la segunda iteración. A continuación se mencionan los que ellos señalaron: se mejoró la administración del tiempo en la fase de planeación; adquirieron conocimientos claros y precisos sobre las actividades y tareas que se debían realizar y los productos que se tenían que generar, lo cual permitió que las actividades del segundo ciclo se desarrollaran con mayor destreza y facilidad; establecieron compromisos precisos en las fechas de entrega de los productos; cada equipo pudo decidir cuáles formas utilizar; acordaron registrar en REGT el tiempo dedicado a actividades de mayor importancia, también utilizaron las formas TAREA y CALENDARIO para tales registros; aconsejaron utilizar RESPL y RESCA al final de cada ciclo; se redujeron los problemas de programación, lo cual les facilitó la implementación; reutilizaron código, lo cual permitió realizar la integración de módulos más fácilmente; redujeron el número de defectos porque las revisiones e inspecciones se realizaron de mejor manera y mejoraron el desempeño de los roles en cada fase.

Motivos que afectaron el curso

El curso fue afectado por las siguientes razones: dado que la mayoría de los problemas surgieron durante el primer ciclo, en ocasiones mientras los alumnos estaban generando documentos de la fase en la cual se estuviera trabajando, también tenían que corregir documentos de la fase anterior, eso provocó desbalance en la asimilación de conocimientos de las dos fases; durante el transcurso del semestre persistieron los problemas en la Sala de Cómputo, no se permitió que los alumnos trabajaran en equipo, otro obstáculo fue que las diversas plataformas de Unix, en las estaciones de trabajo no tenían el mismo compilador; la reducción de los ciclos de trabajo fue origen de varias causas, entre ellas, el desconocimiento e inexperiencia de los alumnos sobre algún proceso para el desarrollo de software, las cuales ocasionaron problemas, principalmente, a partir de la fase de Planeación, porque no sabían que actividades deberían realizar; en el primer ciclo, en la fase de Estrategia, fue necesario realizar ajustes en las primeras estimaciones realizadas respecto al tamaño de los productos, planeación fue la fase que causó mayor conflicto; porque los documentos requeridos debían realizarse en una semana y los métodos para completar las formas no eran fácilmente entendibles; el tiempo dedicado a las actividades del proceso no era suficiente, los alumnos tenían que asignar tiempo para las otras materias de la licenciatura, y algunos de ellos, dedicar otra parte del mismo a su trabajo laboral, por tanto, tales razones no les permitían depurar los productos como hubiesen deseado y, a la vez, provocaba retraso en las actividades de la materia; el tiempo dedicado a temas adicionales, por ejemplo las técnicas de análisis causó retraso en tiempo y problemas a los alumnos para generar los documentos requeridos; principalmente en el primer ciclo porque era la primera vez que aplicaban esas técnicas; la falta de experiencia y disciplina de los alumnos en un proceso ocasionó que generaran un plan de actividades inadecuado para sus equipos; demasiados días festivos y obligatorias interrupciones de clases, ocasionaron el retraso de los alumnos en el desempeño de las actividades por realizar; el nivel de inglés afectó en grado menor la fácil asimilación del material bibliográfico; la ausencia de algunas personas en las juntas de equipo ocasionó que algunos productos a generar no se realizaran tomando en cuenta los acuerdos establecidos; los defectos encontrados en la versión beta de la herramienta TSPi (al inicio del curso) y los problemas surgidos durante la implementación, porque algunos alumnos no estuvieron en contacto con la programación los seis meses anteriores al curso, originándoles conflictos al momento de programar.

Aspectos técnicos

La mayoría de los alumnos se apoyaron en el libro de TSPi y en las clases impartidas por el instructor. Dos equipos utilizaron el depurador del compilador de C para aplicar las pruebas unitarias.

Revisiones del producto de los dos ciclos

Durante la revisión del producto en el primer ciclo, los equipos Phobos D.S. y Dione desarrollaron un sistema que en forma general cumplió con las funciones especificadas en el Plan de Estrategia,

aunque existieron algunas inconsistencias ya mencionadas. El producto entregado por parte del equipo Octavos tuvo mayores problemas, que fueron ocasionados por la falta de organización, de comunicación entre sus integrantes y por no haber aplicado inspecciones, revisiones y pruebas adecuadamente.

En el segundo ciclo se añadió una semana a las actividades del curso para realizar la prueba de aceptación por parte del cliente. A pesar de que esta semana no se contempló dentro del guión original, el Instructor permitió a los equipos que durante la misma pudieran finalizar sus actividades y efectuar las revisiones.

En la segunda revisión de aceptación, a pesar de que el Instructor recalcó ciertas sugerencias importantes para que fueran tomadas en cuenta y que uno de los equipos tuvo problemas en cierta funcionalidad, cabe mencionar que se mejoró bastante la calidad del producto de los tres equipos.

Resumen de roles de acuerdo a la forma EEC

El rol Administrador de Planeación siempre destacó en todas las características evaluadas, el trabajo requerido y la dificultad para el desempeño de sus actividades fueron mayores que en los demás roles. Y la contribución, apoyo y ayuda y desempeño del rol se mantuvo casi siempre en los primeros lugares.

