AAS2927

UNIVERSIDAD CATÓLICA ANDRES BELLO
VICERRECTORADO ACADÉMICO
ESTUDIOS DE POST-GRADO
ÁREA DE CIENCIAS ADMINISTRATIVAS Y DE GESTIÓN
POSTGRADO EN GERENCIA DE PROYECTOS

TRABAJO ESPECIAL DE GRADO

APLICACIÓN DEL MODELO ITERATIVO EN LA GESTIÓN DEL TIEMPO EN PROYECTOS DE ESTABILIZACIÓN DE SOFTWARE

Presentado por:
Márquez Fernández, Gregorio

Para optar el título de:
Especialista en Gerencia de Proyectos

Asesor:
Aponte Figueroa, Gloria María

Caracas, Mayo 2011.

UNIVERSIDAD CATÓLICA ANDRES BELLO
VICERRECTORADO ACADÉMICO
ESTUDIOS DE POST-GRADO
ÁREA DE CIENCIAS ADMINISTRATIVAS Y DE GESTIÓN
POSTGRADO EN GERENCIA DE PROYECTOS

APLICACIÓN DEL MODELO ITERATIVO EN LA GESTIÓN DEL TIEMPO EN PROYECTOS DE ESTABILIZACIÓN DE SOFTWARE

Autor: Márquez Fernández, Gregorio
Asesor: Aponte, Gloria
Año: 2011

RESUMEN

Hoy en día muchas empresas realizan puertas adentro desarrollo de software a su medida para cubrir necesidades específicas de la organización. Lo que usualmente se espera de un software productivo, es que éste funcione correctamente, pero muchas veces no tiene ese desempeño esperado, sino al contrario, por alguna razón se vuelve inestable: se enlentece, comienzan a fallar funcionalidades que antes trabajaban correctamente. Cuando se tiene la certeza o se llega a la conclusión de que se debe intervenir un software para llevarlo al nivel operativo que exige el negocio, debido a que por alguna razón se desestabilizó, y al no contarse con una modelo apropiado que permita abordar de manera organizada un *Proyecto de Estabilización de Software* para resolver la situación, se podría asegurar que se generará una complicación mayor para la empresa, debido a que las fallas, se podrían incrementar al entrar en ciclos infinitos de desarrollo, en búsqueda de la estabilización sin lograrlo, lo que hará que el tiempo a invertir haga al proceso de estabilización financieramente costoso, lento, y afectará el desempeño de las áreas que lo utilizan, y muchas veces sin lograr el objetivo de estabilizar el software. El objetivo principal de este trabajo de grado es aplicar un modelo iterativo de gestión del tiempo, para la gestión de un proyecto de estabilización de software, definir los criterios de estabilización en proyectos de software, analizar el método iterativo como referencia de uso en la gestión de tiempo en proyectos de software y formular las estrategias de aplicación del método iterativo en la gestión de tiempo en proyectos de software. Para lograr el objetivo, se hizo una adaptación del Modelo en Espiral de Boehm (1988), acoplándolo dentro de los procesos de la gestión del tiempo que propone el PMBOK (2008). Se espera entonces contar con un modelo iterativo que permita hacer una gestión del tiempo en proyectos de estabilización de software. La aplicación de este modelo iterativo, permite ir obteniendo en el tiempo versiones de software cada vez más estables, y en cada una de ellas, serán atendidas según prioridades, las fallas reportadas hasta que se llegue a una versión que declare al software estable.

PALABRAS CLAVE: Software Inestable, Modelo, Gestión del Tiempo, Prioridades, Software Estable.

Línea de Trabajo: Gestión del Tiempo

RECONOCIMIENTOS Y AGRADECIMIENTOS

Mis más sinceros agradecimientos y reconocimientos a quienes me acompañaron en todos estos estudios, aportaron conocimiento y acompañaron en todo tipo de momentos que vivimos en éste postgrado:

A la Universidad Católica Andrés Bello, la cual me brindó el conocimiento y estructura con la cual hoy en día cuento respecto a muchos aspectos no sólo relacionados a la Gerencia de Proyectos.

A Consein C.A., empresa que me apoyó financieramente con estos estudios, y le dio valor a esta iniciativa, involucrándome además en proyectos donde esta experiencia y conocimientos fueron aplicados.

A la profesora Gloria Aponte, su apoyo incondicional, y sus sugerencias oportunas me facilitaron el desarrollo de este trabajo especial de grado.

A mis queridísimos compañeros de clases de postgrado, juntos formamos una gran amistad, vivimos situaciones de las que mantengo recuerdos muy agradables, y nos dimos aliento y motivación en muchos momentos, especialmente en el desarrollo de éste casi final y muy importante hito, y por esto último, gracias especiales a Carolina, Adriana, Vanessa, Carola y Juan Carlos.

DEDICATORIA

Por ser tan pacientes, brindarme tanto apoyo, motivarme y sobre todo por ser tan compresivos en el transcurso de estos estudios de postgrado, el fruto de este trabajo va dedicado a Horten, Villo, Mahe, Adrián, Antonio y Carito. Muchas gracias, los quiero muchísimo.

LISTADO DE ACRÓNIMOS Y SIGLAS

- **CIV**, Colegio de Ingenieros de Venezuela.
- **IEEE Computer society**, Sociedad de Computistas del Instituto de Ingenieros Electricistas y Electrónicos.
- **PMBOK**, Guía de los fundamentos de La dirección de proyectos.
- **PMI**, Project Management Institute.
- **Prototipos**: Es la actividad de creación de prototipos de aplicaciones de software.
- **RAD**, Rapid Application Development.
- **SDLC**, Software Development Life Cicle.
- **SEI**, Software Engineering Institute.

ÍNDICE GENERAL

RESUMEN ii

RECONOCIMIENTOS Y AGRADECIMIENTOS iii

DEDICATORIA iv

LISTADO DE ACRÓNIMOS Y SIGLAS v

ÍNDICE GENERAL vi

ÍNDICE DE FIGURAS x

ÍNDICE DE TABLAS xi

INTRODUCCIÓN 1

CAPÍTULO I. PROPUESTA DE LA INVESTIGACIÓN 3

1.1. Planteamiento y delimitación de la problemática 3

Situación actual: Síntomas y causas 3

Pronóstico 6

Control del pronóstico 7

Formulación del problema 7

Sistematización del problema 7

1.2. Objetivos 8

Objetivo general 8

Objetivos específicos 8

1.3. Justificación 8

1.4. Alcance y delimitaciones 9

CAPÍTULO II. MARCO TEÓRICO 10

2.1. Antecedentes 10

2.2. Bases Teóricas 14

2.2.1. Gestión del Tiempo de los Proyectos 14

2.2.2. Desarrollos iterativos 55
2.2.3. Modelo Espiral 57
2.2.4. Asociación entre los procesos definidos por la Gestión del Tiempo PMBOK (2008) y el Modelo Espiral 62
2.2.5. Fases de un proyecto según Microsoft Solution Framework 63
2.2.6. Estructura Desagregada de Trabajo (EDT) 66
2.2.7. Bases Legales 67
2.3. Definiciones 68
CAPÍTULO III. MARCO METODOLÓGICO 70
3.1. Tipo de investigación 70
3.2. Diseño de la Investigación 70
3.3. Unidad de Análisis 71
3.4. Fases de la Investigación 71
Fase 1: El Problema 71
Fase 2: Revisión de propuestas existentes 71
Fase 3: Documentación 71
Fase 4: Análisis de la información 72
Fase 5: Desarrollo 72
3.5. Herramientas y técnicas de recolección de la información 72
3.6. Análisis de la información 73
3.7. Operacionalización de los Objetivos 75
3.8. Consideraciones Éticas 76
Colegio de Ingenieros de Venezuela, Código de Ética Profesional 76
Profesional del Project Management Institute, Código de Ética y Conducta 78
CAPÍTULO IV. VENTANA DE MERCADO 80

CAPÍTULO V. DESARROLLO 81

i. Revisión de la documentación de los procesos automatizados por el software 82

ii. Controlar la planificación de la iteración 83

ITERACIÓN 1 85

1. Recopilación de "puntos de atención" presentados o requerimientos no cubiertos 85

2. Evaluar y categorizar los "puntos de atención" detectados o requerimientos no cubiertos 85

3. Diagnóstico 86

4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional 88

5. Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos 89

6. Estimación de los tiempos de cada actividad 91

7. Identificar las actividades que se entregarán en cada versión 91

8. Desarrollo de la estabilización 92

9. Pruebas de calidad y aceptación por usuarios funcionales 95

10. Pase a producción de la versión 96

ITERACIÓN 2 A LA N 97

Criterios para finalizar del Proyecto de Estabilización del Software 98

Cómo se mide el costo según este modelo 98

Matriz de asignación de prioridades 99

Valores usados por cada campo de la Matriz de Prioridades 100

Criterio de Prioridades 100

Ejemplo de uso 101

CAPÍTULO VI. ANÁLISIS DE LOS RESULTADOS 102

Representación gráfica del modelo iterativo para la gestión del tiempo en proyectos de estabilización de software 104

Dinámica y desempeño aproximado de un proyecto de estabilización de software 105

CAPÍTULO VII. EVALUACIÓN DEL PROYECTO 108

CAPÍTULO VIII. CONCLUSIONES Y RECOMENDACIONES 110

Conclusiones 110

Recomendaciones 111

REFERENCIAS BIBLIOGRÁFICAS 113

ÍNDICE DE FIGURAS

Nº	Figura	Pág.
1:	Diagrama causa-efecto 1, empresas desarrolladoras que no utilizan buenas prácticas	4
2:	Diagrama causa-efecto 2, software que requiere mantenimiento por crecimiento	4
3:	Consecuencia de una incorrecta gestión del tiempo	6
4:	Panorama General de la Gestión del Tiempo del Proyecto	16
5:	Definir las Actividades: Entradas, Herramientas y Técnicas, y Salidas	17
6:	Diagrama de Flujo de Datos del Proceso Definir las Actividades	18
7:	Secuenciar las Actividades: Entradas, Herramientas y Técnicas, y Salidas	22
8:	Diagrama de Flujo de Datos del Proceso Secuenciar las Actividades	23
9:	Método de Diagramación por Precedencia	26
10:	Estimar los Recursos de las Actividades: Entradas, Herramientas y Técnicas, y Salidas	29
11:	Diagrama de Flujo de Datos del Proceso Estimar los Recursos de las Actividades	30
12:	Estimar la Duración de las Actividades: Entradas, Herramientas y Técnicas, y Salidas	35
13:	Diagrama de Flujo de Datos del Proceso Estimar la Duración de las Actividades	36
14:	Desarrollar el Cronograma: Entradas, Herramientas y Técnicas, y Salidas	41
15:	Diagrama de Flujo de Datos del Proceso Desarrollar el Cronograma	42
16:	Desarrollar el Cronograma: Entradas, Herramientas y Técnicas, y Salidas	50
17:	Diagrama de Flujo de Datos del Proceso Controlar el Cronograma	51
18:	Modelo Espiral de los procesos de software	57
19:	Modelo de proceso MSF	64
20:	Estructura Desagregada de Trabajo	66
21:	Ejemplo del uso de la Matriz de Prioridades	101
21:	Modelo iterativo para la gestión del tiempo en proyectos de estabilización de software.	104

22: Desempeño aproximado de un proyecto de estabilización de software..... 106

ÍNDICE DE TABLAS

Nº **Tabla** **Pág.**

1: Asociación entre los procesos definidos por la Gestión del Tiempo y el Modelo Espiral 62
2: Operacionalización de los objetivos. 75
3: Valores a utilizar en los campos de la matriz de prioridades..... 100
3: Asociación entre los procesos definidos por la Gestión del Tiempo del PMBOK (2008) y la Aplicación del Modelo Iterativo propuesto 102
4: Evaluación del cumplimiento del problema 108
5: Evaluación del cumplimiento de los objetivos. 109

INTRODUCCIÓN

El contexto donde se pretende implementar la presente aplicación, es dentro de empresas que desarrollan software a la medida, o empresas prestadoras de servicios de desarrollo de software; donde por distintas razones (que se expondrán más adelante), pierden el control del desempeño y funcionalidades del software que han desarrollado, presentando éste cada vez más fallas hasta ser declarado inestable.

El propósito general de éste trabajo especial grado, es aplicar un modelo iterativo que permita gestionar eficientemente el tiempo en proyectos de estabilización de software apoyado en un modelo iterativo.

El estado del arte que se pretende alcanzar, es el de un desempeño muy alto en la gestión de proyectos de éste tipo, con resultados de alto impacto en la organización que aplique este modelo.

La motivación principal para desarrollar este tema, se basa en la experiencia personal enfrentando proyectos de éste tipo en diferentes empresas nacionales y trasnacionales durante 10 años.

Este trabajo se apoya en el Modelo de Espiral propuesto por Boehm (1988), pero no se implementa literalmente, se propone la aplicación de un modelo orientado a prioridades, mas no netamente a riesgos como se enfoca el Modelo de Espiral (debido la taxonomía específica de los proyectos que en que se usará). Se toma del modelo espiral, lo necesario para aplicar este modelo, haciendo énfasis en la gestión del tiempo como lo propone el PMBOK (2008).

Esta propuesta de aplicación, propone las actividades que deben realizarse en un proyecto de éste tipo y las organiza como procesos en el tiempo según lo indica el PMBOK en el capítulo dedicado a la gestión del tiempo, siguiendo una secuencia lógica

de procesos dentro de iteraciones incrementales para poder hacer entregas de versiones de software estables para ser puestas en producción a lo largo del proyecto.

El presente documento está conformado por los siguientes capítulos:

En el Capítulo I se encuentra la Propuesta de la Aplicación del Modelo Iterativo, la definición del problema y la justificación del mismo definiendo objetivos.

El Capítulo II, se enfoca en las bases teóricas en las que se apoya este trabajo

Capítulo III, contempla cómo seguir la aplicación del modelo para obtener los resultados deseados.

El Capítulo IV, expone la ventana de mercado para esta propuesta, área potencial para su uso, así como costos asociados a la implementación del mismo.

El Capítulo V es la parte central y más importante, ya que allí se hace el planteamiento del modelo a ser aplicado para dar solución al problema.

En los Capítulos del VI al VIII, se analizan los resultados del trabajo, la evaluación del cumplimiento de los objetivos, y finalmente conclusiones y recomendaciones.

Finalmente se presentan las referencias bibliográficas y los anexos.

CAPÍTULO I. PROPUESTA DE LA INVESTIGACIÓN

1.1. Planteamiento y delimitación de la problemática

Situación actual: Síntomas y causas

Algunas empresas que desarrollan software a su propia medida y que no cuentan con la experiencia de gestión de proyectos, metodologías, o buenas prácticas, siguen la tendencia a desarrollar software que se desestabiliza por distintas causas:

- Diseño inicial que no toma en cuenta el crecimiento en cuanto a funcionalidades, e integración con otros sistemas.
- Malas prácticas de desarrollo de software.
- Cliente funcional indeciso, que no define bien sus requerimientos iniciales, o que no estaba claro cuando estaba haciendo el requerimiento.
- No se documentaron los requerimientos iniciales (muchas veces no es firmado por las partes interesadas)
    - Cambios de alcance decididos por los desarrolladores o por los usuarios funcionales y que no fueron aprobados (ni documentados).
- El proyecto de desarrollo de software no pasó por algunas de las siguientes fases del proyecto:
    - Pruebas unitarias
    - Pruebas de integración y calidad
    - Pruebas funcionales (donde se da el visto bueno antes de pasar a producción el software)
- Falta de experiencia del Líder/Gerente de proyecto, o ausencia del mismo.
- Asignación de desarrolladores con un nivel técnico no adecuado para afrontar el proyecto.
- Presiones del negocio para que el software sea implementado antes de tiempo.
- Pérdida de la funcionalidad de algún requisito, al haber un reemplazo del desarrollador que lo atiende (por cualquier motivo)

- Adición de nuevas funcionalidades a lo largo de la vida del software para las que no fue diseñado desde un inicio.
- Falta de mantenimiento, entonación y rediseño de Bases de Datos (hardware y software).

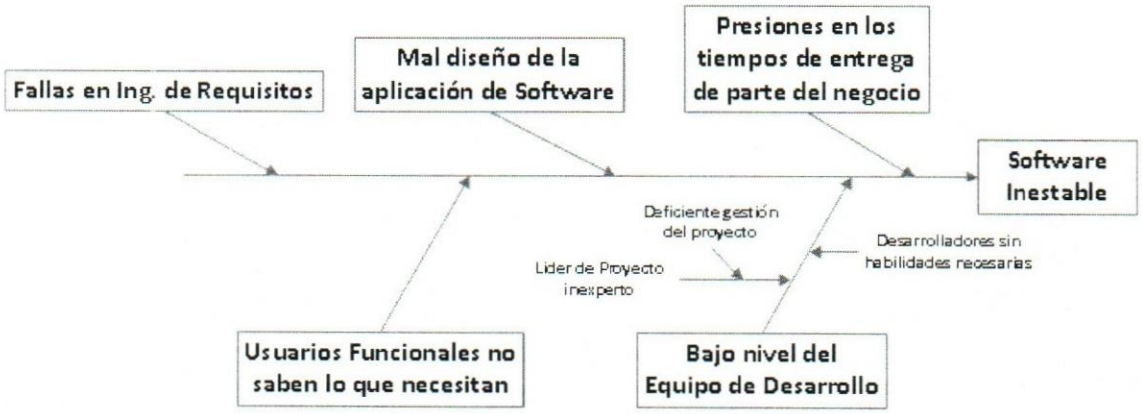

Figura 1: Diagrama causa-efecto 1, empresas desarrolladoras que no utilizan buenas prácticas

Fuente: Gregorio Márquez (Propia)

Otro de los posibles escenarios que se presentan, es que el software no fue diseñado para escalar en funcionalidades, y al adicionar más de estas, tiende a desestabilizarse en el tiempo.

Figura 2: Diagrama causa-efecto 2, software que requiere mantenimiento por crecimiento

Fuente: Gregorio Márquez (Propia)

Los síntomas podrían irse presentando en el tiempo, empezando con pocas fallas hasta hacerse completamente inestable. Algunos de los diversos síntomas que podrían presentarse serian:

- Lentitud en el sistema
- Aumento importante de fallas reportadas
- Excepciones recurrentes
    - "Crash" del sistema
- No ejecuta las funcionalidades requeridas y acordadas
- Se comienzan a identificar necesidades de funcionalidades que no fueron previstas al inicio del proyecto
- Problemas de seguridad
    - Autenticación y/o autorización
    - Acceso por usuarios no autorizados a datos sensibles
    - No existen capacidades de auditorías de funciones críticas
- Interfaces de usuario ineficientes
- Bajo rendimiento por hardware (CPU, RAM, Disco Duro, etc)

Cuando se tiene la certeza o se llega a la conclusión de que se debe intervenir el software para llevarlo al nivel operativo que exige el negocio, el problema que se presenta es mayor; ya que al no haberse seguido metodologías, o por alguna razón no se adoptaron buenas prácticas de gerencia de proyectos, estabilizar el software quizás llegue a ser muy complicado. Bennatan (2006), define este tipo de proyectos como: Un proyecto de desarrollo de software pesadilla es: un proyecto que se encuentra en una espiral rápida que está fuera de control... o simplemente ya es un desastre. Las técnicas convencionales de gestión de proyectos no van a salvar estos proyectos: no hay rescate estándar de los procesos a seguir. Usted necesita hacer algo radicalmente diferente

Al no contarse con un modelo apropiado que permita atacar de manera organizada y metódica al problema, se podría asegurar que se generaría un problema mayor para la

empresa, ya que mientras se resuelven fallas por un lado, podrían generarse nuevas por otro; mientras se resuelven algunas fallas en una entrega, podrían reabrirse otras que ya se habían resuelto, entrando en ciclos infinitos sin llegar a estabilizar el software, haciendo que quizás se desestabilice aún más, lo que hará que el tiempo a invertir haga el sistema financieramente costoso, y operativamente un problema.