Otro rol que destacó durante los dos ciclos en cuanto a contribución, apoyo y ayuda, y desempeño fue Administrador de Apoyo, posicionándose en primer lugar, aunque el trabajo requerido por este rol fue mayor durante el primer ciclo, pero con una dificultad mínima para su realización; en el segundo ciclo el trabajo disminuyó y la dificultad también.

El trabajo requerido por el Administrador de Calidad y Proceso y la dificultad que este rol encontró durante los dos ciclos, ocasionó que se colocara entre los tres primeros lugares, así mismo en el rubro de desempeño; aunque la contribución, apoyo y ayuda durante el primer ciclo fueron mayores que en la segunda iteración (eso significa que la posición estuvo entre los últimos lugares).

El trabajo requerido y la dificultad que encontró el AD para realizar las responsabilidades estuvo entre las últimas posiciones durante los dos ciclos. Siendo mayor en la primera iteración, probablemente por la falta de conocimientos para generar los productos de cada fase. También la contribución durante los dos ciclos, así como el apoyo y ayuda y desempeño durante el segundo ciclo se posicionaron entre los últimos lugares. En el primer ciclo, el trabajo requerido por el Líder de Equipo obtuvo una puntuación mínima y la dificultad que los alumnos percibieron fue evaluada con una puntuación regular, probablemente las actividades de este rol no estuvieron claramente comprendidas, porque en la siguiente iteración el trabajo requerido obtuvo puntuaciones altas, colocándose entre los dos primeros lugares, y la dificultad se mantuvo evaluada regularmente. También la contribución, apoyo y ayuda, y desempeño estuvieron evaluados en forma mínima durante el primer ciclo, mejorando para ello con puntuaciones regulares en el segundo ciclo.

Resumen de criterios de acuerdo a la forma EEC

Los porcentajes obtenidos en los criterios del trabajo en equipo, los cuales fueron el espíritu de equipo, efectividad total, experiencia adquirida, productividad de equipo, calidad del proceso y calidad del producto estuvieron aceptables. En ocasiones, la interacción de personas en un equipo puede generar ciertos problemas, originados probablemente, por la manera de trabajar, el tiempo disponible y la personalidad de cada uno, por mencionar algunos motivos. A pesar de ello, la buena voluntad de cooperación e integración, aún con la falta de experiencia y conocimientos en la aplicación de procesos para el desarrollo de software, permitieron verificar la importancia de la efectividad del trabajo en equipo. Y con ello lograr una productividad aceptable, fundamental en la generación del producto final.

Aunque la calidad del mismo, durante el primer ciclo fue mínima, en la segunda iteración los resultados de tal aspecto tuvieron un incremento bastante aceptable (posicionándose en el primer lugar de los aspectos evaluados).

A pesar de que la calidad del proceso durante el primer ciclo, tuvo una evaluación mínima, en el segundo ciclo mejoró bastante. El aprendizaje de un proceso disciplinado para desarrollar software y de algunos aspectos necesarios que se incluyen en Ingeniería de Software permitieron a los

alumnos adquirir conocimientos y experiencia durante los dos ciclos, este aspecto ocupó siempre los primeros lugares de las puntuaciones, la cual aumentó en el segundo ciclo.

Sugerencias

Los alumnos clasificaron las formas por su utilidad. CAMC, REPC, REPINS, REGD y RESTM se consideraron de uso óptimo. Con respecto a TAREA, CALENDARIO y SEMANA, cada equipo era libre de decidir cuál(es) utilizar. El tiempo de las actividades realizadas se indicaba en la forma REGT, pero los alumnos opinaron que sería mejor sólo registrar las actividades relevantes. Las formas que no se utilizaron se referían al concentrado de resúmenes de defectos, siendo esas, RESDI y RESDE. Otras formas que se consideraron redundantes fueron RESTDES y RESTAREA. A continuación se mencionan las siguientes sugerencias para cursos posteriores: establecer fechas de reunión periódicamente antes de iniciar el proyecto; eliminar las formas que se consideren redundantes y utilizar las que el equipo decida de acuerdo a sus necesidades (por ejemplo SEMANA, TAREA y CALENDARIO); proponer dos ciclos desde el principio; si existiera tiempo, otro más; asignar dos semanas a las fases de Planeación, Diseño e Implementación; y una semana a las demás fases (porque se consideraron de menor tamaño); hacer un adecuado plan de trabajo para las personas que tengan actividades laborales independientes de las académicas e identificar aquellas que puedan realizarse individualmente; establecer mayor disciplina para mantener control sobre cada producto y sus versiones y lograr una mejor organización en la administración de configuración; procurar que el LE interactúe el mayor tiempo posible con los integrantes del equipo para conocer su forma de pensar, lo cual le permitirá tomar mejores decisiones y motivar a cada uno de ellos; brindar información sobre el tema de liderazgo y evitar la documentación en exceso, porque como criterio de los alumnos, opinaron que no proporciona beneficios en el desarrollo de un sistema.

En Resumen

Es importante señalar a los alumnos la necesidad de tener tiempo disponible para el desarrollo del proceso, puesto que el trabajo en equipo, requiere realizar toma de decisiones en conjunto. Como uno de los equipos, en el segundo ciclo realizó el trabajo en parejas, eso les permitió lograr una mejor distribución de trabajo y mejorar la comunicación entre sus integrantes.

Siempre es conveniente considerar durante la aplicación del proceso el compromiso de puntualidad en las reuniones; la comunicación constante de cada ingeniero sobre el trabajo asignado, puesto que con ellos es posible compartir conocimientos y establecer estrategias para la solución de problemas en equipo.

Es importante que la actitud de cada integrante y la constancia se mantengan firmes siempre para lograr un buen desempeño de las habilidades de cada uno y así conjuntar los esfuerzos de todo el equipo.

Los resultados que se obtienen en un equipo son de mayor calidad debido a que un grupo de personas unen un conjunto de conocimientos que posteriormente comparten entre sí; modifican ciertas formas de desempeño al momento de realizar determinadas actividades y logran una mayor eficiencia.

A pesar del número considerado de días festivos, la inestabilidad política en la Universidad, los problemas para aplicar TSPi, la reducción de tres ciclos a dos, los problemas internos de cada equipo y la necesidad de asignar tiempo a otras actividades, se concluye que la aplicación de TSPi se realizó de manera aceptable, gracias al empeño de los alumnos y bajo la dirección y gran apoyo de la Dra. Hanna Oktaba.

Conclusiones

Cuando se va a desarrollar un sistema, mediante un equipo de personas, surgen diversas necesidades. Los integrantes deben definir qué responsabilidades tendrá cada uno de ellos; cómo van a realizar las mismas; que técnicas, herramientas, procedimientos y que proceso deben seguir para lograr un trabajo aceptable. Cuando no se tiene una guía o experiencia adquirida para cumplir con esas necesidades, es mayor la dificultad surgida. Se puede decir que TSPi, es un proceso que presenta un conjunto de pasos que integran las actividades necesarias para la planeación, desarrollo y seguimiento de un proyecto de software, y que permiten cumplir con las necesidades mencionadas anteriormente.

El desarrollo de la presente tesis tuvo como objetivos conocer y evaluar la manera en que el proceso TSPi podía ser aplicado y desarrollado por un grupo de estudiantes mexicanos. Durante la primera lectura realizada al libro de texto, surgió información que no era claramente entendida, posteriormente, surgió la propuesta de realizar el presente trabajo y, de esa manera, fue más clara la comprensión que pude adquirir de la información que no había comprendido, además de que pude vivir personalmente el desempeño de los equipos integrados.

Durante el desarrollo del curso, pude constatar que para los estudiantes fue difícil cumplir con la aplicación del proceso, por las razones mencionadas en la segunda parte de esta tesis y por la relevante razón de que nunca habían utilizado un proceso disciplinado para el desarrollo de software.

Aún así, al cumplir las necesidades descritas, al momento en que un grupo de personas interactúan entre sí, para cumplir un objetivo común, se obtienen beneficios y se generan problemas. Personalmente, una vez más pude verificar que en el trabajo en equipo, no todos los integrantes están de acuerdo con las decisiones tomadas por el equipo, algunos no dedican el tiempo y esfuerzo necesario para este tipo de trabajo, otras personas no están acostumbradas o no tienen facilidad para trabajar con los demás, y por tal razón, la interacción se vuelve más difícil, por mencionar algunas dificultades. En cambio, también pude constatar que mediante el trabajo en equipo, es factible generar un producto final de buena calidad, porque se unen y combinan un conjunto de habilidades o destrezas de los diferentes participantes, se adquieren nuevos conocimientos que ellos mismos aportan entre sí, se presentan diferentes técnicas de trabajo, entre las cuales se puede decidir cuál es más conveniente utilizar para el equipo completo, por mencionar algunos beneficios. Por tales razones, en este tipo de trabajo, es importante que cada integrante tenga la mejor disposición de trabajar en equipo. Aún cuando TSPi es un proceso guía en el desarrollo de software, los problemas no estuvieron alejados en el curso, aunque los estudiantes pudieron conocer y vivir los beneficios que pueden generarse mediante la mejor disposición de las personas que lo integran, porque les permitió valorar que cada uno de ellos es una parte importante para generar un producto final, y que el adecuado desarrollo de las actividades correspondientes, permite generar un producto de buena calidad; también se dieron cuenta que la falta de cumplimiento y colaboración de alguno de los integrantes es una causa que afectaba al equipo completo.

De manera personal, pienso que el desarrollo de software mediante ciclos, permite afinar y profundizar los conocimientos requeridos; aunque el primer ciclo sea más difícil de desarrollar, sobre todo, cuando es la primera vez que se va a utilizar determinado proceso.

En esta tesis se presenta la parte teórica del proceso, los guiones y algunas tablas y formas más útiles traducidas al español, cuyo uso pueda aplicarse en cursos posteriores. Y por último, lo anteriormente descrito se pone a disposición y forma parte de la página de TSPi, la cual es www.dynamics.unam.edu/TSPi.

Referencias bibliográficas

[CMM, 1994]

"CMM, Capability Maturity Model: Guidelines for Improving the Software Process", Carnegie Mellon University, Software Engineering Institute Series in Software Engineering, Addison-Wesley, 1194.

[Davis, 1993]

Davis Alan M., "Software Requirements, Objects, Functions, and States", Prentice Hall, 1999.