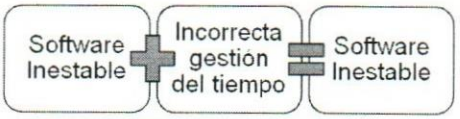

Figura 3: Consecuencia de una incorrecta gestión del tiempo

Fuente: Gregorio Márquez (Propia)

Todos los inconvenientes y escenarios anteriormente planteados afectan a la operatividad correcta del software, y por ende a la empresa, produciendo pérdidas de tiempo y dinero a las áreas de la organización que requieren la aplicación en cuestión, por lo que amerita una eficiente gestión del tiempo para resolver la situación. De esta manera se concluye que el problema se debe a la ausencia de un modelo a seguir para la gestión del tiempo que en proyectos de estabilización de software, permita ir haciendo entregas de versiones, incrementando funcionalidades hasta lograr la estabilización.

Pronóstico

Al no contarse con un modelo apropiado para la gestión del tiempo, en el momento de intentar estabilizar el software podría pronosticarse lo siguiente:

- Al no realizarse una correcta asignación de prioridades para la resolución de las fallas, el equipo podría seguir el camino más largo para estabilizar la aplicación.
- No se identificarán las secuencias correctas de las tareas según la criticidad y prioridad de las mismas, dejándose las funcionalidades más importantes para entregas posteriores.

- La estimación del tiempo que puede llevar desarrollar las actividades, tendrán una desviación negativa muy alta para el proyecto.
- Habrá una falta de equilibrio en la asignación de tareas, causando sobrecarga de trabajo o subutilización de los miembros del equipo.
- No existirá un control y seguimiento eficiente de las actividades.
- No se podrá generar indicadores de rendimiento reales.
- Afectará la rentabilidad de la empresa.
- Subutilización de recursos materiales (como equipos y software)
- Insatisfacción del cliente (bien sea interno o externo), por no cumplir con tiempos, costos y sobre todo calidad.
- Equipo de desarrolladores desmotivados

Control del pronóstico

Este tipo particular de proyectos obliga a desarrollar o adoptar una modelo iterativo para la gestión del tiempo de proyectos de estabilización de software, que siga un modelo iterativo, que permitirá al equipo de desarrollo poder ir haciendo entregas de manera incremental (versiones), de la aplicación a estabilizar cada vez con menos fallas.

Formulación del problema

¿Cuál es la secuencia lógica de procesos que pueden seguirse para lograr una efectiva gestión del tiempo en proyectos de estabilización de software?

Sistematización del problema

- ¿Se le están dando las prioridades correspondientes a las funcionalidades a reparar o desarrollar?

- ¿Se está desarrollando una versión estable única que agrupe todas las funcionalidades requeridas, o se ha planteado la aplicación de un modelo para entrega de versiones?
- ¿Se cuenta con una propuesta de aplicación de un modelo definido que permita hacer una correcta gestión del tiempo de proyectos de este tipo?
- ¿Cómo debería ser la dinámica y desempeño de un proyecto de estabilización de software?

1.2. Objetivos

Objetivo general

Aplicar el modelo iterativo en la gestión del tiempo en proyectos de estabilización de software

Objetivos específicos

- Definir los criterios de estabilización en proyectos de software.
- Analizar el método iterativo como referencia de uso en la gestión de tiempo en proyectos de software.
- Formular las estrategias de aplicación del método iterativo en la gestión de tiempo en proyectos de software.

1.3. Justificación

La necesidad de poder encarar un proyecto de éste tipo, tener un control eficiente de la asignación de tareas, seguimiento y control de las mismas, establecer prioridades en cuanto a funcionalidades a entregar en cada versión, para poder contar en el menor tiempo posible con el software estable, genera la necesidad de aplicar un modelo iterativo de gestión de proyectos tomando lo necesario de las mejores prácticas existentes que cubran esos aspectos.

Se contará entonces con la propuesta de aplicación de un modelo que permitirá llevar una gestión del tiempo eficiente para proyectos de estabilización de software.

1.4. Alcance y delimitaciones

- Esta propuesta de aplicación de modelo iterativo, aunque puede extenderse a otras fases[1], va dirigida a ser implementada en la dinámica que representa la "Fase de Desarrollo" de software.

- El detalle de las distintas metodologías a seguir en cada una de las fases de desarrollo que se citen, no se son cubiertas en esta propuesta (por ejemplo, Documentación de Procesos, que es una sub-tarea de la fase de Ing. de Requisitos).

- Esta investigación no cubre en detalle lo relacionado con técnicas de procura, presupuesto y selección de personal.

- Las diversas técnicas y herramientas específicas que podrían utilizarse en ciertos casos que existen en el mercado, no fueron abordadas, ya que quedan a criterio de quien aplicar el modelo iterativo.

[1] Fases: "Visión y Alcance", "Planificación", "Ingeniería de Requisitos", "Diseño", "Desarrollo", "Pruebas Unitarias", "Pruebas Funcionales", "Puesta en Producción" y "Soporte Post Puesta en Producción"

CAPÍTULO II. MARCO TEÓRICO

2.1. Antecedentes

A continuación se presentan los trabajos de grado más relevantes relacionados con la investigación.

Méndez, E. (2006), en su investigación cuyo título es "Diseño de un modelo de evaluación de metodologías para el desarrollo de software"; cuyo objetivo general: Diseñar un modelo de evaluación de metodologías para el desarrollo de software, y los objetivos específicos: Identificar variables para la evaluación de metodologías aplicadas en el desarrollo de software, diseñar el modelo de evaluación de metodologías de desarrollo de software y desarrollar la matriz de evaluación de metodologías; y mediante una investigación tipo investigación y Desarrollo; plantea los pasos a seguir para llevar a cabo un proyecto de desarrollo de software. Obtiene como producto una herramienta o modelo a seguir para la selección de una metodología que puede ser adecuada a las características de cualquier proyecto de desarrollo de software. El aporte más importante de dicho trabajo es la recopilación de las diferentes metodologías utilizadas para el desarrollo de software, convencionales o de desarrollo ágil que servirá como guía para conocer las características más importantes de cada una de ellas.

Merchán, N (2006), en su trabajo de grado Diseño de un proceso de generación de estimados de tiempo para una empresa de Desarrollo de aplicaciones, cuyo objetivo general es Diseñar el proceso de generación de estimaciones de tiempo requerido para desarrollar aplicaciones web, adaptado a la naturaleza de Adverweb; cuyo enfoque se orientó a establecer una metodología basada en tres fases, que consistió en desarrollar una investigación de los métodos de estimación sugeridos por la bibliografía para proyectos de Tecnología de Información (IT), llevar a cabo un estudio de la manera

actual de estimar en la compañía objeto del estudio, y realizar el diseño del proceso a proponer. Se generó el diseño del proceso basado en el Modelo CMMI para Ingeniería de Software representación por etapas, enfatizando en optimizar las debilidades encontradas en la manera que se generan los estimados actualmente. Los resultados obtenidos presentan un producto que puede ser el comienzo del desarrollo de muchos otros aspectos que contribuyan a fortalecer y formalizar los aspectos de gerencia de proyecto en la empresa en cuestión. El aporte de dicho trabajo está presente en la revisión de las distintas metodologías existentes que se presentan, así como el resultado del estudio, que evidencia que dichas metodologías se enfocan a proyectos que se ejecutan desde el inicio de una idea, manteniendo un control dentro de un marco metodológico bien definido y maduro, pero no abordan el caso particular de proyectos para la estabilización de un software, lo cual motiva a escribir al respecto.

Talavera, M. (2006), cuyo investigación se tituló aplicación de la gestión del tiempo para los proyectos de una empresa desarrolladora de software, cuyo Objetivo principal es Proponer la aplicación de la Gestión de Tiempo basada en el Project Management Institute para los proyectos de una empresa desarrolladora de software; cuya finalidad es proponer las más adecuadas herramientas, técnicas y procedimientos de gestión del tiempo de proyectos, con los cuales puedan realizar la planificación de los mismos así como su posterior seguimiento y control. Contribuyendo de esta manera a que la empresa pueda mantenerse competitiva dentro del segmento de mercado al cual pertenece y logre sus objetivos, minimizando los riesgos de incurrir en sobre costos, disminución de alcance e incremento del tiempo requerido para la entrega de los productos. El Aporte de dicho trabajo resalta en la importancia que tiene para las empresas la gestión del tiempo, y la preocupación de éstas por contar con herramientas y técnicas para gerenciar las tareas.

Páez, C. (2008), cuyo investigación se tituló Plan de mejora del desempeño de la gestión del tiempo en los proyectos del departamento de servicios de tecnología de Hewlett-Packard Venezuela, cuyo Objetivo principal es evaluar los procesos de la gestión del tiempo aplicados actualmente en los proyectos de Tecnología de Información (TI) de Hewlett-Packard Venezuela (HP). Actualmente, el departamento de Tecnología de Servicios de HP está enfrentado problemas para completar los proyectos en los tiempos planificados, trayendo como consecuencia el aumento de los costos no contemplados, sobrecarga de trabajo en los ingenieros especialistas y gerentes de proyectos y aumento de inconformidad de los clientes. A tal efecto, surgió la necesidad de desarrollar un plan de mejora que plantee nuevas alternativas para mejorar el desempeño de la gestión de los proyectos. Para lograr los objetivos planteados, se estableció una metodología basada en fases que consistió en colectar información de los proyectos ejecutados, revisión de las herramientas y recursos que dispone localmente el departamento, y la identificación de las brechas que existen entre la realidad y el ideal deseado basado en las mejores prácticas de gerencia de proyectos del "Project Management Institute" (PMI), que permita a la empresa mejorar dicha gestión en sus proyectos de TI. La investigación a realizar es del tipo proyectiva, o también conocida como proyecto factible. El Aporte de dicho trabajo resalta en la toma de decisiones respecto al problema de la gestión del tiempo, el aseguramiento del logro de los objetivos a tiempo, el énfasis en la integración de todo el equipo para la toma de decisiones, y la importancia que tiene para las empresas aplicar correctivos cuando los proyectos salen de control.

Barry, B. (1998), en su trabajo de grado Un modelo espiral de desarrollo de software y de mejoramiento, cuyo objetivo general es eejemplificar el debate sobre los modelos de procesos de software basados en ciclos de vida; cuyo desarrollo está enfocado en un modelo en espiral utiliza un enfoque cíclico para el desarrollo de sistemas mediante la definición del sistema. Esto da lugar a versiones diferentes. Permite al administrador de proyecto construir una prueba de concepto para el cliente desde el principio. El modelo es acerca de prototipos rápidos, y se centra en el desarrollo evolutivo, utilizando el

enfoque de modelo en cascada para cada fase del proyecto. Con este enfoque, el detalle de todo el sistema no está totalmente definido en el primer intento. En cambio, los desarrolladores evalúan las características por la más alta prioridad, definen su aplicación, y luego obtienen información de retroalimentación de los clientes, lo cual expone un desarrollo incremental de predicción. Basado en ese proceso de flujo, el equipo va y comienza a definir e implementar características adicionales. Un modelo en espiral es mucho más que una metodología basada en el riesgo, porque el riesgo se evalúa antes de cada fase y se volverán a analizar en los ciclos siguientes. A veces, los directores de proyectos consideran un enfoque en espiral como un enfoque de solución rápida pero hay mucho más que eso. Una estructura todavía tiene que ser aplicada a cualquier proyecto de RAD. El modelo en espiral es más difícil que la metodología de cascada, pero con la práctica, su técnica y el proceso es más fácil. Una metodología iterativa se inclina más hacia la forma, los desarrolladores / diseñadores de trabajo, en lugar de la forma en que un gerente de proyecto trabaja. Hay muchas variantes en el modelo en espiral y avances aún mayor se hará en el futuro, pero este método funciona bien ahora. El aporte de este trabajo está centrado en la definición de las bases de una metodología cíclica-incremental, que puede ser adaptada a distintos tipos de proyectos de desarrollo de software, definiendo los entregables en base al análisis de riesgos.

2.2. Bases Teóricas

En el siguiente apartado se presentan las bases teóricas que sustentan esta investigación.

2.2.1. Gestión del Tiempo de los Proyectos

Según PMI (2008), la Gestión del Tiempo del Proyecto incluye los procesos requeridos para administrar la finalización del proyecto a tiempo. La figura 4 proporciona un panorama general de los procesos de Gestión del Tiempo del Proyecto a saber:

Definir las Actividades Es el proceso que consiste en identificar las acciones específicas a ser realizadas para elaborar los entregables del proyecto.

Secuenciar las Actividades Es el proceso que consiste en identificar y documentar las interrelaciones entre las actividades del proyecto.

Estimar los Recursos de las Actividades Es el proceso que consiste en estimar el tipo y las cantidades de materiales, personas, equipos o suministros requeridos para ejecutar cada actividad.

Estimar la Duración de las Actividades Es el proceso que consiste en establecer aproximadamente la cantidad de períodos de trabajo necesarios para finalizar cada actividad con los recursos estimados.

Desarrollar el Cronograma Es el proceso que consiste en analizar la secuencia de las actividades, su duración, los requisitos de recursos y las restricciones del cronograma para crear el cronograma del proyecto.

Controlar el Cronograma Es el proceso por el que se da seguimiento al estado del proyecto para actualizar el avance del mismo y gestionar cambios a la línea base del cronograma.

El desarrollo del cronograma utiliza las salidas de los procesos Definir las Actividades, Secuenciar las Actividades, Estimar los Recursos de las Actividades y Estimar la

Duración de las Actividades. La mayor parte del esfuerzo en el área de conocimiento de la Gestión del Tiempo del Proyecto se realizará durante el proceso Controlar el Cronograma para asegurar que el trabajo del proyecto se complete de manera oportuna.

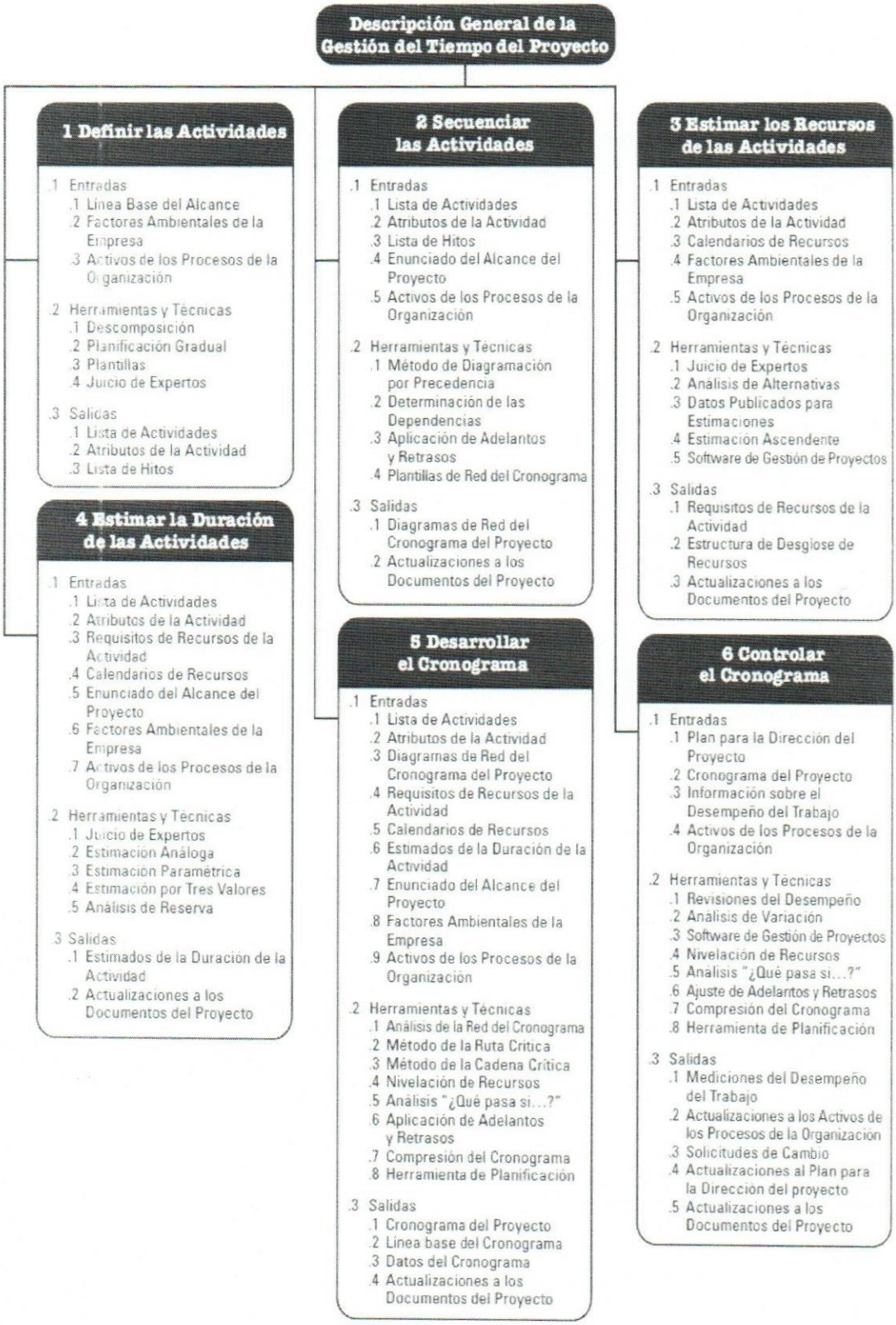

Figura 4: Panorama General de la Gestión del Tiempo del Proyecto
Fuente (PMI, 2008)

A continuación se pasa a describir en detalle cada uno de los procesos de gestión del tiempo propuestos en el PMI (2008).

Definir las Actividades

Definir las Actividades es el proceso que consiste en identificar las acciones específicas a ser realizadas para elaborar los entregables del proyecto. El proceso Crear la EDT identifica los entregables en el nivel más bajo de la estructura de desglose del trabajo (EDT), denominado paquetes de trabajo. Los paquetes de trabajo del proyecto se descomponen normalmente en componentes más pequeños llamados actividades, que representan el trabajo necesario para completar los paquetes de trabajo. Las actividades proporcionan una base para la estimación, planificación, ejecución, seguimiento y control del trabajo del proyecto. La definición y la planificación de las actividades del cronograma están implícitas en este proceso, de modo que se cumplan los objetivos del proyecto.

Figura 5: Definir las Actividades: Entradas, Herramientas y Técnicas, y Salidas
Fuente: (PMI, 2008).

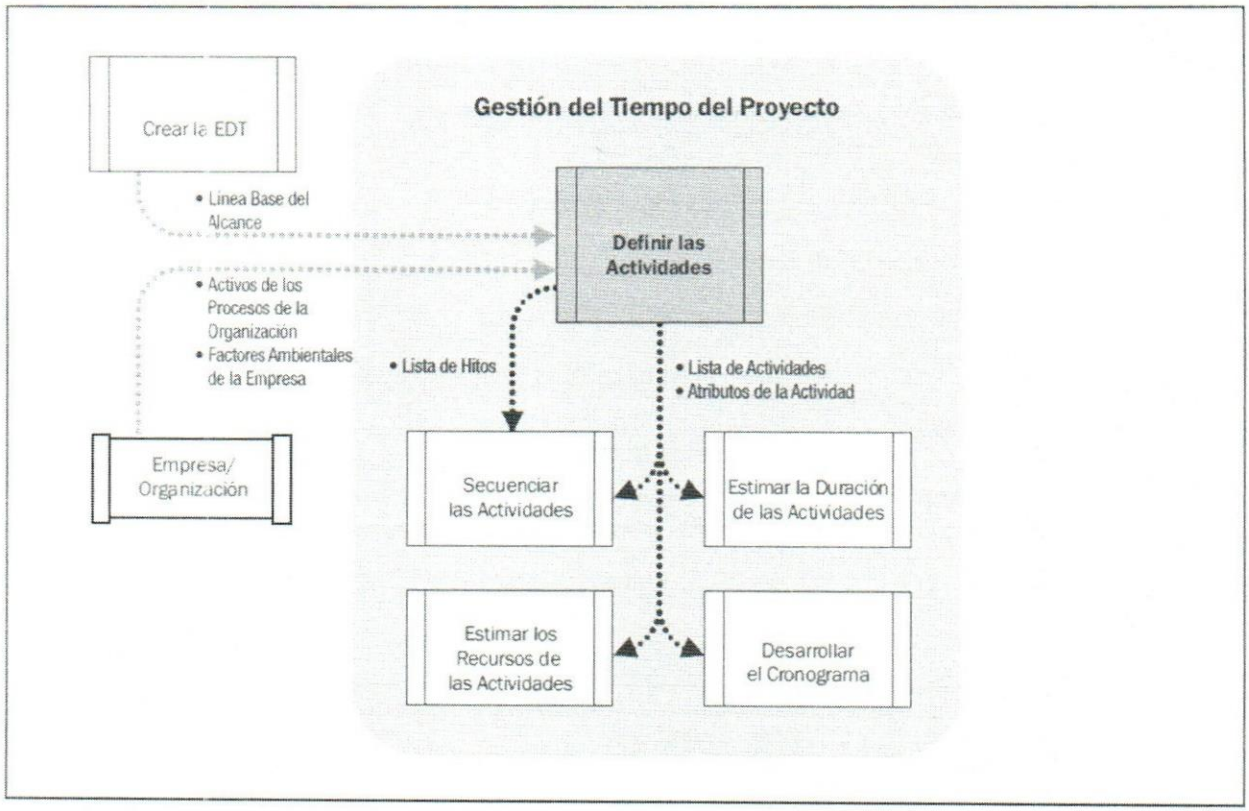

Figura 6: Diagrama de Flujo de Datos del Proceso Definir las Actividades
Fuente: (PMI, 2008)

Definir las Actividades: entradas

.1 Línea Base del Alcance

Los entregables, restricciones y supuestos del proyecto que están documentados en la línea base del alcance del proyecto deben considerarse explícitamente al definir las actividades que incluyen:

- **La declaración del alcance del proyecto.** La declaración del alcance del proyecto incluye la descripción del alcance del producto y los entregables del proyecto, y define los criterios de aceptación establecidos por el usuario del producto.

• **La EDT.** La EDT define cada entregable y su descomposición en paquetes de trabajo.

• **El diccionario de la EDT.** El diccionario de la EDT contiene una descripción detallada del trabajo y documentación técnica acerca de cada elemento de la EDT.

.2 Factores Ambientales de la Empresa

Entre los factores ambientales de la empresa que pueden influir en el proceso Definir las Actividades, se encuentra el sistema de información de la gestión de proyectos (PMIS).

.3 Activos de los Procesos de la Organización

Los activos de los procesos de la organización que pueden influir en el proceso Definir las Actividades incluyen, entre otras:

- Las políticas, procedimientos y lineamientos existentes, ya sean formales o informales relacionados con la planificación de las actividades, tales como la metodología de planificación, que se consideran en la definición de las actividades.
- La base de conocimiento de lecciones aprendidas que contiene información histórica relativa a las listas de actividades utilizadas en proyectos anteriores similares.

Definir las Actividades: Herramientas y Técnicas

.1 Descomposición

La técnica de descomposición, tal como se aplica para definir las actividades, consiste en subdividir los paquetes de trabajo del proyecto en componentes más pequeños y más fáciles de manejar, denominados actividades. Las actividades representan el esfuerzo necesario para completar un paquete de trabajo. El proceso Definir las Actividades define las salidas finales como actividades y no como entregables, como se realiza en el proceso Crear la EDT.

La lista de actividades, la EDT y el diccionario de la EDT pueden elaborarse de manera secuencial o simultánea, usando la EDT y el diccionario de la EDT como base para el desarrollo de la lista final de actividades. Cada paquete de trabajo dentro de la EDT se descompone en las actividades necesarias para producir los entregables del paquete de trabajo. La participación de los miembros del equipo en la descomposición puede conducir a obtener resultados mejores y más precisos.

.2 Planificación Gradual

La planificación gradual es una forma de planificación mediante elaboración gradual, donde se planifica en detalle el trabajo que debe desarrollarse en el corto plazo y el trabajo futuro se planifica a un nivel superior de la EDT. Por lo tanto, dependiendo de su ubicación en el ciclo de vida del proyecto, el trabajo puede existir en diferentes niveles de detalle. Por ejemplo, durante la planificación estratégica temprana, donde la información está menos definida, los paquetes de trabajo pueden descomponerse a nivel de hitos. Conforme se conoce más acerca de los próximos eventos en el corto plazo, pueden descomponerse en actividades.

.3 Plantillas

Una lista de actividades estándar o una parte de una lista de un proyecto previo, puede utilizarse a menudo como plantilla para un nuevo proyecto. La información relacionada con los atributos de las actividades de las plantillas también puede incluir otra información descriptiva útil para la definición de las actividades. Las plantillas también pueden utilizarse para identificar hitos típicos del cronograma.

.4 Juicio de Expertos

Los miembros del equipo del proyecto u otros expertos con experiencia y habilidad en el desarrollo de declaraciones de alcance del proyecto detalladas, la EDT y los cronogramas del proyecto, pueden aportar su experiencia para definir las actividades.

Definir las Actividades: Salidas

.1 Lista de Actividades

La lista de actividades es una lista exhaustiva que abarca todas las actividades del cronograma necesarias para el proyecto. La lista de actividades incluye el identificador de la actividad y una descripción del alcance del trabajo para cada actividad, con el nivel de detalle suficiente para que los miembros del equipo del proyecto comprendan el trabajo que deben realizar.

.2 Atributos de la Actividad

Los atributos de la actividad amplían la descripción de la actividad, identificando los múltiples componentes relacionados con cada una de ellas. Los componentes de cada actividad evolucionan con el tiempo. Durante las etapas iniciales del proyecto, estos atributos incluyen el identificador de la actividad, el identificador de la EDT y el nombre de la actividad, y una vez terminado, pueden incluir los códigos de la actividad, la descripción de la actividad, las actividades predecesoras, las actividades sucesoras, las relaciones lógicas, los adelantos y los retrasos, los requisitos de recursos, las fechas impuestas, las restricciones y los supuestos. Los atributos de la actividad pueden usarse para identificar a la persona responsable de ejecutar el trabajo, la zona geográfica o el lugar donde debe realizarse el trabajo y el tipo de actividad, tal como nivel de esfuerzo, esfuerzo discreto y esfuerzo prorrateado. Los atributos de la actividad se utilizan para el desarrollo del cronograma y para seleccionar, ordenar y clasificar las actividades del cronograma planificadas de diferentes maneras dentro de los informes. La cantidad de atributos varía según el área de aplicación.

.3 Lista de Hitos

Un hito es un punto o evento significativo dentro del proyecto. Una lista de hitos identifica todos los hitos e indica si éstos son obligatorios, como los exigidos por contrato, u opcionales, como los basados en la información histórica.

Secuenciar las Actividades

Secuenciar las Actividades es el proceso que consiste en identificar y documentar las relaciones entre las actividades del proyecto. La secuencia de actividades se establece mediante relaciones lógicas. Cada actividad e hito, a excepción del primero y del último, se conecta con al menos un predecesor y un sucesor. Puede ser necesario incluir adelantos o retrasos entre las actividades para poder sustentar un cronograma del proyecto realista y viable. La secuencia puede establecerse utilizando un software de gestión de proyectos o empleando técnicas manuales o automatizadas.

Figura 7: Secuenciar las Actividades: Entradas, Herramientas y Técnicas, y Salidas
Fuente: (PMI, 2008)

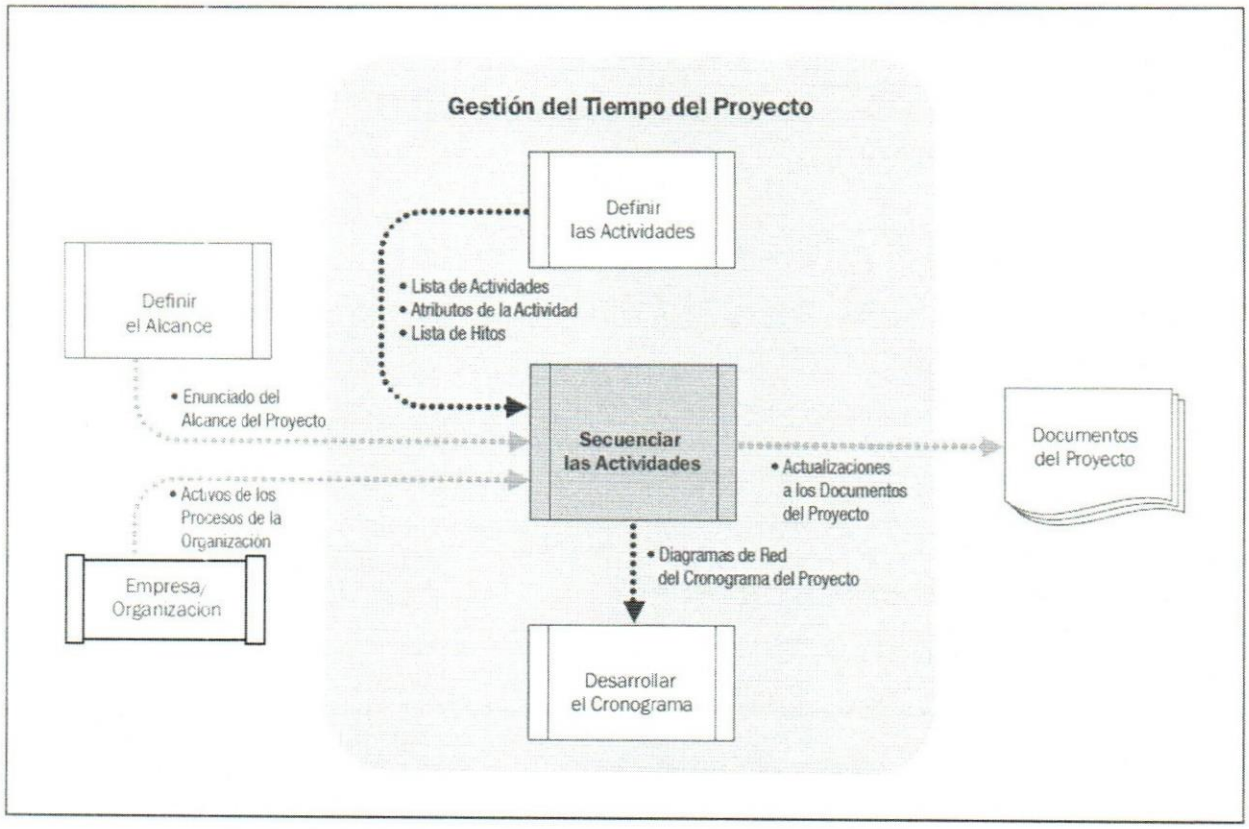

Figura 8: Diagrama de Flujo de Datos del Proceso Secuenciar las Actividades
Fuente (PMI, 2008)

Secuenciar las Actividades: Entradas

.1 Lista de Actividades

Descrita en el punto Definir las Actividades de Salida de la parte de Herramientas y Técnicas.

.2 Atributos de la Actividad

Descritos en Definir las Actividades de Salida de la parte de Herramientas y Técnicas. Los atributos de la actividad pueden describir una secuencia necesaria de eventos o definir relaciones predecesoras o sucesoras.

.3 Lista de Hitos

La lista de hitos puede incluir fechas programadas para hitos específicos.

.4 Declaración del Alcance del Proyecto

La declaración del alcance del proyecto contiene la descripción del alcance del producto, que incluye las características del producto que pueden afectar el establecimiento de la secuencia de las actividades, tal como la disposición física de una planta que se va a construir o las interfaces del subsistema en un proyecto de software. Aunque estos efectos a menudo son visibles en la lista de actividades, por lo general la descripción del alcance del producto se revisa para corroborar su exactitud.

.5 Activos de los Procesos de la Organización

Entre los activos de los procesos de la organización que pueden influir en el proceso Secuenciarlas Actividades, se encuentran los archivos de proyecto provenientes de la base de conocimiento de la empresa, que se utilizan en la metodología de planificación.

Secuenciar las Actividades: Herramientas y Técnicas

.1 Método de Diagramación por Precedencia (PDM)

El método de diagramación por precedencia (PDM) es utilizado en el método de la ruta crítica (CPM) para crear un diagrama de red del cronograma del proyecto que utiliza casillas o rectángulos, denominados nodos, para representar las actividades, que se conectan con flechas que muestran sus relaciones lógicas. La Figura 9 muestra un diagrama de red simple del cronograma del proyecto, elaborado utilizando el método de diagramación por precedencia.

Esta técnica también se denomina actividad en el nodo (AON) y es el método utilizado por la mayoría de los paquetes de software de gestión de proyectos.

El método de diagramación por precedencia incluye cuatro tipos de dependencias o relaciones lógicas.

- **Final a Inicio (FI).** El inicio de la actividad sucesora depende de la finalización de la actividad predecesora.
- **Final a Final (FF).** La finalización de la actividad sucesora depende de la finalización de la actividad predecesora.
- **Inicio a Inicio (II).** El inicio de la actividad sucesora depende del inicio de la actividad predecesora.
- **Inicio a Final (IF).** La finalización de la actividad sucesora depende del inicio de la actividad predecesora.

El tipo de relación de precedencia final a inicio es el más comúnmente utilizado por el método de diagramación por precedencia. La relación inicio a final se usa esporádicamente, pero se incluye aquí para proporcionar una lista completa de los tipos de relaciones de este método.

.2 Determinación de Dependencias

Para definir la secuencia entre las actividades, se emplean tres tipos de dependencias:

- **Dependencias obligatorias**. Las dependencias obligatorias son aquéllas requeridas por contrato, o inherentes a la naturaleza del trabajo. El equipo del proyecto determina qué dependencias son obligatorias durante el proceso de establecimiento de la secuencia de las actividades. Las dependencias obligatorias a menudo implican limitaciones físicas, como en un proyecto de construcción, donde es imposible erigir la superestructura hasta tanto no se construyan los cimientos; o en un proyecto de electrónica, donde se debe construir el prototipo antes de poder probarlo. A veces se utiliza la expresión "lógica dura" para referirse a las dependencias obligatorias.

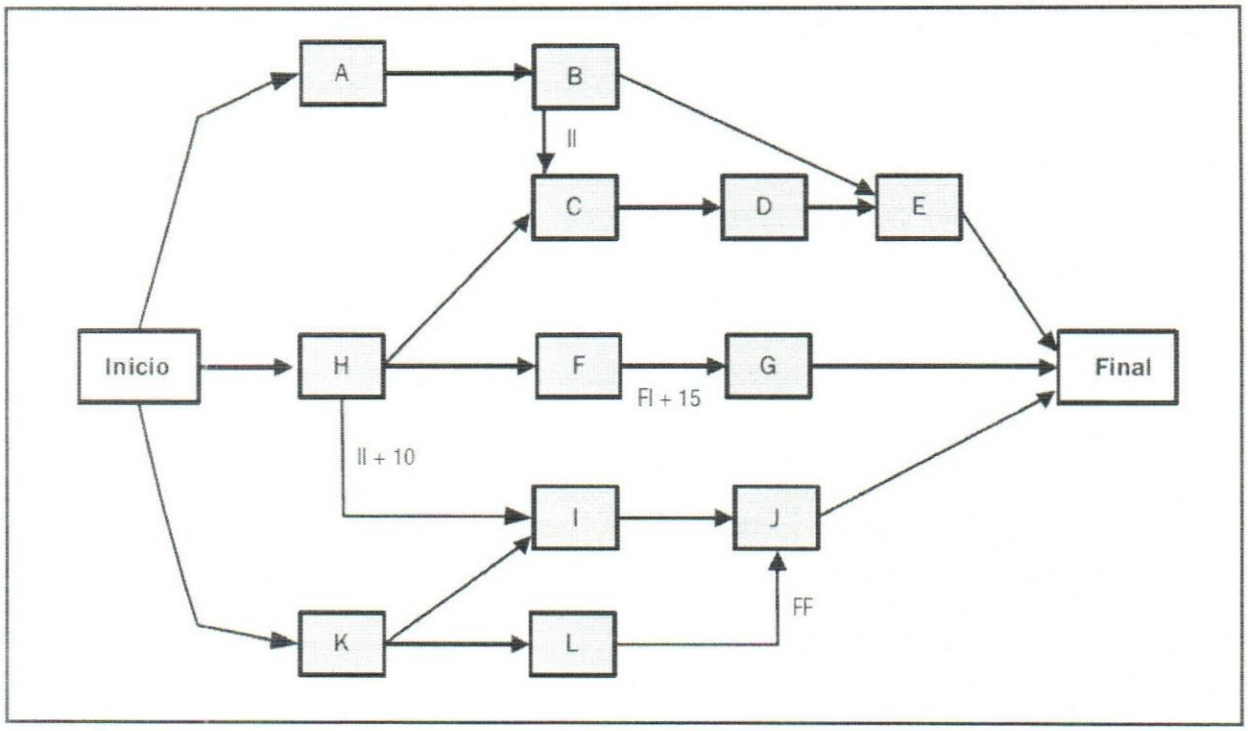

Figura 9: Método de Diagramación por Precedencia
Fuente: PMI (2008).

- **Dependencias discrecionales.** El equipo del proyecto determina qué dependencias son discrecionales durante el proceso de establecimiento de la secuencia de las actividades. A veces, las dependencias discrecionales se denominan lógica preferida, lógica preferencial o lógica blanda. Las dependencias discrecionales se establecen con base en el conocimiento de las mejores prácticas dentro de un área de aplicación determinada o a algún aspecto poco común del proyecto, donde se desea una secuencia específica, aunque existan otras secuencias aceptables. Las dependencias discrecionales deben documentarse totalmente, ya que pueden crear valores arbitrarios de holgura total y pueden limitar las opciones posteriores de planificación. Cuando se emplean técnicas de ejecución rápida, estas dependencias discrecionales deben revisarse, y debe considerarse su modificación o eliminación.