[Humphrey, 1995]

Humphrey Watts. S, "A Discipline for Software Engineering", Addison-Wesley, 8va. Impresión, 1995.

[Humphrey, 1997]

Humphrey Watts. S, "Introduction to the Personal Software Process", SEI Series in Software Engineering, Addison Wesley, 1997.

[Humphrey, 2000]

Humphrey Watts. S, "Introduction to The Team Software Process", SEI Series in Software Engineering, Addison Wesley, 2000.

[Pressman, 1997]

Pressman Roger S., "Software Engineering: A Practitioner's Approach", New York: McGraw - Hill Companies, Inc., 4ª. Edición, 1997.

[Sommerville and Sawyer, 1997]

Sommerville, I., and Sawyer, P., "Requirements Engineering: A Good Practice Guide", New York: John Wiley & Sons, 1997.

Carpetas de trabajo de los equipos

[Equipo Dione, 2000]

Carpeta de trabajo del equipo Dione, generada durante el curso de Ingeniería de Software a nivel licenciatura, UNAM, 2000.

[Equipo Octavos, 2000]

Carpeta de trabajo del equipo Octavos generada durante el curso de Ingeniería de Software a nivel licenciatura, UNAM, 2000.

[Equipo Phobos, D. S., 2000]

Carpeta de trabajo del equipo Phobos, D. S. generada durante el curso de Ingeniería de Software a nivel licenciatura, UNAM, 2000.

Sitio en internet

[WEB1]

<http://www.sei.cmu.edu/tsp/>

Anexos

Defect profile

Una gráfica que representa la historia en la eliminación de defectos de un programa por fase. Generalmente está dada en defectos/KLOC.

DIS

Identificador que se utiliza para hacer referencia a la fase de Diseño o a cualquier elemento que intervenga en la misma.

DIS1

Identificador que se refiere al primer ciclo de la fase de Diseño.

DISn

Identificador que se refiere a ciclos posteriores de la fase de Diseño.

EDS

Especificación de Diseño de Software.

EEC (Team and Peer Evaluation, PEER)

Forma Evaluación del Equipo por Colegas; se utiliza para evaluar el trabajo en equipo y de cada colega, se aplica al terminar cada ciclo.

ERS

Especificación de Requerimientos de Software

ESTRA

Identificador que se utiliza para hacer referencia a la fase de Estrategia o a cualquier elemento que intervenga en la misma.

ESTRA1

Identificador que se refiere al primer ciclo de la fase de Estrategia.

ESTRAn

Identificador que se refiere a ciclos posteriores de la fase de Estrategia.

IMP

Identificador que se utiliza para hacer referencia a la fase de Implementación o a cualquier elemento que intervenga en la misma.

IMP1

Identificador que se refiere al primer ciclo de la fase de Implementación.

IMPn

Identificador que se refiere a ciclos posteriores de la fase de Implementación.

INFO (Student Information Sheet, INFO)

Forma que se completa con información de los alumnos.

INS

Guión INS, es aquel que se utiliza para realizar inspecciones.

KLOC

Miles de líneas de código.

KPA's

Áreas Claves del Proceso.

Actividades cuya aplicación permite alcanzar un conjunto de objetivos importantes en el proceso de maduración durante el desarrollo y mantenimiento de software.

PPS

Plan de Pruebas del Sistema.

PRUEBAS

Identificador que se utiliza para hacer referencia a la fase de Pruebas de Integración y del Sistema o a cualquier elemento que intervenga en la misma.

PRUEBAS1

Identificador que se refiere al primer ciclo de la fase de Pruebas de Integración y del Sistema.

PRUEBASn

Identificador que se refiere a ciclos posteriores de la fase de Pruebas de Integración y del Sistema.

PU

Guión de Pruebas Unitarias.

RAR (Issue Tracking Log, ITL)

Forma Registro de Asuntos y Riesgos.

REGD (Defect Recording Log, LOGD)

Forma Registro de Defectos.

REGPR

Forma Registro de Pruebas.

REGT (Time Recording Log, LOGT)

Forma Registro de Tiempo para registrar el tiempo dedicado a las actividades. Pertenece a PSP.

REPC (Configuration Status Report, CSR)

Reporte de Estado de Configuración.

REPINS (Inspection Report, INS)

Reporte de Inspección.

REQ

Identificador que se utiliza para hacer referencia a la fase de Requerimientos o a cualquier elemento que intervenga en la misma.

REQ1

Identificador que se refiere al primer ciclo de la fase de Requerimientos.

REQn

Identificador que se refiere a ciclos posteriores de la fase de Requerimientos.

RESCA (Quality Plan, SUMQ)

Forma Plan de Calidad.

RESDE (Defects Removed Summary, SUMDR)

Forma Resumen de Defectos Eliminados.

RESDI (Defects Injected Summary, SUMDI)

Forma Resumen de Defectos Inyectados.

RESPL (Plan Summary, SUMP)

Forma Resumen del Plan del Proyecto.

El equipo Dione comentó: "Nuestro principal problema fue durante la fase de Planeación, porque era la primera vez que planeábamos el desarrollo de un sistema, involucrando el código con la respectiva documentación indicada por TSPi".