- **Dependencias externas.** El equipo de dirección del proyecto determina qué dependencias son externas durante el proceso de establecimiento de la secuencia de las actividades. Las dependencias externas implican una relación entre las

actividades del proyecto y aquéllas que no pertenecen al proyecto. Normalmente, estas dependencias están fuera del control del equipo del proyecto. Por ejemplo, la actividad de prueba en un proyecto de software puede depender de la entrega del hardware por parte de una fuente externa, o en el caso de un proyecto de construcción, puede ser necesario realizar informes gubernamentales de evaluación del impacto ambiental antes de iniciar la preparación del emplazamiento.

.3 Aplicación de Adelantos y Retrasos

El equipo de dirección de proyecto determina las dependencias que pueden necesitar un adelanto o un retraso para definir con exactitud la relación lógica. No deben utilizarse adelantos y retrasos para sustituir la lógica de la planificación. Deben documentarse las actividades y sus supuestos relacionados.

Un adelanto permite una aceleración de la actividad sucesora. Por ejemplo, en un proyecto para la construcción de un nuevo edificio de oficinas, puede planificarse que el desmonte del terreno comience dos semanas antes de la fecha programada para completar la lista de pendientes. Esto debe mostrarse como una relación lógica final a inicio, con un adelanto de dos semanas.

Un retraso ocasiona una demora en la actividad sucesora. Por ejemplo, un equipo de redacción técnica puede comenzar a editar el borrador de un documento extenso quince días después de haber comenzado a escribirlo. Esto puede mostrarse como una relación lógica inicio a inicio con un retraso de 15 días.

.4 Plantillas de Red del Cronograma

Para acelerar la preparación de las redes de actividades del proyecto, pueden emplearse plantillas normalizadas del diagrama de red del cronograma del proyecto. Pueden abarcar un proyecto completo o sólo una parte del mismo. Las partes de un diagrama de red del cronograma del proyecto se denominan a menudo subred o fragmento de red. Las plantillas de las subredes son especialmente útiles cuando un proyecto abarca varios entregables idénticos o casi idénticos, como los pisos de un edificio alto de oficinas, los estudios clínicos de un proyecto de investigación

farmacológica, los módulos de codificación de programas de un proyecto de software o la fase de lanzamiento de un proyecto de desarrollo.

Secuenciar las Actividades: Salidas

.1 Diagramas de Red del Cronograma del Proyecto

Los diagramas de red del cronograma del proyecto son una representación esquemática de las actividades del cronograma del proyecto y de sus relaciones lógicas, también denominadas dependencias. La figura 10 muestra un diagrama de red del cronograma del proyecto. La elaboración de un diagrama de red del cronograma del proyecto puede hacerse en forma manual o mediante la utilización de un software de gestión de proyectos. Puede incluir todos los detalles del proyecto o contener una o más actividades resumen. Una narración resumida puede adjuntarse al diagrama y describir la metodología básica utilizada para secuenciar las actividades. Todas las secuencias inusuales de las actividades dentro de la red deberán describirse íntegramente en la narración.

.2 Actualizaciones a los Documentos del Proyecto

Entre los documentos del proyecto que pueden actualizarse, se incluyen, entre otros:

- Las listas de actividades
- Los atributos de la actividad
- El registro de riesgos

Estimar los Recursos de las Actividades

Estimar los Recursos de las Actividades es el proceso que consiste en estimar el tipo y las cantidades de materiales, personas, equipos o suministros requeridos para ejecutar cada actividad. Véanse los Gráficas Figura 10: Estimar los Recursos de las Actividades: Entradas, Herramientas y Técnicas, y Salidas y Figura 11: Diagrama de Flujo de Datos del

Proceso Estimar los Recursos de las Actividades. El proceso Estimar los Recursos de las Actividades está estrechamente coordinado con el proceso Estimar los Costos

Por ejemplo:

- El equipo de un proyecto de construcción necesitará familiarizarse con los códigos de edificación locales. A menudo, es posible acceder fácilmente a este conocimiento a través de los proveedores locales. Sin embargo, si la mano de obra local carece de la experiencia en el uso de técnicas de construcciones inusuales o especializadas, el costo adicional de la contratación de un consultor puede ser la manera más efectiva de asegurar el conocimiento de los códigos de edificación locales.
- Un equipo de diseño de un automóvil necesitará familiarizarse con las técnicas de ensamblado automatizado más recientes. El conocimiento requerido puede obtenerse mediante la contratación de un consultor, el envío de un diseñador a un seminario de robótica o la incorporación de alguna persona del departamento de producción al equipo del proyecto.

Figura 10: Estimar los Recursos de las Actividades: Entradas, Herramientas y Técnicas, y Salidas
Fuente: PMI (2008).

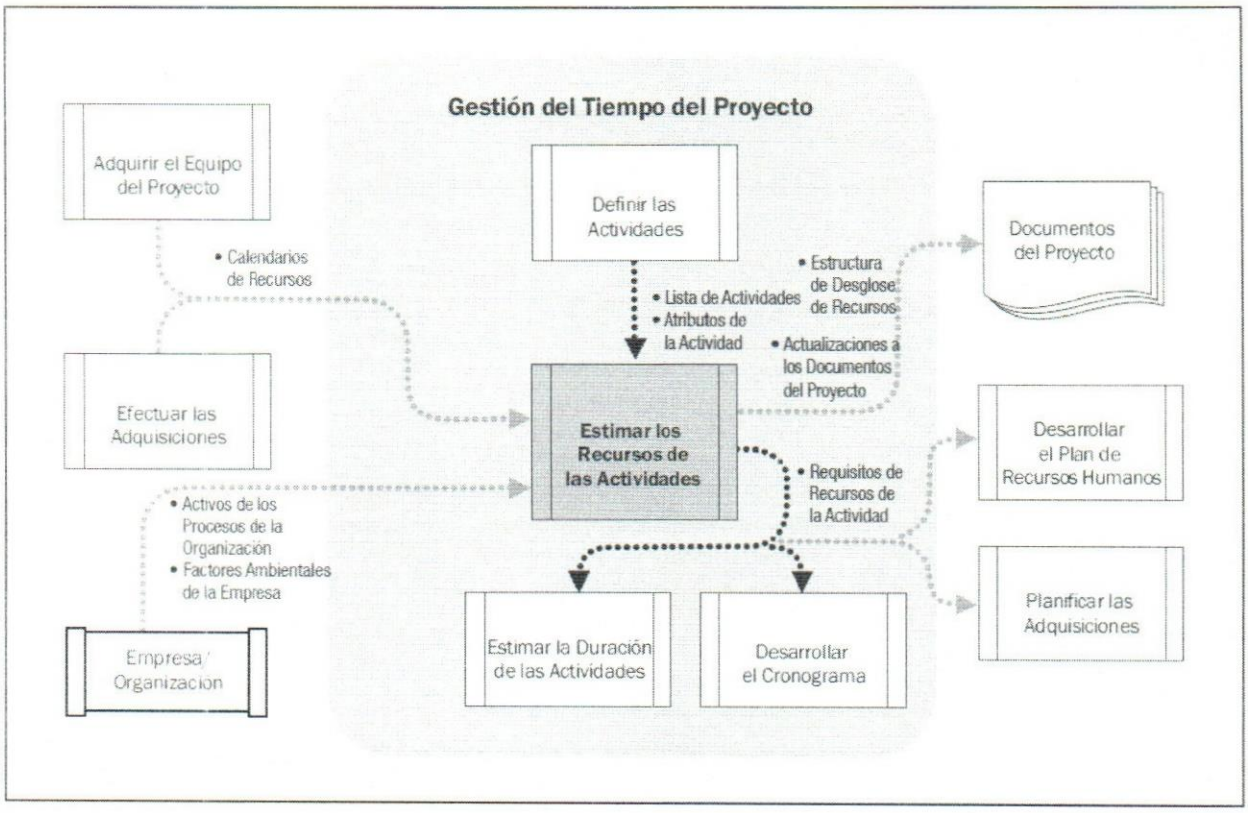

Figura 11: Diagrama de Flujo de Datos del Proceso Estimar los Recursos de las Actividades
Fuente: PMI (2008).

Estimar los Recursos de las Actividades: Entradas

.1 Lista de Actividades

La lista de actividades identifica las actividades que necesitarán recursos.

.2 Atributos de la Actividad

Los atributos de la actividad desarrollados durante los procesos Definir las Actividades y Secuenciar las Actividades, proporcionan la entrada de datos principal que se utilizará para estimar los recursos necesarios para cada una de las actividades de la lista.

.3 Calendarios de Recursos

La información sobre los recursos (como personas, equipos y material) potencialmente disponibles durante la ejecución de las actividades planificadas, permite estimar la utilización de recursos. Los calendarios de recursos especifican cuándo y por cuánto

tiempo estarán disponibles los recursos identificados del proyecto durante la ejecución del mismo. Esta información puede proporcionarse a nivel de la actividad o del proyecto. Este conocimiento abarca la consideración de atributos, tales como la experiencia y/o el nivel de habilidad de los recursos, así como las diferentes ubicaciones geográficas de las que provienen los recursos y cuándo pueden estar disponibles.

El calendario de recursos compuesto abarca la disponibilidad, las capacidades y las habilidades de los recursos humanos. Por ejemplo, durante las fases iniciales de un proyecto de diseño de ingeniería, el grupo de recursos podría incluir una gran cantidad de ingenieros jóvenes e ingenieros experimentados. Sin embargo, durante las fases posteriores del mismo proyecto, el grupo puede limitarse a aquellas personas con mayor conocimiento en el proyecto, por haber trabajado en las fases iniciales del mismo.

.4 Factores Ambientales de la Empresa

Entre los factores ambientales de la empresa que pueden influir en el proceso Estimar los Recursos de las Actividades, se encuentran la disponibilidad y las habilidades de los recursos.

.5 Activos de los Procesos de la Organización

Los activos de los procesos de la organización que pueden influir en el proceso Estimar los Recursos de las Actividades incluyen, entre otros:

- Las políticas y procedimientos relativos a los recursos humanos
- Las políticas y procedimientos relacionados con el alquiler y la adquisición de suministros y equipos
- La información histórica acerca de los tipos de recursos utilizados para trabajos similares en proyectos anteriores.

Estimar los Recursos de las Actividades: Herramientas y Técnicas

.1 Juicio de Expertos

A menudo, se requiere el juicio de expertos para evaluar las entradas a este proceso relacionadas con los recursos. Cualquier grupo o persona con conocimientos especializados en planificación y estimación de recursos puede proporcionar dicha experiencia.

.2 Análisis de Alternativas

Muchas actividades del cronograma cuentan con métodos alternativos de realización. Éstos abarcan el uso de distintos niveles de capacidad o habilidades de los recursos, diferentes tamaños y tipos de máquinas, diferentes herramientas (manuales vs. automáticas) y la decisión de fabricar o comprar los recursos.

.3 Datos de Estimación Publicados

Muchas empresas publican periódicamente los índices de producción actualizados y los costos unitarios de los recursos para una gran variedad de industrias, materiales y equipos, en diferentes países y en diferentes ubicaciones geográficas dentro de esos países.

.4 Estimación Ascendente

Cuando una actividad no puede estimarse con un grado razonable de confianza, el trabajo dentro de esa actividad se descompone a un nivel mayor de detalle. Se estiman las necesidades de recursos. Estos estimados se suman luego en un total para cada uno de los recursos de la actividad. Las actividades pueden o no tener dependencias entre sí, que pueden afectar la aplicación y el uso de los recursos. Si existen dependencias, este patrón de uso de recursos se refleja y se documenta en los requisitos estimados de la actividad.

.5 Software de Gestión de Proyectos

El software de gestión de proyectos tiene la capacidad de ayudar a planificar, organizar y gestionar los grupos de recursos, y de desarrollar estimados de los mismos. En función de la complejidad del software, pueden definirse las estructuras de desglose de recursos, su disponibilidad y sus costos, así como diversos calendarios, para ayudar en la optimización del uso de recursos.

Estimar los Recursos de las Actividades: Salidas

.1 Requisitos de Recursos de la Actividad

La salida del proceso Estimar los Recursos de las Actividades identifica los tipos y la cantidad de recursos necesarios para cada actividad de un paquete de trabajo. Estos requisitos pueden sumarse para determinar los recursos estimados para cada paquete de trabajo. El nivel de detalle y especificidad de las descripciones de los requisitos de recursos puede variar según el área de aplicación. La documentación de los requisitos de recursos para cada actividad puede incluir la base de la estimación de cada recurso, así como los supuestos considerados al determinar los tipos de recursos que se aplican, su disponibilidad y en qué cantidad se utilizan.

.2 Estructura de Desglose de Recursos

La estructura de desglose de recursos es una estructura jerárquica de los recursos, identificados por categoría y tipo de recurso. Algunos ejemplos de categorías de recursos son la mano de obra, el material, los equipos y los suministros. Los tipos de recursos pueden incluir el nivel de habilidad, el nivel de formación u otra información apropiada para el proyecto. La estructura de desglose de recursos es útil para organizar y comunicar los datos del cronograma del proyecto, incluyendo la información sobre utilización de recursos.

.3 Actualizaciones a los Documentos del Proyecto

Entre los documentos del proyecto que pueden actualizarse, se incluyen, entre otros:

- La lista de actividades
- Los atributos de la actividad
- Los calendarios de recursos

Estimar la Duración de las Actividades

Estimar la Duración de las Actividades es el proceso que consiste en establecer aproximadamente la cantidad de períodos de trabajo necesarios para finalizar cada actividad con los recursos estimados. La estimación de la duración de las actividades utiliza información sobre el alcance del trabajo de la actividad, los tipos de recursos necesarios, las cantidades estimadas de los mismos y sus calendarios de utilización. Las entradas para los estimados de la duración de las actividades surgen de la persona o grupo del equipo del proyecto que esté más familiarizado con la naturaleza del trabajo en la actividad específica. El estimado de la duración se elabora de manera gradual, y el proceso evalúa la calidad y disponibilidad de los datos de entrada. Por ejemplo, conforme va evolucionando el trabajo de ingeniería y diseño del proyecto, se dispone de datos más detallados y precisos, lo cual mejora la exactitud de los estimados de la duración. Por lo tanto, puede suponerse que el estimado de la duración será cada vez más preciso y de mejor calidad. Véanse los Gráficos 12 y 13.

El proceso Estimar la Duración de las Actividades requiere que se estime la cantidad de esfuerzo de trabajo requerido y la cantidad de recursos para completar la actividad; esto permite determinar la cantidad de periodos de trabajo (duración de la actividad) necesarios para completar la actividad. Se documentan todos los datos y supuestos que respaldan el estimado de la duración para cada estimado de duración de la actividad.

La mayor parte del software de gestión de proyectos para planificación manejará esta situación mediante el calendario del proyecto y los calendarios de recursos de periodos

de trabajo alternativos que, por lo general, se identifican por los recursos que requieren periodos de trabajo específicos. Además de la lógica de secuencia, las actividades se realizarán de acuerdo con el calendario del proyecto y los calendarios de recursos correspondientes.

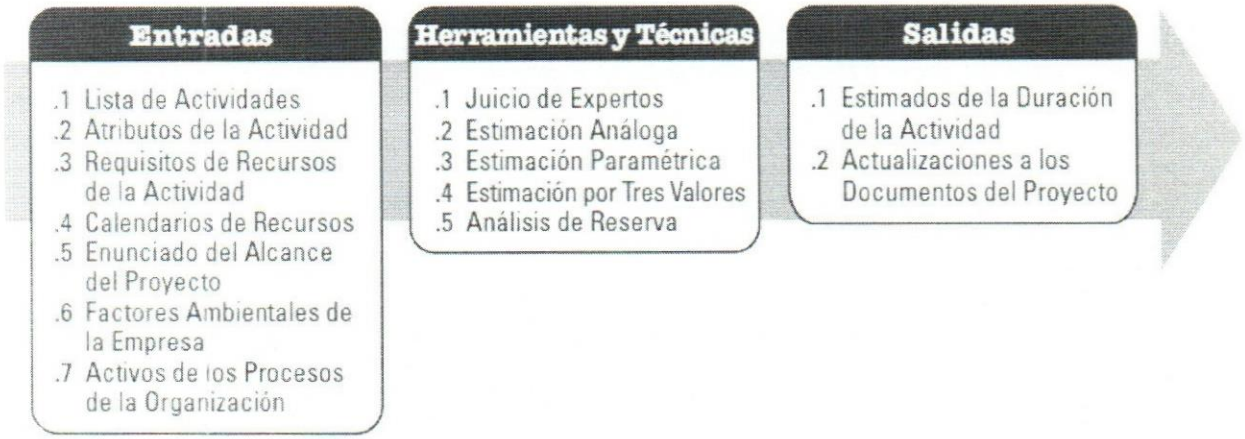

Figura 12: Estimar la Duración de las Actividades: Entradas, Herramientas y Técnicas, y Salidas
Fuente: (PMI, 2008)

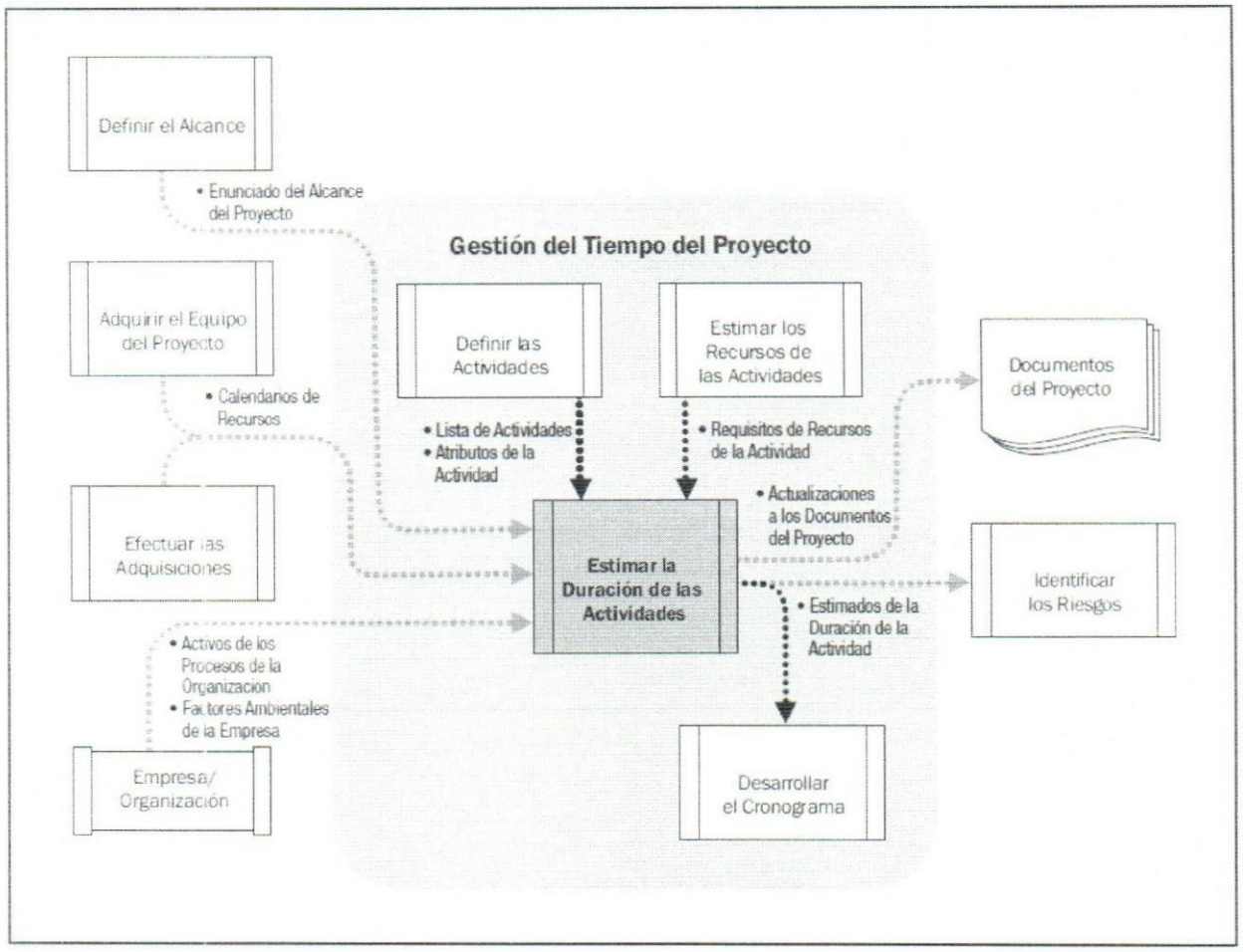

Figura 13: Diagrama de Flujo de Datos del Proceso Estimar la Duración de las Actividades
Fuente: (PMI, 2008).