El equipo Octavos comentó: "La fase de Planeación es la etapa más laboriosa del proceso, creemos que lo más conveniente es asignar más de un responsable como AP, y no sólo una persona como lo sugiere TSPi. Opinamos lo anterior porque al momento de tomar decisiones existen más personas con los conocimientos de esta fase, lo cual es benéfico para los demás integrantes del equipo al momento de desempeñar las tareas asignadas".

Análisis de las inspecciones y revisiones

A continuación se presentan los comentarios de los alumnos sobre las revisiones e inspecciones que realizaron durante el curso.

Phobos, D. S. comentó: "Las inspecciones no se pudieron hacer de manera general, debido al problema de incompatibilidad de horarios entre nuestros integrantes".

Dione comentó: "Sí pudimos realizar las inspecciones y revisiones a cada documento, puesto que teníamos un horario fijo para ello".

Octavos comentó: "No teníamos compatibilidad en los horarios, por lo que al inicio del proyecto sí pudimos aplicar las pruebas como lo indica TSPi, pero al final fue más difícil".

A continuación se presenta la tabla que agrupa los problemas y mejoras durante el primer ciclo.

Equipo	Proceso	Roles	Reuniones de Equipos	Revisión de formas
Dione	<p>Problemas:</p> <ul style="list-style-type: none"> Falta de tiempo por ser demasiado el trabajo a realizar. En la fase de planeación. Se hizo un DBN de baja calidad. <p>Mejoras:</p> <ul style="list-style-type: none"> Que dos personas desempeñen el rol de AP. En DBN se debe de explicar todo en términos de lenguaje común. 	<p>Problemas:</p> <ul style="list-style-type: none"> En cuanto a la planeación, el AP opinó que era demasiada la información que se tenía que procesar y calcular. 	<p>Al principio si se generaba la agenda (hasta la mitad del ciclo 1), después ya no.</p> <p>Mejoras:</p> <ul style="list-style-type: none"> Hacer nuevo acuerdo de las reuniones semanales. Realizar un seguimiento de lo que ocurre en cada reunión apuntando todo. 	<p>Acordar cuales revisiones se van a utilizar y entregar al AP.</p>
Octavos	<p>Problemas:</p> <ul style="list-style-type: none"> Horario difícil entre los integrantes. En DBN y en las inspecciones no había tiempo para hacer inspecciones minuciosas y esto generó problemas en la consistencia de integración de módulos. <p>Mejora:</p> <ul style="list-style-type: none"> Se propuso usar la forma PMP para 	<p>Problemas:</p> <ul style="list-style-type: none"> Problemas en el desempeño de los roles de AP y AD. 	<p>Al principio si se generaba la agenda, después de la fase de diseño no.</p> <p>Mejoras:</p> <ul style="list-style-type: none"> Hacer nuevo acuerdo de las reuniones semanales. Realizar un seguimiento de lo que ocurre en cada reunión apuntando 	<p>Acordar cuales revisiones se van a utilizar y entregar al AP.</p>

Forma	Utilidad	Equipo Dione	Equipo Phobos D. S.	Equipo Octavos
Sirvió de apoyo para completar RESPL.				
RESTDES	NO			
RESTAREA	NO			
RESTDES y RESTAREA fueron formas redundantes y sirvieron de apoyo para completar RESPL.				

Tabla B.3 Análisis de formas de planeación

La siguiente tabla muestra los datos registrados en RESPL y RESCA, indicando datos estimados y reales.

Forma	Característica	Documento	Equipos					
			Estimado			Real		
			Dione	Phobos	Octavos	Dione	Phobos	Octavos
RESPL	Tamaño del producto	ERS	6	6	12	15	10	10
		Otros			153	60	45	10
		DAN	7	7	25	12	190	200
		DBN	3	9	25	13		
		LOC añadidas	150	270	240	242	320	600
	Tiempo en horas	ERS	26	38.84	25	26.3	12	24
		Inspecciones de DAN	18.5	22.8	8	16	12	14
		Codificación	50	36.75	27	34.55	24	40
	Defectos encontrados	DAN	5	7	13	8	6	8
		Compilación	21	47	42	10	5	10
		Total de defectos	339	149	445	291.3	104	105
RESCA		LOC/Hr.						
		Requisitos	0.25	0.49	0.25	1.25	0.30	1.29

Tabla B.4 Datos estimados y reales

Gráficas correspondientes a la evaluación de los equipos

La presente sección contiene las representaciones gráficas de los porcentajes obtenidos a partir de los resultados generados mediante la aplicación de la forma EEC.

A continuación se representan las gráficas correspondientes al trabajo requerido y a la dificultad encontrada durante el primer y segundo ciclo.

EVALUACIÓN DEL EQUIPO

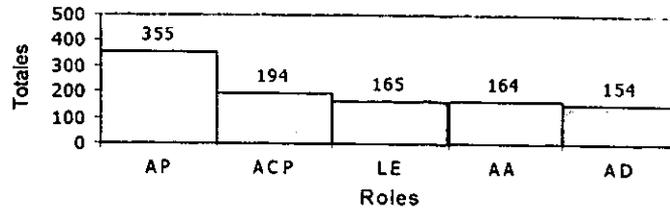


Figura B.4 Dificultad en cada rol durante el segundo ciclo

A continuación se representan las gráficas correspondientes a los Criterios durante el primer y segundo ciclo

EVALUACIÓN DEL EQUIPO

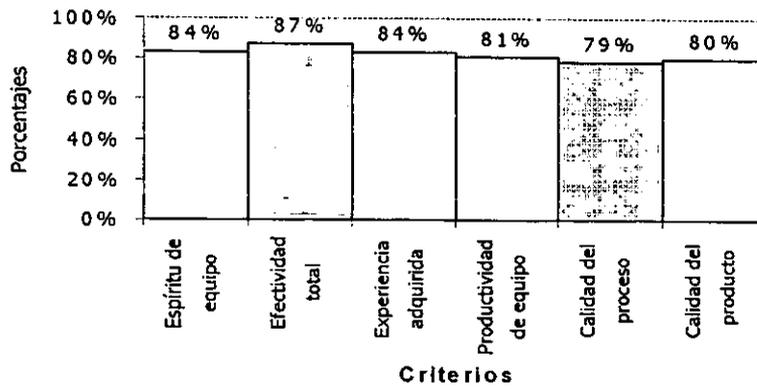


Figura B.5 Porcentaje de criterios durante el primer ciclo

A continuación se representan las gráficas correspondientes al Apoyo y Ayuda de cada rol durante el primer y segundo ciclo.

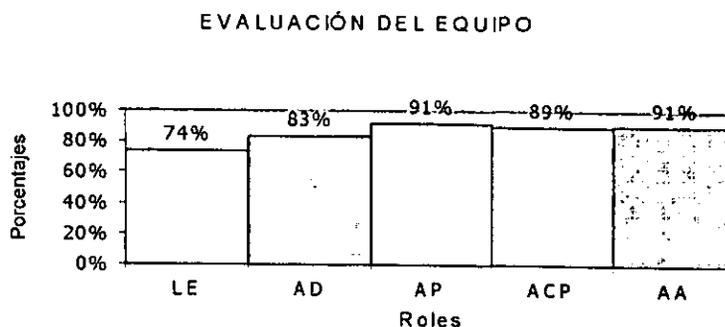


Figura B.9 Porcentaje de Apoyo y Ayuda durante el primer ciclo

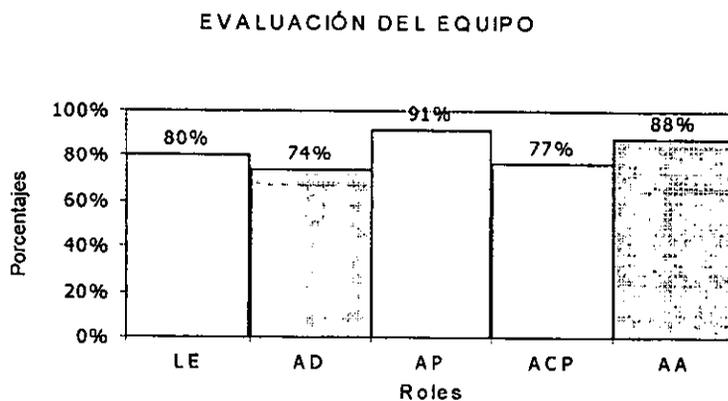


Figura B.10 Porcentaje de Apoyo y Ayuda durante el segundo ciclo

A continuación se representan las gráficas correspondientes al Desempeño de cada rol durante el primer y segundo ciclo.

Describa brevemente cualquier experiencia en liderazgo o en administración (ya sea en el área laboral o en organizaciones)

Si existen preferencias en equipo indíquelas

Liste su horario de clases y de otras actividades semanales, tales como trabajo, organizaciones, equipos de deporte, etc.							
Hora	Lunes	Martes	Miércoles	Jueves	Viernes	Sábado	Domingo
8:00-9:00							
9:05-10:00							
10:05-11:00							
11:05-12:00							
12:05-13:00							
13:05-14:00							
14:05-15:00							
15:05-16:00							

Marque los roles de su preferencia utilizando un rango desde 1 (menor) hasta 5 (mayor)					
Líder de equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad/Proceso	1	2	3	4	5
Administrador de Apoyo	1	2	3	4	5

Forma C.1 Forma INFO

En el siguiente guión se describen las actividades que deben cumplirse para llevar a cabo las reuniones que semanalmente debe realizar cada equipo.

Objetivo	Guiar al equipo a dirigir la reunión semanal.	
Criterio de entrada	<ul style="list-style-type: none"> • Todos los integrantes del equipo deben estar presentes. • Todos han entregado las formas TAREA, CALENDARIO y SEMANA al AP. • El AP ha generado una forma SEMANA, la cual es el reporte semanal de los datos de las actividades de cada integrante. • El LE ha realizado la agenda de la reunión. 	
General	<p>Para avanzar, el LE ha:</p> <ul style="list-style-type: none"> • Pedido a los integrantes que identifiquen los asuntos de la agenda semanal. • Preparado y distribuido la agenda de la reunión. <p>El LE dirige la reunión semanalmente.</p> <ul style="list-style-type: none"> • El ACP registra los asuntos de la reunión. • Cada integrante reporta su trabajo de acuerdo al rol y su trabajo de desarrollo al mismo tiempo. <p>Después de la reunión el LE:</p> <ul style="list-style-type: none"> • Distribuye el reporte de la reunión. • Anexa una copia en la carpeta del proyecto. 	
Paso	Actividades	Descripción