Estimar la Duración de las Actividades: Entradas

.1 Lista de Actividades

Descritas en el punto de definición de actividades de salida de herramientas y técnicas.

.2 Atributos de la Actividad

Descritas en el punto de definición de actividades de salida de herramientas y técnicas.

.3 Requisitos de Recursos de la Actividad

Los requisitos estimados de recursos de las actividades tendrán un efecto sobre la duración de las actividades, puesto que los recursos asignados a cada actividad y la disponibilidad de los mismos influirán de manera significativa en la duración de la mayoría de las actividades. Por ejemplo, si se asignan recursos adicionales o con menos habilidades a una actividad, puede producirse una disminución del rendimiento o de la productividad debido a que se incrementarán las necesidades de comunicación, capacitación y coordinación.

.4 Calendarios de Recursos

El calendario de recursos, desarrollado como parte del proceso Estimar los Recursos de las Actividades, puede abarcar el tipo de recursos humanos, su disponibilidad y su capacidad. También se consideran, si corresponde, el tipo, la cantidad, la disponibilidad y la capacidad tanto de los equipos como de los recursos materiales, que pueden influir significativamente en la duración de las actividades del cronograma. Por ejemplo, cuando se asigna con dedicación completa a un miembro del personal junior y a uno *senior*, por lo general se espera que el miembro *senior* realice una actividad determinada en menos tiempo que el miembro junior.

.5 Declaración del Alcance del Proyecto

Las restricciones y supuestos de la declaración del alcance del proyecto se tienen en cuenta al estimar la duración de las actividades. Entre los ejemplos de supuestos, se incluyen, entre otros:

- Las condiciones existentes
- Las disponibilidad de información
- La frecuencia de los periodos de presentación de informes

Entre los ejemplos de restricciones, se incluyen:

- La disponibilidad de recursos capacitados
- Los términos y requisitos del contrato

.6 Factores Ambientales de la Empresa

Los factores ambientales de la empresa que pueden influir en el proceso Estimar la Duración de las Actividades incluyen, entre otros:

- Las bases de datos de los estimados de la duración y otros datos de referencia
- Las métricas de productividad
- La información comercial publicada

.7 Activos de los Procesos de la Organización

Los activos de los procesos de la organización que pueden influir en el proceso Estimar la Duración de las Actividades incluyen, entre otros:

- La información histórica relativa a la duración
- Los calendarios del proyecto
- La metodología de planificación
- Las lecciones aprendidas

Estimar la Duración de las Actividades: Herramientas y Técnicas

.1 Juicio de Expertos

El juicio de expertos, guiado por la información histórica, puede proporcionar información sobre el estimado de la duración o las duraciones máximas recomendadas, procedentes de proyectos similares anteriores. El juicio de expertos también puede utilizarse para determinar si es conveniente combinar métodos de estimación y cómo conciliar las diferencias entre ellos.

.2 Estimación Análoga

La estimación análoga utiliza parámetros de un proyecto anterior similar, tales como la duración, el presupuesto, el tamaño, la carga y la complejidad, como base para estimar los mismos parámetros o medidas para un proyecto futuro. Cuando se trata de estimar duraciones, esta técnica utiliza la duración real de proyectos similares anteriores como base para estimar la duración del proyecto actual. Es un método de estimación del

valor bruto, que a veces se ajusta en función de diferencias conocidas en cuanto a la complejidad del proyecto.

La estimación análoga de la duración se emplea frecuentemente para estimar la duración de un proyecto cuando existe una cantidad limitada de información detallada sobre el mismo, como es el caso, por ejemplo, en las fases iniciales del proyecto. La estimación análoga utiliza la información histórica y el juicio de expertos.

Por lo general, la estimación análoga es menos costosa y requiere menos tiempo que las otras técnicas, pero también es menos exacta. La estimación análoga de costos puede aplicarse a todo un proyecto o a partes del mismo, y puede utilizarse en conjunto con otros métodos de estimación. La estimación análoga es más confiable cuando las actividades anteriores son similares, no sólo en apariencia sino por hecho, y cuando los miembros del equipo del proyecto responsables de efectuar los estimados poseen la experiencia necesaria.

.3 Estimación Paramétrica

La estimación paramétrica utiliza una relación estadística entre los datos históricos y otras variables (por ej., pies cuadrados en la construcción) para calcular una estimación de parámetros de una actividad tales como costo, presupuesto y duración.

La duración de la actividad puede determinarse cuantitativamente multiplicando la cantidad de trabajo por realizar por la cantidad de horas de trabajo por unidad de trabajo. Por ejemplo, en un proyecto de diseño, la duración de una actividad puede estimarse multiplicando el número de planos por la cantidad de horas de trabajo necesarias para cada plano; o para una instalación de cable, multiplicando los metros de cable por la cantidad de horas de trabajo necesarias para instalar cada metro de cable. Si, por ejemplo, el recurso asignado es capaz de instalar 25 metros de cable por hora, la duración requerida para instalar 1.000 metros sería de 40 horas. (1.000 metros divididos por 25 metros por hora).

Con esta técnica pueden lograrse niveles más altos de exactitud, dependiendo de la sofisticación y de los datos que utilice el modelo. La estimación paramétrica de tiempo

puede aplicarse a todo un proyecto o a partes del mismo, en conjunto con otros métodos de estimación.

.4 Estimación por Tres Valores

La precisión de los estimados de la duración de la actividad puede mejorarse tomando en consideración el grado de incertidumbre y de riesgo de la estimación. Este concepto se originó con la Técnica de Revisión y Evaluación de Programas (método PERT). El método PERT utiliza tres estimados para definir un rango aproximado de duración de una actividad:

- Más probable. Es la duración de la actividad, en función de los recursos que probablemente se asignarán, de su productividad, de las expectativas realistas de disponibilidad para la actividad, de las dependencias de otros participantes y de las interrupciones.
- Optimista. La duración de la actividad está basada en el análisis del mejor escenario posible para esa actividad.
- Pesimista. La duración de la actividad está basada en el análisis del peor escenario posible para esa actividad.

Estimar la Duración de las Actividades: Salidas

.1 Estimados de la Duración de la Actividad

Los estimados de la duración de las actividades son valoraciones cuantitativas de la cantidad probable de periodos de trabajo que se necesitarán para completar una actividad. Los estimados de duración no incluyen ningún retraso. Los estimados de la duración de las actividades pueden incluir alguna indicación del rango de resultados posibles.

.2 Actualizaciones a los Documentos del Proyecto

Entre los documentos del proyecto que pueden actualizarse, se incluyen, entre otros:

- Los atributos de la actividad

- Los supuestos hechos durante el desarrollo del estimado de la duración de las actividades, como los niveles de habilidad y disponibilidad

Desarrollar el Cronograma

Desarrollar el Cronograma es el proceso que consiste en analizar el orden de las actividades, su duración, los requisitos de recursos y las restricciones para crear el cronograma del proyecto.

La incorporación de las actividades, duraciones y recursos a la herramienta de planificación genera un cronograma con fechas planificadas para completar las actividades del proyecto. A menudo, el desarrollo de un cronograma aceptable del proyecto es un proceso iterativo que determina las fechas de inicio y finalización planificadas para las actividades del proyecto y los hitos. El desarrollo del cronograma puede requerir el repaso y revisión de los estimados de la duración y de los recursos para crear un cronograma de proyecto aprobado que pueda servir como línea base con respecto a la cual se pueda medir el avance. La revisión y el mantenimiento de un cronograma realista continúan a lo largo del proyecto conforme el trabajo avanza, el plan para la dirección del proyecto cambia y la naturaleza de los eventos de riesgo evoluciona. Ver los Gráficas 14 y 15.

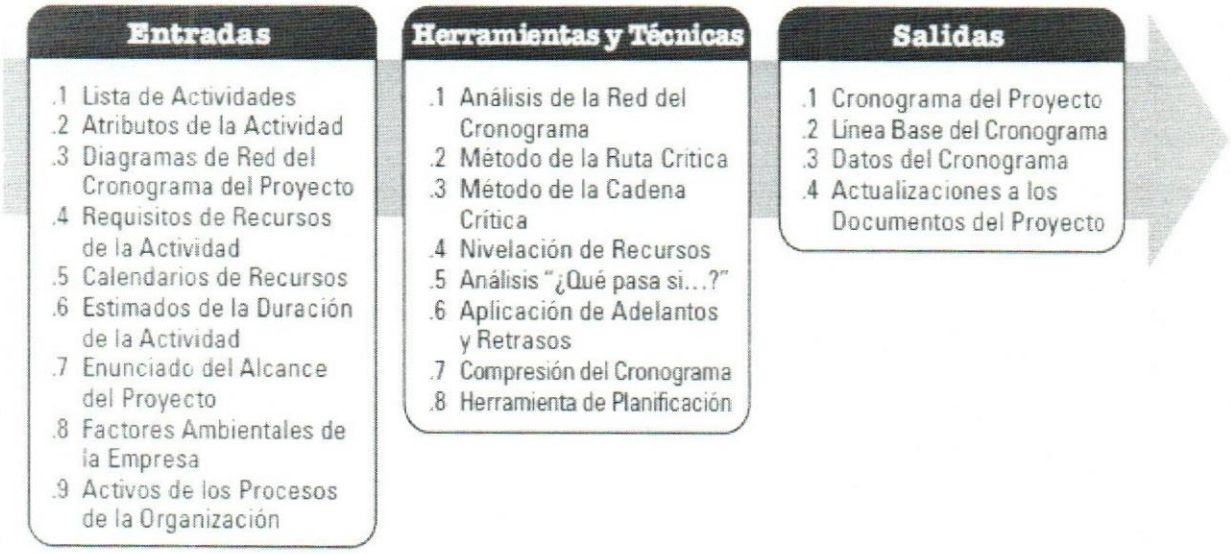

Figura 14: Desarrollar el Cronograma: Entradas, Herramientas y Técnicas, y Salidas
Fuente: (PMI, 2008)

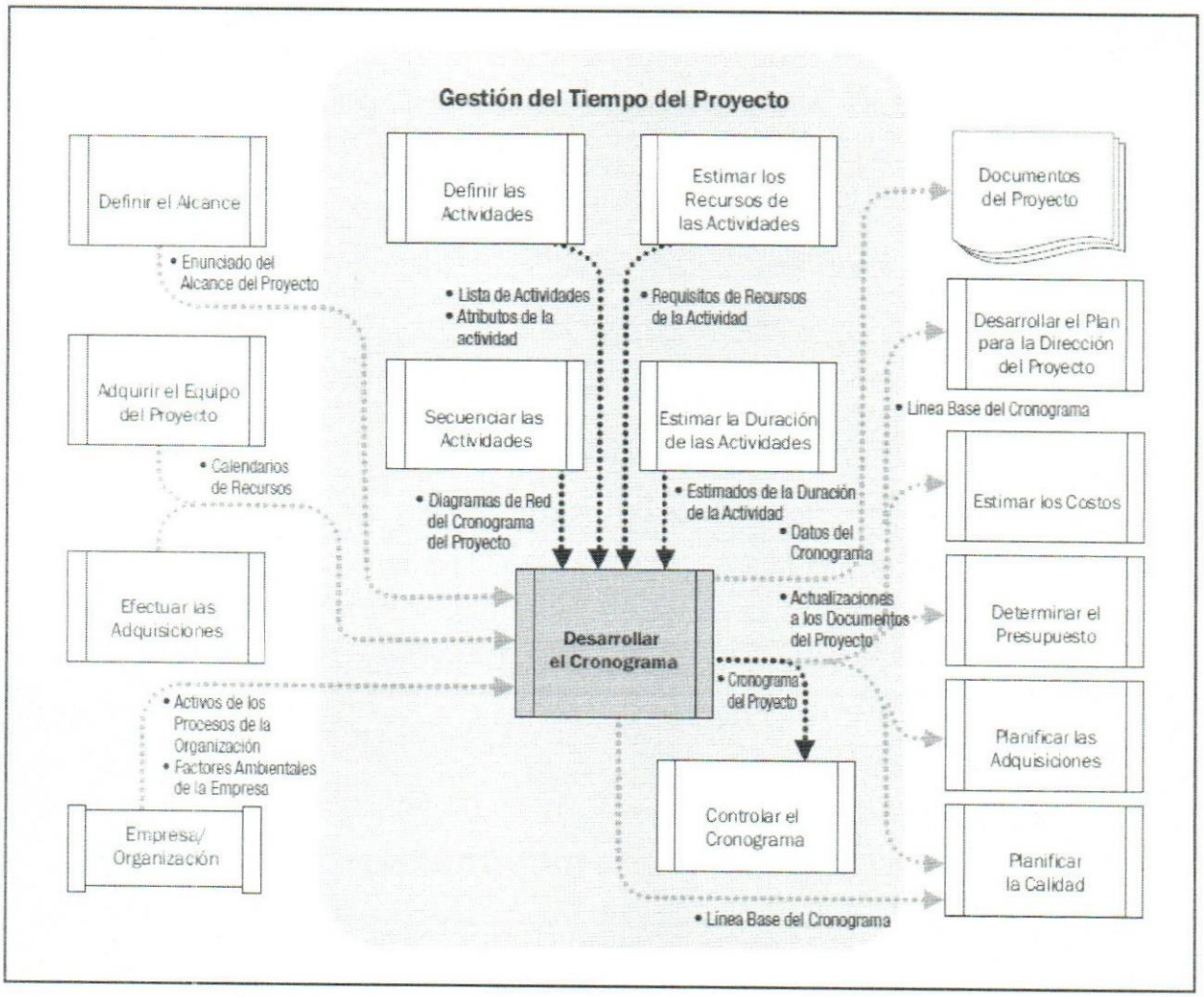

Figura 15: Diagrama de Flujo de Datos del Proceso Desarrollar el Cronograma
Fuente: (PMI, 2008)

Desarrollar el Cronograma: Entradas

.1 Lista de Actividades

Descrita en el punto Definir las Actividades que constituye el primer proceso de gestión de tiempo.

.2 Atributos de la Actividad

Descritos en el punto definir las actividades de salida (atributos de la actividad)

.3 Diagramas de Red del Cronograma del Proyecto

Descritos en el punto Estimar los Recursos de las actividades de entrada

.4 Requisitos de Recursos de la Actividad

Descritos en el punto Estimar los Recursos de las actividades de salida

.5 Calendarios de Recursos

Descritos en el punto Estimar los Recursos de las actividades de entrada, calendario de recursos.

.6 Estimados de la Duración de la Actividad

Descritos en el punto Estimar la Duración de las Actividades

.7 Declaración del Alcance del Proyecto

La declaración del alcance del proyecto contiene supuestos y restricciones que pueden causar un impacto en el desarrollo del cronograma del proyecto.

.8 Factores Ambientales de la Empresa

Entre los factores ambientales de la empresa que pueden influir en el proceso Desarrollar el Cronograma, se encuentra la herramienta de planificación que puede utilizarse para el desarrollo del cronograma.

.9 Activos de los Procesos de la Organización

Los activos de los procesos de la organización que pueden influir en el proceso Desarrollar el Cronograma incluyen, entre otros:

- La metodología de planificación
- El calendario del proyecto

Desarrollar el Cronograma: Herramientas y Técnicas

.1 Análisis de la Red del Cronograma

El análisis de la red del cronograma es una técnica utilizada para generar el cronograma del proyecto. Emplea diversas técnicas analíticas, tales como el método de

la ruta crítica, el método de la cadena crítica, el análisis "¿Qué pasa si...?" y la nivelación de recursos, para calcular las fechas de inicio y finalización tempranas y tardías para las partes no completadas de las actividades del proyecto. Algunos caminos de red pueden tener puntos de convergencia o divergencia de rutas que pueden identificarse y emplearse en el análisis de compresión del cronograma o en otros análisis.

.2 Método de la Ruta Crítica

El método de la ruta crítica calcula las fechas teóricas de inicio y finalización tempranas y tardías para todas las actividades, sin considerar las limitaciones de recursos, realizando un análisis que recorre hacia adelante y hacia atrás toda la red del cronograma. Las fechas de inicio y finalización tempranas y tardías resultantes no constituyen necesariamente el cronograma, sino que más bien indican los periodos dentro de los cuales pueden planificarse las actividades, teniendo en cuenta las duraciones de las actividades, las relaciones lógicas, los adelantos, los retrasos y otras restricciones conocidas.

Las fechas de inicio y finalización tempranas y tardías calculadas pueden ser afectadas por la holgura total de la actividad que proporciona flexibilidad al cronograma y cuyo valor puede ser positivo, negativo o nulo. En cualquier camino de red, la flexibilidad del cronograma se mide por la diferencia positiva entre las fechas tempranas y tardías, lo cual se conoce como "holgura total". Las rutas críticas tienen una holgura total igual a cero o negativa y las actividades del cronograma en una ruta crítica reciben el nombre de "actividades críticas". Una ruta crítica se caracteriza normalmente por el hecho de que su holgura total es igual a cero. Las redes pueden tener varias rutas casi críticas. Puede ser necesario realizar ajustes a las duraciones de las actividades, a sus relaciones lógicas, a los adelantos y a los retrasos, o a otras restricciones del cronograma para lograr caminos de red con una holgura total igual a cero o positiva. Una vez que se ha calculado la holgura total de un camino de red, entonces puede determinarse la holgura libre, que es la cantidad de tiempo que una actividad puede retrasarse dentro de un camino de red, sin demorar la fecha de inicio temprana de cualquier actividad sucesora inmediata dentro de dicho camino de red.

.3 Método de la Cadena Crítica

La cadena crítica es una técnica de análisis de la red del cronograma que permite modificar el cronograma del proyecto para adaptarlo a los recursos limitados. Inicialmente, el diagrama de red del cronograma del proyecto se elabora mediante los estimados de la duración, con las dependencias requeridas y las restricciones definidas como entradas. Entonces se calcula la ruta crítica. Una vez que se ha identificado la ruta crítica, se ingresa la disponibilidad de recursos y se determina el resultado del cronograma con recursos limitados. A menudo, el cronograma resultante presenta una ruta crítica modificada.

La ruta crítica con restricciones de recursos se conoce como cadena crítica. El método de la cadena crítica agrega colchones de duración, que son actividades del cronograma que no requieren trabajo y que se utilizan para manejar la incertidumbre. Un colchón que se coloca al final de la cadena crítica se conoce como colchón del proyecto y protege la fecha de finalización objetivo contra cualquier retraso a lo largo de la cadena crítica. Se colocan colchones adicionales, conocidos como colchones de alimentación, en cada punto donde una cadena de tareas dependientes, que está fuera de la cadena crítica, la alimenta. De este modo, los colchones de alimentación protegen la cadena crítica contra retrasos a lo largo de las cadenas de alimentación. La dimensión de cada colchón debe tener en cuenta la incertidumbre en la duración de la cadena de tareas dependientes que conducen a ese colchón. Una vez que se han determinado las actividades del cronograma con colchón, las actividades previstas se planifican en base a sus fechas posibles de inicio y finalización programadas más tardías.

Consecuentemente, en lugar de gestionar la holgura total de los caminos de red, el método de la cadena crítica se concentra en gestionar las duraciones restantes de los colchones en función de las duraciones restantes de las cadenas de tareas.

.4 Nivelación de Recursos

La nivelación de recursos es una técnica de análisis de la red del cronograma que se aplica a un cronograma que ya ha sido analizado por medio del método de la ruta crítica. La nivelación de recursos puede utilizarse cuando los recursos compartidos o

críticos necesarios sólo están disponibles en ciertos momentos o en cantidades limitadas, o para mantener la utilización de recursos en un nivel constante. La nivelación de recursos es necesaria cuando los recursos han sido sobre asignados, es decir, cuando un recurso se ha asignado a dos o más tareas para el mismo periodo, o cuando los recursos compartidos o críticos necesarios sólo están disponibles en ciertos periodos o en cantidades limitadas. La nivelación de recursos provoca a menudo cambios en la ruta crítica.

.5 Análisis "¿Qué pasa si...?"

Éste es un análisis de la pregunta "¿Qué pasa si se produce la situación representada por el escenario 'X'?" Se realiza un análisis de la red del cronograma, usando el cronograma para calcular los diferentes escenarios, tales como un retraso en la entrega de un componente principal, la prolongación de la duración de un diseño específico o la introducción de factores externos, como una huelga o un cambio en el procedimiento para la obtención de permisos. Los resultados del análisis del escenario "Qué pasa si..." pueden usarse para evaluar la viabilidad del cronograma del proyecto bajo condiciones adversas, y para preparar planes de contingencia y respuesta para superar o mitigar el impacto de situaciones inesperadas. La simulación implica calcular múltiples duraciones del proyecto a partir de diferentes conjuntos de supuestos sobre las actividades. La técnica más común es la del Análisis Monte Carlo, en el cual se define una distribución de duraciones posibles para cada actividad, que es usada para calcular una distribución de posibles resultados para todo el proyecto.

.6 Aplicación de Adelantos y Retrasos

Los adelantos y retrasos son refinamientos que se aplican durante el análisis de la red para desarrollar un cronograma viable.

.7 Compresión del Cronograma

La compresión del cronograma reduce el calendario del proyecto sin modificar el alcance del mismo, para cumplir con las restricciones del cronograma, las fechas

impuestas u otros objetivos del cronograma. Las técnicas de compresión del cronograma incluyen:

- Compresión. Una técnica de compresión del cronograma en la cual se analizan las concesiones entre costo y cronograma para determinar cómo obtener la mayor compresión con el menor incremento de costo. Ejemplos de compresión pueden incluir la aprobación de horas suplementarias, el aporte de recursos adicionales o un pago adicional para acelerar la entrega de las actividades que se encuentran en la ruta crítica. La compresión sólo funciona para actividades en las que los recursos adicionales permiten acortar la duración. La compresión no siempre resulta una alternativa viable y puede ocasionar un incremento del riesgo y/o del costo.
- Ejecución rápida. Una técnica de compresión del cronograma en la cual las fases o actividades que normalmente se realizarían en forma secuencial, se realizan en paralelo.

Un ejemplo de esto es la construcción de los cimientos de un edificio antes de finalizar todos los planos arquitectónicos. La ejecución rápida puede dar como resultado un reproceso y un aumento del riesgo. La ejecución rápida sólo funciona en actividades que pueden superponerse para acortar la duración.

.8 Herramienta de Planificación

Las herramientas automatizadas de planificación aceleran el proceso de planificación, generando fechas de inicio y finalización basadas en las entradas de actividades, los diagramas de red, los recursos y las duraciones de las actividades. Una herramienta de planificación puede utilizarse conjuntamente con otro software de gestión de proyectos, así como con métodos manuales.

Desarrollar el Cronograma: Salidas

.1 Cronograma del Proyecto

El cronograma del proyecto debe contener, como mínimo, una fecha de inicio y una fecha de finalización programadas para cada actividad. Si la planificación de recursos

se realiza en una etapa temprana, entonces el cronograma mantendrá su carácter preliminar hasta que se hayan confirmado las asignaciones de recursos y se hayan establecido las fechas de inicio y finalización planificadas. Por lo general, este proceso se lleva a cabo antes de la conclusión del plan para la dirección del proyecto. También puede desarrollarse un cronograma planificado del proyecto con fechas de inicio y finalización objetivo definidas para cada actividad. El cronograma del proyecto puede presentarse en forma de resumen, denominado a veces cronograma maestro o cronograma de hitos, o presentarse en forma detallada. Aunque el cronograma del proyecto puede tener forma de tabla, se presenta más a menudo en forma gráfica, utilizando uno o más de los siguientes formatos:

- **Diagramas de hitos**. Estos diagramas son similares a los diagramas de barras, pero sólo identifican el inicio o la finalización programada de los principales entregables y las interfaces externas clave.
- **Diagramas de barras**. Estos diagramas, con barras que representan las actividades, muestran las fechas de inicio y finalización de las actividades, así como las duraciones esperadas. Los diagramas de barras son relativamente fáciles de leer y se utilizan frecuentemente en presentaciones de dirección. Para la comunicación de control y de dirección, se utiliza una actividad resumen más amplia y completa, denominada a veces actividad resumen, entre hitos o a través de múltiples paquetes de trabajo interdependientes, y se representa en informes de diagrama de barras.
- **Diagramas de red del cronograma del proyecto**. Estos diagramas, con la información de la fecha de las actividades, normalmente muestran la lógica de la red del proyecto y las actividades del cronograma que se encuentran dentro de la ruta crítica del proyecto. Estos diagramas pueden presentarse con el formato de diagrama de actividad en el nodo, que a veces se denomina diagrama lógico de barras.

.2 Línea base del Cronograma

La línea base del cronograma es una versión específica del cronograma del proyecto desarrollada a partir del análisis de la red del cronograma. El equipo de dirección del

proyecto la acepta y aprueba como la línea base del cronograma, con fechas de inicio y fechas de finalización de línea base. La línea base del cronograma es un componente del plan para la dirección del proyecto.

.3 Datos del Cronograma

Los datos para el cronograma del proyecto abarcan, por lo menos, los hitos del cronograma, las actividades del cronograma, los atributos de las actividades y la documentación de todos los supuestos y restricciones identificadas. La cantidad de datos adicionales varía según el área de aplicación. La información suministrada frecuentemente como detalles de soporte incluye, entre otras:

- Los requisitos de recursos por periodo de tiempo, a menudo presentados en el formato de un histograma de recursos.
- Los cronogramas alternativos, tales como el mejor o el peor escenario, sin nivelación de recursos, con nivelación de recursos, con o sin fechas impuestas.
- La planificación de las reservas para contingencias. Los datos del cronograma también podrían abarcar elementos tales como histogramas de recursos, proyecciones del flujo de caja y cronogramas de pedidos y entregas.

.4 Actualizaciones a los Documentos del Proyecto

Entre los documentos del proyecto que pueden actualizarse, se incluyen:

- **Requisitos de recursos de la actividad**. La nivelación de recursos puede tener un efecto significativo en los estimados preliminares de los tipos y cantidades de recursos necesarios. Si el análisis de nivelación de recursos modifica los requisitos de recursos del proyecto, estos últimos son actualizados.
- **Atributos de las actividades**. Los atributos de las actividades se actualizan para incluir todos los requisitos de recursos revisados y cualquier otra revisión generada por el proceso Desarrollar el Cronograma.
- **Calendario**. El calendario para cada proyecto puede utilizar diferentes unidades de calendario como base para planificar el proyecto.

- **Registro de riesgos**. El registro de riesgos puede necesitar actualizarse para reflejar las oportunidades o las amenazas identificadas al establecer los supuestos de la planificación.

Controlar el Cronograma

Controlar el Cronograma es el proceso por el que se da seguimiento al estado del proyecto para actualizar el avance del mismo y gestionar cambios a la línea base del cronograma. Véanse los Gráficas 16 y 17. Controlar el Cronograma consiste en:

- Determinar el estado actual del cronograma del proyecto
- Influir en los factores que generan cambios en el cronograma
- Determinar que el cronograma del proyecto ha cambiado
- Gestionar los cambios reales conforme suceden

Controlar el Cronograma es un componente del proceso de Realizar el Control Integrado de Cambios.

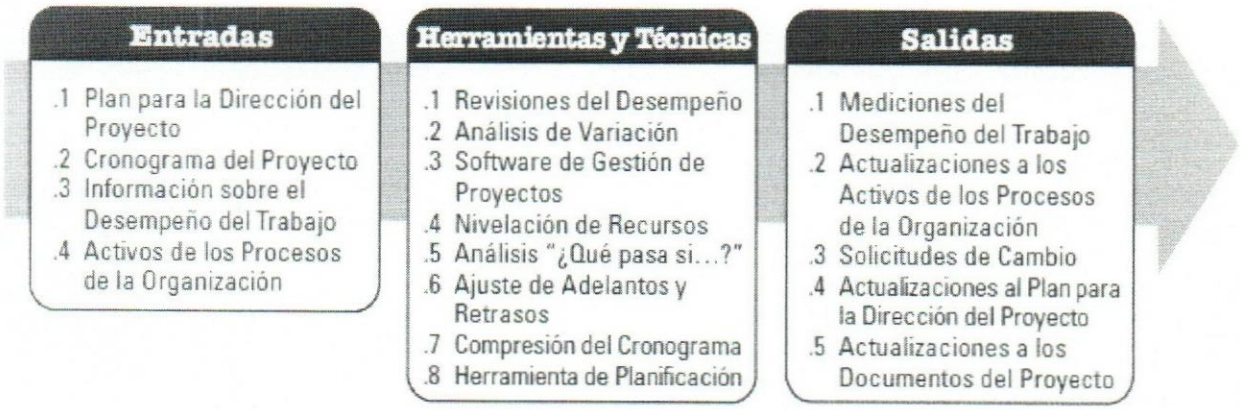

Figura 16: Desarrollar el Cronograma: Entradas, Herramientas y Técnicas, y Salidas
Fuente: PMI (2008).

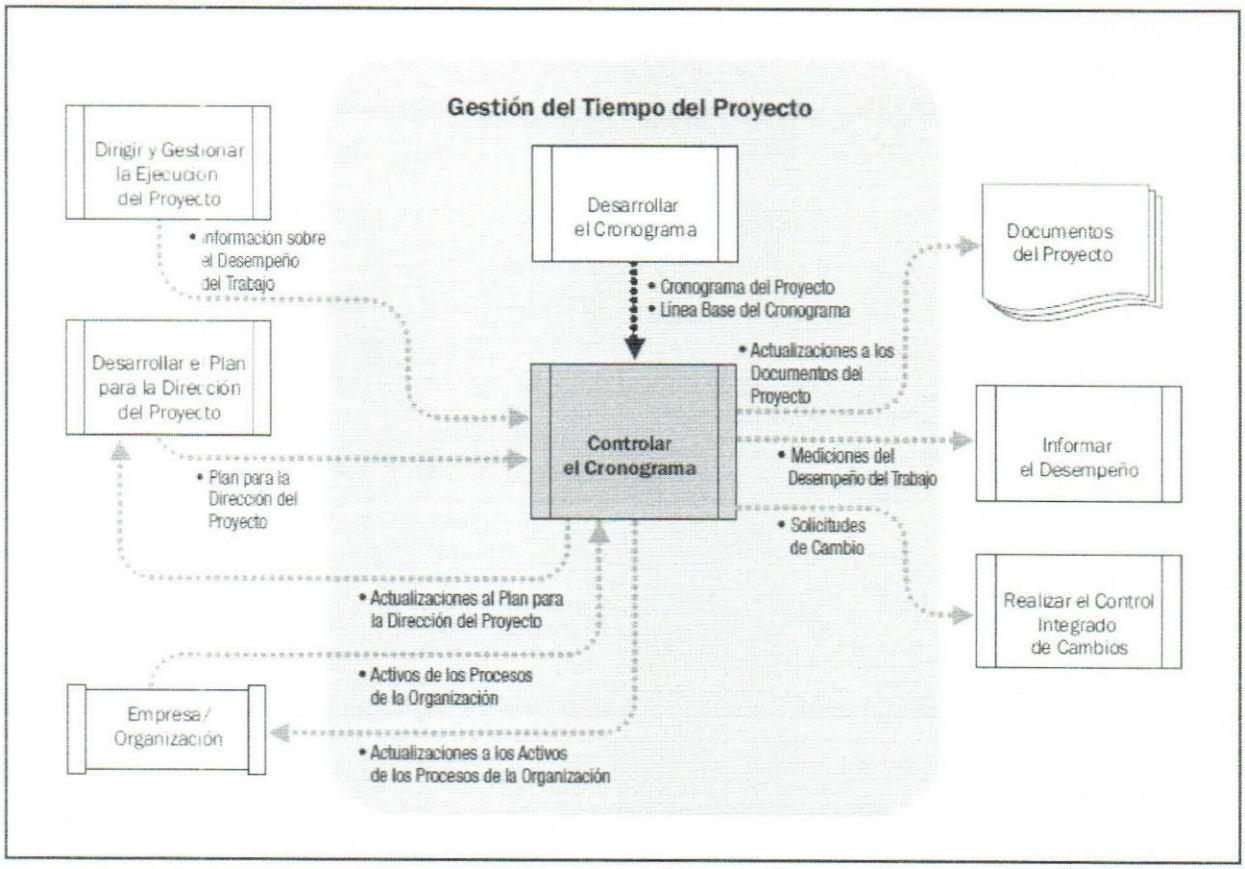

Figura 17: Diagrama de Flujo de Datos del Proceso Controlar el Cronograma
Fuente: (PMI, 2008)

Controlar el Cronograma: Entradas

.1 Plan para la Dirección del Proyecto

El plan para la dirección del proyecto, contiene el plan de gestión del cronograma y la línea base del cronograma. El plan de gestión del cronograma describe cómo se gestionará y controlará el cronograma del proyecto. La línea base del cronograma se compara con los resultados reales para determinar si es necesario un cambio o una acción preventiva o correctiva.

.2 Cronograma del Proyecto

Se trata de la versión más reciente del cronograma del proyecto, con anotaciones que indican las actualizaciones, las actividades terminadas y las actividades iniciadas a la fecha de los datos indicada.

.3 Información sobre el Desempeño del Trabajo

Es la información sobre el avance del proyecto, tal como qué actividades se han iniciado, su avance y qué actividades se han terminado.

.4 Activos de los Procesos de la Organización

Los activos de los procesos de la organización que influyen en el proceso Controlar el

Cronograma incluyen, entre otros:

- Las políticas, procedimientos y lineamientos existentes, formales e informales, relacionados con el control del cronograma
- Las herramientas de control del cronograma
- Los métodos de seguimiento e información que se utilizarán

Controlar el Cronograma: Herramientas y Técnicas

.1 Revisiones del Desempeño

Las revisiones del desempeño permiten medir, comparar y analizar el desempeño del cronograma, en aspectos como las fechas reales de inicio y finalización, el porcentaje completado y la duración restante para el trabajo en ejecución. Si se utiliza la gestión del valor ganado, se usan la variación del cronograma (SV) y el índice de desempeño del cronograma (SPI) para evaluar la magnitud de las variaciones del cronograma. Una parte importante del control del cronograma es decidir si la variación del cronograma requiere acciones correctivas.

.2 Análisis de Variación

Las mediciones del desempeño del cronograma (SV, SPI) se utilizan para evaluar la magnitud de variación con respecto a la línea base original del cronograma. La

variación de la holgura total es también un componente esencial de la planificación para evaluar el desempeño del proyecto en el tiempo. Los aspectos importantes del control del cronograma del proyecto incluyen la determinación de la causa y del grado de variación con relación a la línea base del cronograma y la decisión de la necesidad de aplicar o no acciones preventivas o correctivas.

.3 Software de Gestión de Proyectos

El software de gestión de proyectos para la elaboración de cronogramas permite hacer un seguimiento de las fechas planificadas en comparación con las fechas reales, y de proyectar los efectos de los cambios al cronograma del proyecto.

.4 Nivelación de Recursos

La nivelación de recursos se utiliza para optimizar la distribución del trabajo entre los recursos.

.5 Análisis "¿Qué pasa si...?"

El análisis "¿Qué pasa si...?" se utiliza para revisar diferentes escenarios para realinear el cronograma con el plan.

.6 Ajuste de Adelantos y Retrasos

El ajuste de adelantos y retrasos se usa para encontrar maneras de realinear con el plan las actividades retrasadas del proyecto.

.7 Compresión del Cronograma

Las técnicas de compresión del cronograma se usan para encontrar maneras de realinear con el plan las actividades retrasadas del proyecto.

.8 Herramienta de Planificación

Los datos del cronograma se actualizan y compilan en el cronograma para reflejar el avance real del proyecto y el trabajo que queda pendiente. La herramienta de planificación y los datos de apoyo del cronograma se utilizan conjuntamente con

métodos manuales u otro software de gestión de proyectos para realizar el análisis de la red del cronograma y generar un cronograma actualizado del proyecto.

6.6.3 Controlar el Cronograma: Salidas

.1 Mediciones del Desempeño del Trabajo

Los valores calculados de la variación del cronograma (SV) y del índice de desempeño del cronograma (SPI) para los componentes de la EDT, en particular los paquetes de trabajo y las cuentas de control, se documentan y comunican a los interesados.

.2 Actualizaciones a los Activos de los Procesos de la Organización

Entre los activos de los procesos de la organización que pueden actualizarse, se incluyen, entre otros:

- Las causas de las variaciones
- Las acciones correctivas seleccionadas y la razón de su selección
- Otros tipos de lecciones aprendidas procedentes del control del cronograma del proyecto

.3 Solicitudes de Cambio

El análisis de la variación del cronograma, junto con la revisión de los informes de avance, resultados de las medidas de desempeño y las modificaciones al cronograma del proyecto, pueden dar como resultado solicitudes de cambio a la línea base del cronograma y/o a otros componentes del plan para la dirección del proyecto. Las solicitudes de cambio se procesan para su revisión y tratamiento por medio del proceso Realizar el Control Integrado de Cambios. Las acciones preventivas pueden incluir cambios recomendados para reducir la probabilidad de variaciones negativas del cronograma.