Datos semanales	<ul style="list-style-type: none"> • Se indica el total de horas utilizadas y las planeadas en la semana. • Se indica el total de horas acumuladas planeadas y actuales durante el ciclo de desarrollo. • Se indica el valor planeado y el valor actual ganado en la semana. • Se indica el valor planeado acumulado y el valor ganado acumulado durante el ciclo de desarrollo a la fecha. • Se indican las horas totales planeadas y actuales para las tareas terminadas durante el ciclo de desarrollo a la fecha.
Datos semanales de cada integrante del equipo	<p>Para cada integrante reporte:</p> <ul style="list-style-type: none"> • El tiempo total actual y planeado por cada ingeniero. • El valor planeado y ganado de cada ingeniero en la semana. • Las horas totales planeadas y trabajadas por cada ingeniero.
Datos semanales del equipo	<p>Para el reporte semanal del equipo registre:</p> <ul style="list-style-type: none"> • El tiempo total actual y planeado por el equipo. • El valor planeado y ganado por el equipo. • Las horas totales planeadas y trabajadas del equipo.
Tareas de desarrollo terminadas	<p>Para las tareas terminadas en la semana indique:</p> <ul style="list-style-type: none"> • El nombre de cada tarea. • El tiempo total actual y planeado para cada tarea. • Número de semana en que se planeó la tarea. • Valor ganado para la tarea.
Seguimiento de asuntos y riesgos	<ul style="list-style-type: none"> • Resuma el estado de asuntos y riesgos, así como cualquier cambio importante acontecido en la semana.
Otros elementos importantes	<ul style="list-style-type: none"> • Liste cualquier tarea realizada o evento ocurrido en la semana. • Ejemplos de roles que incluyan el estándar de código terminado, procedimiento de control de cambios aprobado, etc. • Ejemplos de desarrollo que incluyan diseño, codificación, inspección o pruebas de varios elementos del producto.

Tabla C.3 Tabla de instrucciones para llenar la forma SEMANA

unitarias		
Defectos de revisión de código/defectos de compilación	> 2.0	Sólo defectos mayores
Tasas en el tiempo de desarrollo		
Inspección de requerimientos/Tiempo en requerimientos	> 0.25	Sólo tiempo utilizado en requerimientos, no se asigna tiempo dedicado al estudio
Inspección en DAN/Tiempo en DAN	> 0.5	Sólo tiempo dedicado a diseño, no se asigna tiempo dedicado al estudio
DBN/Tiempo en codificación	> 1.00	
Revisión de DBN/Tiempo en DBN	> 0.5	
Revisión de código/Tiempo en codificación	> 0.5	
Tasas de revisión e inspección		
Páginas de los requerimientos/hora	< 2	Páginas de texto con interlineado sencillo
Páginas de DBN/hora	< 5	Lógica del diseño con formato
Líneas de texto de DBN/hora	< 100	Aproximadamente 3 LOC por cada línea pseudocódigo
LOC/hora	< 200	LOC lógicas
Tasas en inyección de defectos		
Defectos en requerimientos/hora	0.25	Sólo defectos mayores
Defectos en DAN/hora	0.25	Sólo defectos mayores
Defectos en DBN/hora	2.0	Sólo defectos de diseño
Defectos en código/hora	4.0	Sólo defectos mayores
Defectos en compilación/hora	0.3	Defectos señalados por el compilador
Defectos en pruebas unitarias/hora	0.2	Sólo defectos mayores
Tasas en eliminación de defectos		
Defectos en inspección de requerimientos/hora	0.5	Sólo defectos mayores
Defectos en inspección de DAN/hora	0.5	Sólo defectos mayores
Defectos en revisión de DBN/hora	2.0	Sólo defectos de diseño
Defectos en inspección de DBN/hora	0.5	Sólo defectos de diseño
Defectos en revisión de código/hora	6.0	Sólo defectos mayores
Defectos en inspección de código/hora	1.0	Sólo defectos mayores
Producción por fase		
Inspecciones de requerimientos	~ 70%	Sin contar comentarios
Revisiones e inspecciones de diseño	~ 70%	Usando análisis de estado y tablas de seguimiento
Revisiones e inspecciones de código	~ 70%	Usando listas de verificación personales
Compilación	~ 50%	Defectos de sintaxis en un 90% o más
Pruebas unitarias con 5 o menos defectos/KLOC	~ 90%	Para defectos mayores/KLOC: 50-75%
Construcción, integración, pruebas del sistema < 1.0 defectos/KLOC	~ 80%	Para defectos mayores/KLOC: 30-65%
Producción en el proceso		
Antes de la compilación	> 75%	Aplicando métodos de diseño
Antes de las pruebas unitarias	> 85%	Aplicando revisiones lógicas
Antes de la construcción e integración	> 97.5%	Para productos pequeños, 1 defecto máximo
Antes de las pruebas del sistema	> 99%	Para productos pequeños, 1 defecto máximo

Tabla C.5 Criterio estándar de calidad

A continuación se presenta la forma RESCA, en la cual se deberán indicar los valores correspondientes al plan de calidad del equipo.