.4 Actualizaciones al Plan para la Dirección del Proyecto

Entre los elementos del plan para la dirección del proyecto que pueden actualizarse, se encuentran:

- Línea base del cronograma. Los cambios a la línea base del cronograma se incorporan en respuesta a las solicitudes de cambio aprobadas relacionadas con cambios en el alcance del proyecto, en los recursos de las actividades o en los estimados de la duración de las actividades.
- Plan de gestión del cronograma.
- Línea base de costo. La línea base de costo puede actualizarse para reflejar los cambios originados por las técnicas de compresión del cronograma.

.5 Actualizaciones a los Documentos del Proyecto

Entre los documentos del proyecto que pueden actualizarse, se incluyen:

- Datos del cronograma. Pueden desarrollarse nuevos diagramas de red del cronograma del proyecto para reflejar las duraciones restantes aprobadas y las modificaciones al plan de trabajo. En algunos casos, los retrasos en el cronograma del proyecto pueden ser tan graves que se deberá desarrollar un nuevo cronograma objetivo, con fechas de inicio y finalización proyectadas, para proporcionar datos realistas a fin de dirigir el trabajo y medir el desempeño y el avance.
- Cronograma del proyecto. Se generará un cronograma actualizado del proyecto a partir de los datos actualizados del cronograma, para reflejar los cambios al mismo y gestionar el proyecto.

2.2.2. Desarrollos iterativos

Un desarrollo Iterativo e incremental, según Stapleton (1997), es "El desarrollo iterativo e incremental es necesario para converger sobre una solución empresarial correcta." (p. 18).

Según (Bittner & Spence, 2006), previamente se deben plantear las siguientes interrogantes:

- ¿Qué es lo que hace a un proceso de desarrollo iterativo e incremental?
- ¿Qué significa trabajar de una manera iterativa e incremental?

Para dar respuesta a estas preguntas, el Diccionario de la (Real Academia Española , 2009) ofrece las siguientes definiciones de los verbos involucrados:

Iterar: (Del lat. iterāre). tr. repetir.

Repetir: (Del lat. repetĕre). tr. Volver a hacer lo que se había hecho.

Incrementar: (Del lat. incrementāre). tr. aumentar (ll acrecentar).

Con las definiciones anteriores se logra obtener una respuesta que demuestra que la diferencia es etérea, pero sin dejar de ser importante:

En el contexto de desarrollo de software, se puede decir que el significado de incremental es que se construye sobre lo que ya se desarrolló, y que se van adicionando funcionalidades.

En cuanto a iterativo, que nace de repeticiones, en otras palabras, el software no pasa una sola vez por las fases de desarrollo y se cierra, sino que al quedar pendientes funcionalidades por desarrollar, se reinicia el ciclo de desarrollo siguiendo las fases definidas.

Un proyecto de software donde se desarrolla un prototipo en su primer ciclo, en el ciclo siguiente, se deja a un lado dicho prototipo, y se inicia el desarrollo de la aplicación, se trata de un proyecto iterativo pero que no es incremental debido a que no se agregan nuevas funcionalidades.

En un desarrollo "iterativo incremental", se debe analizar el impacto de cada nuevo cambio para no destruir lo que ya se desarrolló completamente, sino modificarlo levemente para extenderlo al costo mínimo, agregando funcionalidades en las iteraciones siguientes; según Bittner & Spence (2006) lo definen:

> "La necesidad de iteración surge de la necesidad de entregar resultados predecibles en un contexto incierto. Debido a que no se desea de laguna forma la incertidumbre, necesitamos de una técnica para dominarla. El desarrollo Iterativo e incremental nos proporciona una técnica que permite dominar la

incertidumbre, o al menos de manera sistemática ponerla bajo control lo suficientemente para lograr los resultados deseados."(Cap. I)

2.2.3. Modelo Espiral

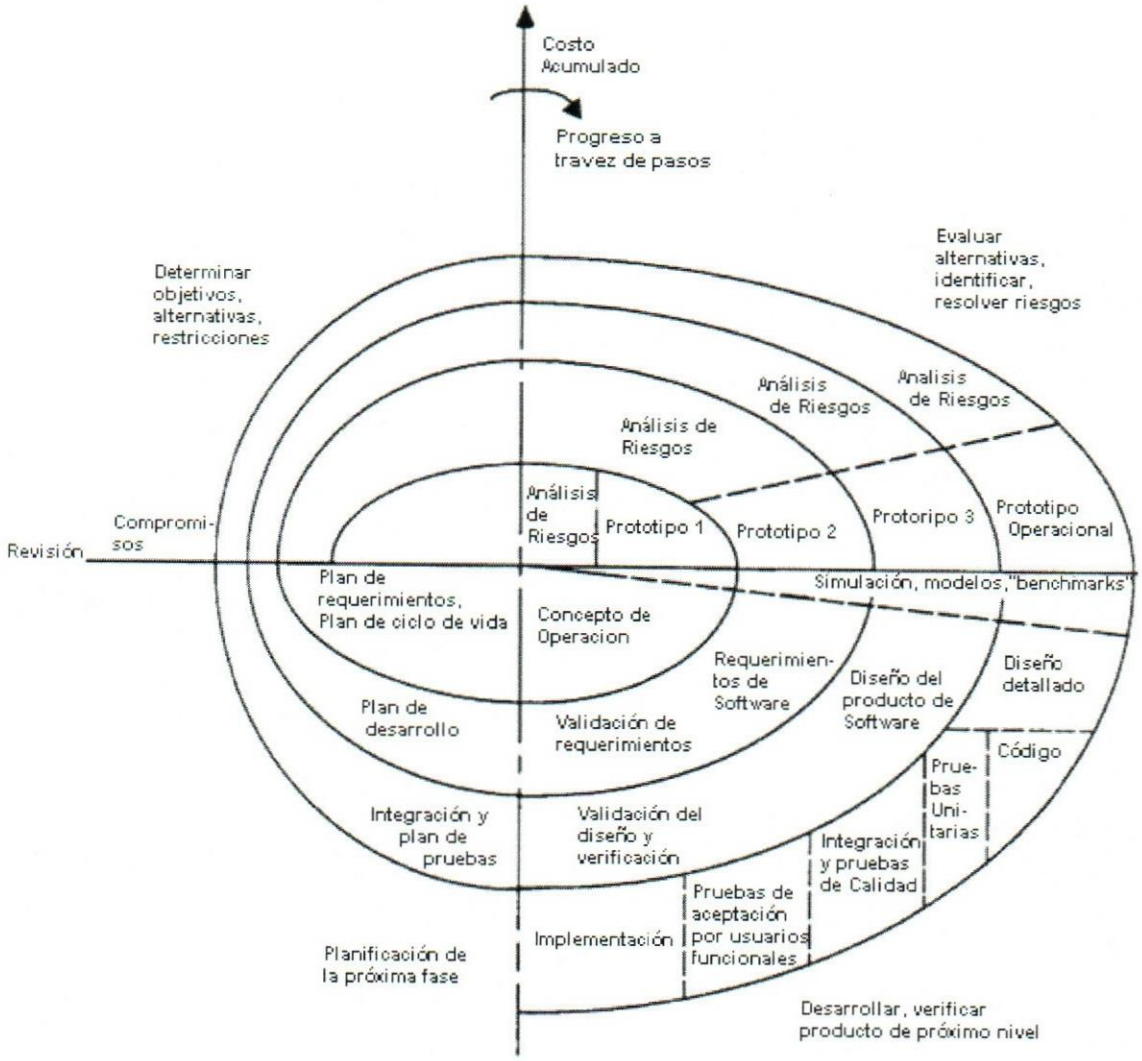

Figura 18: Modelo Espiral de los procesos de software
Fuente: (Boehm, 1988)

Según (Boehm, 1988), el modelo en espiral del proceso del software (ver Figura 18) ha ido evolucionando desde hace varios años, con base en la experiencia sobre diversas mejoras al modelo de cascada que se aplica a los grandes proyectos de software.

Como se verá, el modelo en espiral puede adaptarse a la mayoría de los modelos anteriores como casos especiales y proporciona más una orientación en cuanto a la combinación de los modelos anteriores que mejor se adaptan a una situación dada del software. La dimensión radial en la Figura 18: Modelo Espiral de los procesos de software representa el coste acumulado que se incurre en el cumplimiento de las medidas hasta la fecha, la dimensión angular representa el progreso realizado en completar cada ciclo o espiral. El modelo refleja el concepto subyacente de que cada ciclo, consiste en una progresión que se refiere a la misma secuencia de pasos, para cada porción del producto y para cada uno de sus niveles de elaboración, a partir de un concepto general del documento de la operación hasta la codificación de cada programa individual.

Un ciclo típico de la espiral

Paso 1

Cada ciclo de la espiral comienza con la identificación de:

- Los objetivos de la porción del producto que se elabora (rendimiento, funcionalidad, capacidad de adaptarse a los cambios, etc.);
- Los medios alternativos de aplicación de la presente parte del producto (diseño A, diseño B, la reutilización, comprar, etc.), y
- Las restricciones impuestas sobre la aplicación (costo, horario, interfaz, etc.)

Paso 2

El siguiente paso es evaluar las alternativas en relación con los objetivos y limitaciones. Con frecuencia, en este proceso se identifican las áreas de incertidumbre que son fuentes importantes de riesgo del proyecto.

Paso 3

Si es así, el siguiente paso debe incluir la formulación de una estrategia costo-efectiva para la resolución de las fuentes de riesgo. Esto puede implicar la creación de prototipos, simulación, verificación de referencias de evaluación comparativa,

administración de cuestionarios de usuario, modelado analítico, o combinaciones de estas y otras técnicas de resolución de riesgo.

Paso 4

Una vez que los riesgos se evalúan, el siguiente paso es determinar los riesgos relativos restantes. Si el rendimiento o los riesgos de interfaz de usuario son dominantes sobre el desarrollo, o existen riesgos de control interno de la interfaz,

Paso 5

El siguiente paso puede ser un desarrollo evolutivo: un esfuerzo mínimo para especificar la naturaleza global del producto, un plan para el siguiente nivel de creación de prototipos, y el desarrollo de un prototipo más detallado a seguir para resolver los problemas de riesgo importante.

Si este prototipo es operativamente útil y lo suficientemente robusto como para servir como una base de bajo riesgo para la evolución futura del producto, los pasos subsiguientes impulsados por el riesgo sería la evolución de la serie de prototipos evolutivos (va hacia la derecha en la Figura 18). En este caso, se opta por guiones de especificaciones. Por lo tanto, las consideraciones de riesgo pueden llevar a la ejecución en un proyecto de sólo un subconjunto de pasos posibles en el modelo.

Paso 6

Por otro lado, si los esfuerzos con prototipos anteriores ya han resuelto todos los temas de rendimiento, o los riesgos de interfaz de usuario, y el desarrollo del programa o interfaz a controlar son dominados, el siguiente paso sigue el enfoque en cascada básicos (concepto de operación, requisitos de software, diseño preliminar, etc, (ver Figura 18), modificado según proceda, para incorporar el desarrollo incremental. Cada nivel de especificación de software en la figura es seguida por una etapa de validación y la preparación de planes para el ciclo siguiente. En este caso, las opciones para crear prototipos, simulación, modelo, etc. se tratan, pero no ejerce, lo que lleva a la utilización de un subconjunto de los diferentes pasos.

Este subconjunto de pasos orientados al riesgo del modelo espiral, permiten que el modelo se adapte a cualquier mezcla de especificaciones orientadas a: prototipo, simulación, sistemas automáticos, aproximación, transformación, o al desarrollo de software. En tales casos, la estrategia mixta adecuada se elige considerando la magnitud relativa de los riesgos del programa, en la eficacia relativa de las diferentes técnicas y en la resolución de los riesgos. De manera similar, las consideraciones de gestión de riesgos puede determinar la cantidad de tiempo y esfuerzo que debe dedicarse a otras actividades del proyecto tales como la planificación, gestión de configuración, control de calidad, verificación formal, y las pruebas. En particular, las especificaciones, orientadas al riesgo pueden tener distintos grados de integridad, formalidad y granularidad, en función de los riesgos relativos de hacer muchas o pocas especificaciones.

Una característica importante del modelo en espiral, como la mayoría de los otros modelos, es que cada ciclo se completa con una revisión que incluya a las partes interesadas o las organizaciones interesadas en el producto. Esta revisión cubre todos los productos desarrollados durante el ciclo anterior, incluyendo los planes para el siguiente ciclo y los recursos necesarios para llevarlas a cabo. La revisión de los principales objetivos es garantizar que todas las partes están mutuamente comprometidas con el enfoque para la siguiente fase.

Los planes para las futuras fases también pueden incluir una partición del producto en incrementos sucesivos de desarrollo o de los componentes a desarrollar por las distintas organizaciones o personas (versiones o entregas). Para este último caso, se debe visualizar una serie de ciclos en espiral paralelo. Uno para cada componente, añadiendo una tercera dimensión al concepto presentado en la Figura 18. Por ejemplo, espirales separados podrían ser la evolución de los componentes de software independientes o incrementos. Así, el paso de revisión y el compromiso puede variar de un individuo a través del diseño de los componentes individuales de un programador, a una revisión que incluya los requisitos principales para desarrolladores, clientes, usuarios y organizaciones que harán el mantenimiento.

Iniciar y terminar la espiral

Cuatro preguntas fundamentales surgen al considerar esta presentación del modelo en espiral:

1. ¿Cómo funciona la espiral una vez que ha comenzado?
2. ¿Cómo salir de la espiral cuando conviene dar por concluido un proyecto antes de tiempo?
3. ¿Por qué la espiral finaliza tan abruptamente?
4. ¿Qué pasa con la mejora del software (o mantenimiento)?

La respuesta a estas preguntas implica una observación de que el modelo en espiral se aplica igualmente a los esfuerzos de desarrollo o mejora. En cualquier caso, la espiral se inicia con la hipótesis de que una misión particular de funcionamiento (o un conjunto de misiones) se pueden mejorar mediante un esfuerzo del software. El proceso en espiral a continuación, consiste en una prueba de esta hipótesis: en cualquier momento, si la hipótesis no pasa la prueba (por ejemplo, si los retrasos causan un producto de software que hacen perder su oportunidad de mercado, o si un producto comercial superior está disponible), la espiral es terminada. De lo contrario, termina con la instalación del nuevo software o el modificado, y la hipótesis se comprueba al observar el efecto sobre la misión operativa. Por lo general, la experiencia con la misión operativa conduce a nuevas hipótesis acerca de las mejoras del software, y una espiral de mantenimiento nueva se inicia para probar la hipótesis. Iniciación, terminación e iteración de las tareas y productos de ciclos anteriores son lo que implícitamente se define en el modelo en espiral (aunque no están incluidos en la Figura 18 para simplificar su presentación).

2.2.4. Asociación entre los procesos definidos por la Gestión del Tiempo PMBOK (2008) y el Modelo Espiral

Tabla 1: Asociación entre los procesos definidos por la Gestión del Tiempo y el Modelo Espiral

Asociación de los procesos definidos por		
Gestión del Tiempo (PMBOK, 2008)	**Modelo Espiral** (Boehm, 1988)	Cuadrante
6.1 Definir las Actividades	Plan de requerimientos Requerimientos de software Validación de requerimientos	1 4 4
6.2 Secuenciar actividades 6.3 Estimar los recursos de las actividades	Plan ciclo de vida Plan de desarrollo Plan de pruebas Planificación próxima fase	1 1 1 1
6.4 Estimar la Duración de las Actividades	Compromisos	2
No disponible	Análisis de riesgos	3
6.5 Desarrollar el Cronograma	Concepto de operación Simulación, modelos "benchmark" Prototipos Diseño del producto de software Validación del diseño y verificación Diseño detallado Código Pruebas Unitarias Integración y Pruebas de Calidad Pruebas de aceptación por usuarios funcionales Implementación	3 4 3 4 4 4 4 4 4 4 4
6.6 Controlar el Cronograma	Validación de requerimientos	4

Es notorio que el Modelo en Espiral de Boehm, no está diseñado sobre una secuencia de procesos como la que sugiere el PMBOK (2008), respecto a la gestión del tiempo,

enfocando la mayor carga de trabajo (como en la mayoría de los proyectos de desarrollo de software), en la fase de desarrollo.

2.2.5. Fases de un proyecto según Microsoft Solution Framework

Microsoft Solution Framework (MSF), es un conjunto de principios, modelos, disciplinas, conceptos y directrices para la entrega de soluciones de tecnología de la información propuesto por Microsoft. MSF no se limita al desarrollo de aplicaciones solamente, se aplica también a otros proyectos de TI como los proyectos de implementación, la creación de redes o infraestructura. MSF no obliga al desarrollador a utilizar una metodología específica (Catarata, Agile), pero les permite decidir qué método utilizar.

Meta de MSF

MSF es un conjunto de procesos de ingeniería de software, principios y prácticas probadas con el de objeto permitir a los desarrolladores lograr el éxito en el ciclo de desarrollo de software (SDLC). MSF proporciona una guía adaptable, basada en las experiencias y las mejores prácticas dentro y fuera de Microsoft, para aumentar la posibilidad de la entrega exitosa de una solución de tecnología de la información al cliente, trabajando rápido, disminuyendo el número de personas en el equipo del proyecto, evitando los riesgos , al tiempo que permite resultados de alta calidad.

Fases de Proyecto

Como propone el modelo de procesos MSF (Microsoft, 2003), consta de cinco fases distintas, en la Figura 19 se muestra el flujo de trabajo de la metodología a través de sus fases.

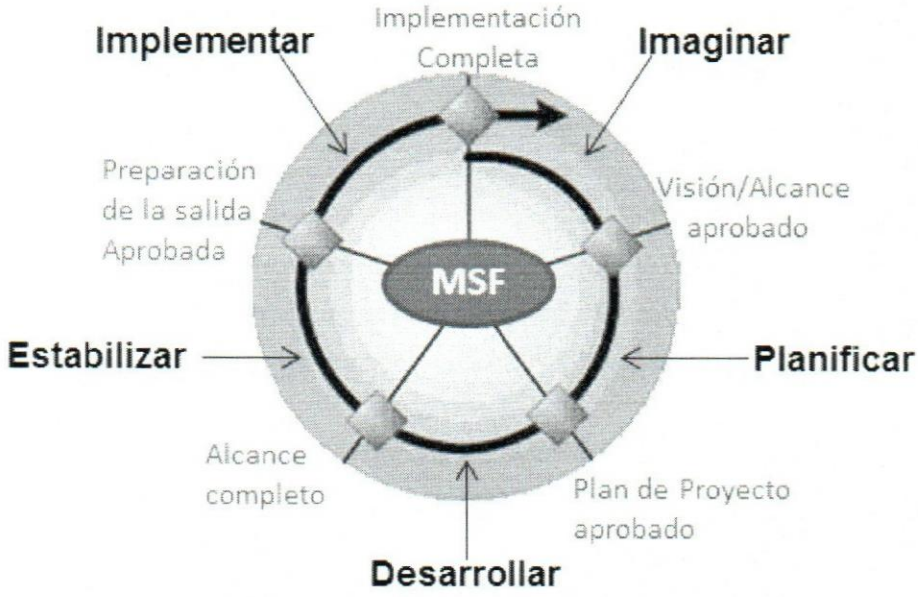

Figura 19: Modelo de proceso MSF
Fuente: Microsoft (2003).