Compilación		
Pruebas unitarias		
Construcción e integración		
Pruebas del sistema		
Tasas en eliminación de defectos (defectos/hora)		
Inspección de requerimientos		
Inspección en diseño de alto nivel		
Revisión en diseño detallado		
Inspección en diseño detallado		
Revisión en código		
Compilación		
Inspección en código		
Pruebas unitarias		
Construcción e integración		
Pruebas del sistema		
Producción por fase		
Inspección en requerimientos		
Inspección en diseño de alto nivel		
Inspección en diseño detallado		
Revisión en código		
Compilación		
Inspección en código		
Pruebas unitarias		
Construcción e integración		
Pruebas del sistema		
Producción en el proceso		
% antes de la compilación		
% antes de pruebas unitarias		
% antes de construcción e integración		
% antes de las pruebas del sistema		
% antes de la liberación del sistema		

Forma C.5 Forma para el Plan de Calidad, RESCA

A continuación se presenta la forma TAREA y posteriormente la forma CALENDARIO.

Nombre: _____ Fecha: _____
 Equipo: _____ Instructor: _____
 Ensamble: _____ Ciclo: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Fecha	Número	Tipo	Inyectado	Eliminado	Tiempo de modificación	Defecto modificado
<input type="text"/>	<input type="text"/>					

Descripción: _____

Forma C.9 Forma Registro de Defectos, REGD

	<p>Utilice esta forma para registrar el tiempo dedicado a cada tarea del proyecto</p> <ul style="list-style-type: none"> • Puede tener una bitácora para anotar la tarea y el elemento del producto o tener bitácoras separadas por cada tarea mayor. • Se registra el tiempo dedicado al proyecto. • Se registra el tiempo en minutos. • Sea lo más preciso posible. • Si se necesita más espacio, se utiliza otra forma.
General	

Nombre: _____ Equipo: _____ Instructor: _____
 Fecha: _____ No. ciclo: _____ No. semana: _____

Para cada rol, evalúe el porcentaje de trabajo requerido y la dificultad durante este ciclo		
Rol	Trabajo requerido	Dificultad en el rol
Líder de Equipo		
Administrador de Desarrollo		
Administrador de Planeación		
Administrador de Calidad y Proceso		
Administrador de Apoyo		
Contribución total (100%)		

Evaluación del equipo sobre los siguientes criterios. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Espíritu de equipo	1	2	3	4	5
Efectividad total	1	2	3	4	5
Experiencia adquirida	1	2	3	4	5
Productividad del equipo	1	2	3	4	5
Calidad del proceso	1	2	3	4	5
Calidad del producto	1	2	3	4	5

Evalúe la contribución de cada rol. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Apoyo	1	2	3	4	5

Evalúe cada rol de acuerdo al apoyo y ayuda proporcionado. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Apoyo	1	2	3	4	5

Evalúe cada rol de acuerdo a su desempeño. Encierre un número desde el 1 (bajo) hasta el 5 (alto)					
Líder de Equipo	1	2	3	4	5
Administrador de Desarrollo	1	2	3	4	5
Administrador de Planeación	1	2	3	4	5
Administrador de Calidad y Proceso	1	2	3	4	5
Administrador de Apoyo	1	2	3	4	5

Forma C.11 Forma Evaluación del Equipo por Colegas, EEC

• ¿Cuáles?

8. ¿En que porcentaje te basaste en el libro para saber cómo desarrollar las responsabilidades establecidas?

100% 50% 25% Otro ____

9. ¿En que porcentaje te basaste en las clases para saber cómo desarrollar las responsabilidades establecidas?

100% 50% 25% Otro ____

10. Después de haber generado los productos del primer ciclo, ¿cómo evalúas la dificultad de desarrollo en el segundo ciclo?

- | | | |
|--|---|---------|
| a) Fue más fácil porque ya se tenían conocimientos previos de las fases. | b) No hubo diferencia, tanto en el primero como en el segundo ciclo existió la misma dificultad | c) Otra |
| | | _____ |
| | | _____ |
| | | _____ |

11. ¿Cuál consideras que fue la razón de haber dedicado mayor tiempo de lo que se planeo a algunas fases?

12. ¿Qué eliminarías, agregarías o modificarías en TSPi?

13. ¿Tu nivel de inglés causó dificultad en asimilar el material del libro?

Si ____ No ____ Otra _____

14. Menciona otras causas que a tu juicio han afectado el desarrollo del curso.

15. ¿Fue el trabajo en equipo una experiencia que ha valido la pena aprender?

Si ____ No ____

- ¿Por qué?

16. Si tuvieras la oportunidad de participar en otro proyecto en equipo responderías lo siguiente:

Si, me agradó colaborar con otras personas	No, no es fácil colaborar con otras personas	Otra)

17. ¿Qué enseñanzas has adquirido con el trabajo en equipo?

18. Evalúa los conocimientos adquiridos en las áreas siguientes. Encierra un número para tu evaluación desde el 1 (más bajo) hasta el 5 (más alto).

Conocimientos	Numeración				
Métricas	1	2	3	4	5
Técnicas de documentación	1	2	3	4	5
Roles	1	2	3	4	5
Fases	1	2	3	4	5
Inspecciones y revisiones	1	2	3	4	5
Administración de Configuración	1	2	3	4	5