Visión

Esta fase como su nombre lo indica pretende dar una visión general sobre lo que se espera obtener al final del proyecto e implica identificar las tareas y entregables que permitirán cumplir con los requerimientos y objetivos del mismo. Esta fase culmina con el punto de control visión/alcance, el cual contiene: sentencia de visión, perfiles de usuario, alcance del proyecto y concepto de la solución.

Planificar

En esta fase se realiza la preparación de la especificación funcional, diseño de la solución, planes de trabajo, costos estimados y calendarios para los entregables. Adicionalmente se realizan diseño conceptual, diseño lógico y diseño físico.

Desarrollar

La meta de la fase de desarrollo es la construcción de los elementos y entregables de la solución, incluidos los códigos de los componentes, infraestructura (software, hardware, red) y la documentación para el uso de las operaciones.

Estabilizar

La solución es probada en un entorno real de exploración el cual permite determinar que cumple los criterios de calidad necesarios para pasar a la fase de implementación. Se reproducen condiciones reales y el equipo se concentra en detectar y priorizar errores, preparando la solución para su despliegue.

Implementar

Es la última fase de modelo en la cual se instalan los componentes, se estabiliza el proyecto y se obtiene la aprobación por parte del cliente tras la ejecución de un plan de pruebas.

Microsoft a través de MSF, ha recopilado las mejores prácticas empleadas por sus grupos de producto, organización interna de tecnología, sus clientes y socios de negocios en todo el mundo y ha reunido los factores de éxito que son comunes a todas ellas, integrándolos en modelos reutilizables con etapas y logros medibles que pueden guiar las decisiones tecnológicas en un contexto de negocios.

Microsoft a través de MSF, ha recopilado las mejores prácticas empleadas por sus grupos de producto, organización interna de tecnología, sus clientes y socios de negocios en todo el mundo y ha reunido los factores de éxito que son comunes a todas ellas, integrándolos en modelos reutilizables con etapas y logros medibles que pueden guiar las decisiones tecnológicas en un contexto de negocios.

2.2.6. Estructura Desagregada de Trabajo (EDT)

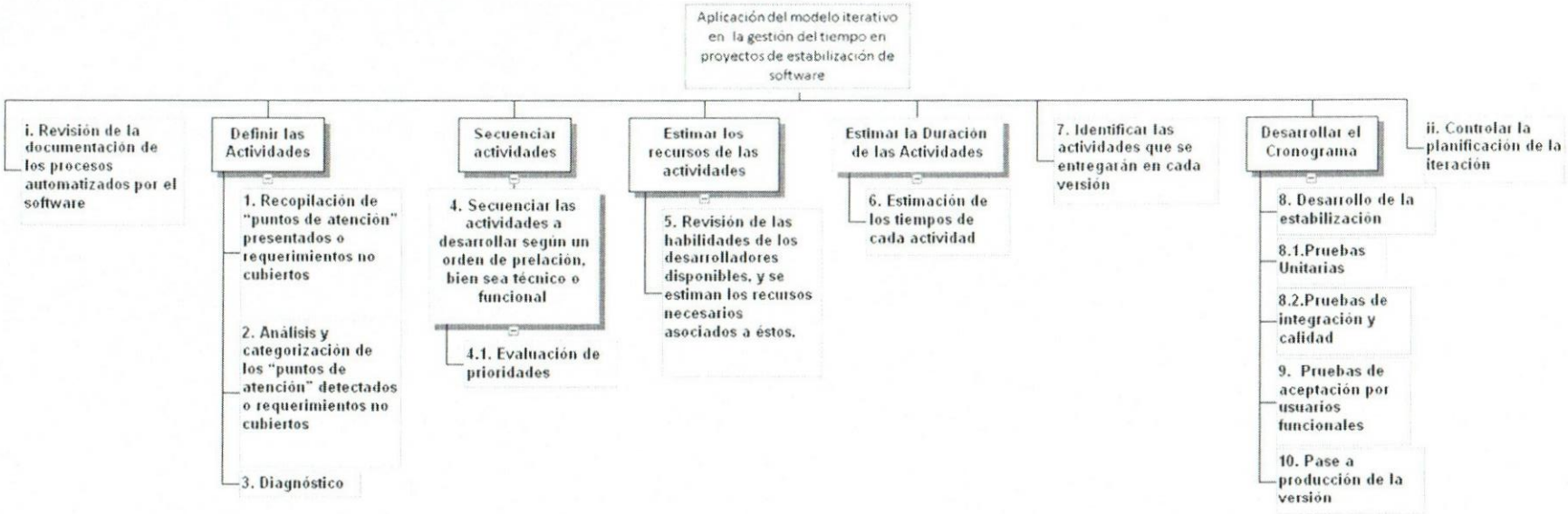

Figura 20: Estructura Desagregada de Trabajo

2.2.7. Bases Legales

Las bases legales que aplican para éste Trabajo Especial de Grado, se enmarcan en la "*Ley sobre el derecho de autor*" de la República Bolivariana de Venezuela, ley basada en la Decisión de la Comisión del Acuerdo de Cartagena de fecha 14 de agosto de 1993, publicada en gaceta oficial en Caracas, 1° de octubre de 1993, Número 4.638 Extraordinario.

Este trabajo en particular, por tratarse de una obra literaria y/o científica, como obra de ingenio e intelecto, y por ser innata de un ser humano, es respaldado por el artículo 1° de dicha ley.

Por ser este trabajo comprendido como una obra literaria o científica que puede ser divulgada o publicada, queda respaldado también por el artículo 2° de la citada ley.

"Título I. De los derechos protegidos. Capitulo I. Disposiciones generales, sección primera, de las obras del ingenio.

Artículo 1°.- Las disposiciones de esta Ley protegen los derechos de los autores sobre todas las obras del ingenio de carácter creador, ya sean de índole literaria, científica o artística, cualquiera que sea su género, forma de expresión, mérito o destino.

Los derechos reconocidos en esta Ley son independientes de la propiedad del objeto material en el cual esté incorporada la obra y no están sometidos al cumplimiento de ninguna formalidad.

Quedan también protegidos los derechos conexos a que se refiere el Titulo IV de esta Ley.

Artículo 2°.- Se consideran comprendidas entre las obras del ingenio a que se refiere el artículo anterior especialmente las siguientes: los libros folletos y otros escritos literarios, artísticos y científicos; incluidos los programas de computación, así como su documentación técnica y manuales de uso; las conferencias, alocuciones sermones y otras obras de la misma naturaleza; las obras dramáticas o dramático-musicales, las

obras coreográficas y pantomímicas, cuyo movimiento escénico se haya fijado por escrito o en otra forma; las composiciones musicales con o sin palabras; las obras cinematográficas y demás obras audiovisuales expresadas por cualquier procedimiento; las; obras de dibujo, pintura, arquitectura, grabado y litografía; las obras de arte aplicado, que no sean meros modelos y dibujos industriales; las ilustraciones y cartas geográficas; los planos, obras plásticas y croquis relativos a la geografía, a la topografía, a la arquitectura o a las ciencias; y en fin , toda producción literaria, científica o artística susceptible de ser divulgada o publicada por cualquier medio o procedimiento." (Congreso de la República de Venezuela, 1993)

2.3. Definiciones

- **Bug-tracking:** Un Sistema de "bug-tracking" es una aplicación de software que está diseñado para ayudar a garantizar la calidad del software y no perder de vista los errores detectados del mismo. Puede ser considerado como un tipo de sistema de seguimiento de problemas o de fallas.
- **Crash,** es una condición en la que un equipo o un programa, ya sea una aplicación o parte del sistema operativo, deja de funcionar correctamente, a menudo se cierra después de encontrar los errores.
- **Patrón de desarrollo**: En ingeniería de software, un patrón de diseño es una solución general reutilizable a un problema que ocurren comúnmente en el diseño de software. Un patrón de diseño no es un diseño de acabado que se puede transformar directamente en el código.
- **Pruebas de caja blanca**: pruebas de caja clara, las pruebas caja de cristal, pruebas de caja transparente, o pruebas estructurales, es un método de pruebas de software que pone a prueba las estructuras internas de funcionamiento o de una solicitud, a diferencia de su funcionalidad (es decir, las pruebas de caja negra). Las pruebas de caja blanca se enfocan en el punto de vista interno del sistema, así como se requieren conocimientos de programación y se utiliza para diseñar casos de prueba. El probador elige entradas para ejercer las rutas a través del código y determinar los resultados apropiados.

- **Pruebas de caja gris**: implica tener conocimiento de las estructuras de datos internas y algoritmos de efectos de diseño de los casos de prueba, pero las pruebas en el usuario en nivel de caja negra. La manipulación de los datos de entrada y el formato de salida que no califican como caja gris, debido a la entrada y salida que están claramente fuera de la "caja negra" que estamos ap0licando al sistema en prueba.
- **Pruebas de caja negra**: es un método de pruebas de software que pone a prueba la funcionalidad de una aplicación a diferencia de sus estructuras internas o funcionamiento (véase la prueba de caja blanca).
- **Simulación**: es la imitación de algo real, el estado de cosas, o proceso. El acto de simulación de algo que implica en general que representan ciertas características clave o comportamientos de un sistema seleccionado físico o abstracto.

CAPÍTULO III. MARCO METODOLÓGICO

3.1. Tipo de investigación

El presente Trabajo Especial de Grado es una investigación del tipo "Investigación-desarrollo", el cual se define de la siguiente manera:

"Consiste en la investigación, elaboración y desarrollo de una propuesta de un modelo operativo viable para solucionar problemas, requerimientos o necesidades de organizaciones o grupos sociales; puede referirse a la formulación de políticas, programas, tecnologías, métodos o procesos. La propuesta del modelo debe tener apoyo en una investigación de tipo documental, de campo o un diseño que incluya ambas modalidades)" (Barrios, 2006).

En base las características más relevantes que definen el enfoque cualitativo de investigación, se puede afirmar que este trabajo especial de grado encaja en dicho enfoque: "Utiliza la recolección de datos sin medición numérica para descubrir o afinar preguntas de investigación en el proceso de interpretación". (Hernández, Fernández, & Baptista, 2006, pág. 8)

3.2. Diseño de la Investigación

El diseño de la investigación es un paso importante, para un trabajo de este tipo, ya que siguiendo el diseño se podrán ir obteniendo sistemáticamente las respuestas que a las preguntas de investigación planteadas. Existen varios patrones de diseño ya establecidos y bien documentados, tales como el Experimental, No Experimental o múltiple.

Para este caso, el patrón de diseño apropiado es el No Experimental, el cual trata de "Estudios que se realizan sin manipulación deliberada de variables y en los que solo se observan los fenómenos en su ambiente natural para después analizarlos.

El tipo específico de diseño No Experimental a utilizar será el llamado "Diseño transeccional correlacional – causal", que según lo define (Hernández, Fernández, & Baptista, 2006, pág. 213) como: "El que describe las relaciones entre dos o más

categorías, conceptos o variables en un momento determinado, ya sea en términos correlaciónales, o en función de la relación causa-efecto"

3.3. Unidad de Análisis

La unidad de análisis se limita al portafolio de proyectos de desarrollo de software que se encuentra escalando en funcionalidades, o se encuentran en producción pero en un estado de inestabilidad que afecta la operatividad del negocio.

3.4. Fases de la Investigación

A continuación se presenta, la estrategia que fue seguida para alcanzar los objetivos planteados para desarrollar la investigación.

Fase 1: El Problema

- Definición del problema.

Fase 2: Revisión de propuestas existentes

- Investigación de la metodología a seguir para desarrollar la propuesta de aplicación del modelo iterativo.
- Revisión en detalle del proceso de gestión del tiempo del PMBOK (2008).

Fase 3: Documentación

- Documentación de la información relacionada la propuesta de aplicación del modelo.
- Documentación de la asociación entre la propuesta de aplicación del modelo a seguir y los procesos definidos por la Gestión del Tiempo del PMBOK (2008).

Fase 4: Análisis de la información

- Análisis y asociación de los procesos del modelo escogido con los procesos de la gestión del tiempo del PMBOK (2008)

Fase 5: Desarrollo

- Desarrollar el modelo metodológico a ser propuesto
- Análisis de los resultados, conclusiones y recomendaciones

3.5. Herramientas y técnicas de recolección de la información

Según (Hernández, Fernández, & Baptista, 2006):

Los principales métodos para recabar datos cualitativos son:

- La observación
- La Entrevista
- Los grupos de enfoque
- La recolección de documentos y materiales
- Y las historias de la vida

El análisis cualitativo implica organizar los datos recogidos, transcribirlos a texto cuando resulta necesario y codificarlos. La codificación tiene dos planos o niveles. Del primero, se genera unidades de significados y categorías. Del segundo, emergen temas y relaciones entre conceptos, Al final se produce teoría enraizada en los datos.

Para el presente trabajo de investigación se usaron las siguientes técnicas y herramientas:

- **La observación:** "No es mera completación ("sentarse a el mundo y tomar notas"); nada de eso, implica adentrarnos en profundidad a situaciones sociales y mantener un papel activo, así como una reflexión permanente. Estar atento a los detalles, sucesos, eventos e interacciones.

Los buenos observadores necesitan utilizar todos sus sentidos para captar los ambientes y a sus actores. Todo puede ser relevante: Clima físico, colores, aromas, espacios, iluminación, etc." (Hernández, Fernández, & Baptista, 2006, pág. 587)

- **Grupos de enfoque:** o sesiones de profundidad, trata de entrevistas grupales, consisten en reuniones de grupos pequeños a medianos, en las cuales los participantes conversan en torno a uno o varios temas en un ambiente relajado e informal.

  "Existe un interés por parte del investigador por cómo los individuos forman un esquema o perspectiva de un problema" (Hernández, Fernández, & Baptista, 2006, pág. 606)

- **La recolección de documentos y materiales**: Se obtienen solamente datos provenientes de documentos.

- **Biografías e historias de vida:** Esta recolección de datos puede ser individual o colectiva, obteniendo información de testimonios, experiencias y lecciones aprendidas.

3.6. Análisis de la información

Según (Hernández, Fernández, & Baptista, 2006):

En el análisis de la información de un proceso cualitativo, la recolección de la información y el análisis de la misma ocurren prácticamente en paralelo; además, el análisis no es estándar, ya que cada estudio requiere de un esquema propio de análisis.

Lo siguiente se trata de una propuesta que se podría seguir para realizar el análisis de la información, ya que podrían adoptarse o no de acuerdo con las circunstancias de un estudio en particular.

En la recolección de datos, el proceso esencial consiste en que se reciben datos no estructurados, pero se les da estructura. Los datos a analizar pueden ser muy variados, pero en esencia son narraciones de los participantes:

a) Visuales (fotografías, videos, pinturas, etc.)
b) Auditivas (grabaciones)
c) Textos escritos (documentos, cartas, etc.)
d) Expresiones verbales y no verbales

Además de las narraciones del investigador.

Las principales características que definen la naturaleza de un análisis cualitativo son las siguientes:

1. Estructurar datos
2. Comprensión del contexto
3. Interpretación
4. Relacionar resultados
5. Organizar datos
6. Agregar el conocimiento del investigador
7. Analizar los datos contextualmente
8. Revisión continua de la información
9. Generar un plan de análisis propio
10. Generar conclusiones
11. Correlacionar datos
12. Categorizar datos
13. Deducir similitudes y diferencias

Después de hacer numerosos análisis, y no se consigue información novedosa, el análisis concluye. Si no se está satisfecho, o aún se necesita más información, comienza un nuevo ciclo, iniciando por la búsqueda de más información y analizarla nuevamente hasta quedar saturados de información que no sea novedosa y cerrar el análisis. Ese comportamiento cíclico que presenta el análisis de información, es conocido como "Espiral de análisis de los datos cualitativos".

3.7. Operacionalización de los Objetivos

Según (Hernández, Fernández, & Baptista, 2006, pág. 146), Una definición operacional especifica qué actividades u operaciones deben realizarse para medir una variable. La siguiente tabla especifica que operaciones, actividades mediciones de variables, etc, deben realizarse para lograr los objetivos planteados.

Tabla 2: Operacionalización de los objetivos.

Objetivo General	Objetivos Específicos	Dimensiones	Técnicas	Instrumento de Recolección	Fuente
Aplicar el modelo iterativo en la gestión del tiempo en proyectos de estabilización de software	Definir los criterios de estabilización en proyectos de software.	Evaluación de las técnicas propuestas por el PMI, en el control y seguimientos de proyectos en las áreas de Tiempo, costo y Recursos Humanos.	Grupos de enfoque	Notas de trabajo	Equipos multidisciplinarios en el área de desarrollo de software: Líderes de proyectos, analistas de desarrollo, arquitectos e ingenieros de software
	Analizar el método iterativo como referencia de uso en la gestión de tiempo en proyectos de software.	Evaluación del modelo en espiral propuesto por Boehm	Revisión de bibliografía especializada.	whitepapers	TRW Defense System Group, A Spiral Model of Software Development and Enhancement (Boehm)
	Formular las estrategias de aplicación del método iterativo en la gestión de tiempo en proyectos de software.	Estudiar la información generada y la necesitada para gestionar los recursos, tiempo y costos de los proyectos.	Experiencias	Hoja de cálculo	Equipos multidisciplinarios en el área de desarrollo de software: Líderes de proyectos, analistas de desarrollo, arquitectos e ingenieros de software

3.8. Consideraciones Éticas

Colegio de Ingenieros de Venezuela, Código de Ética Profesional

Según el CIV, se considera contrario a la ética e incompatible con el digno ejercicio de la profesión, para un miembro del Colegio de Ingenieros de Venezuela:

1. Actuar en cualquier forma que tienda a menoscabar el honor, la responsabilidad y aquellas virtudes de honestidad, integridad y veracidad que deben servir de base a un ejercicio cabal de la profesión. **(Virtudes)**

2. Violar o permitir que se violen las leyes, ordenanzas y reglamentaciones relacionadas con el cabal ejercicio profesional. **(Ilegalidad)**

3. Descuidar el mantenimiento y mejora de sus conocimientos técnicos, desmereciendo así la confianza que al ejercicio profesional concede la sociedad. **(Conocimiento)**

4. Ofrecerse para el desempeño de especialidades y funciones para las cuales no tengan capacidad, preparación y experiencia razonables. **(Seriedad)**

5. Dispensar, por amistad, conveniencia o coacción, el cumplimiento de disposiciones obligatorias, cuando la misión de su cargo sea de hacerlas respetar y cumplir. **(Dispensa)**

6. Ofrecer, solicitar o prestar servicios profesionales por remuneraciones inferiores a las establecidas como mínimas, por el Colegio de Ingenieros de Venezuela. **(Remuneración)**

7. Elaborar proyectos o preparar informes, con negligencia o ligereza manifiestas, o con criterio indebidamente optimista. **(Proyectos)**

8. Firmar inconsultamente planos elaborados por otros y hacerse responsable de proyectos o trabajos que no están bajo su inmediata dirección, revisión o supervisión. **(Firma)**

9. Encargase de obras, sin que se hayan efectuado todos lo estudios técnicos indispensables para su correcta ejecución, o cuando para la realización de las mismas se hayan señalado plazos incompatibles con la buena práctica profesional. **(Obras)**

10. Concurrir deliberadamente o invitar, a licitaciones de Estudio y/o proyectos de obras. **(Licitaciones)**

11. Ofrecer, dar o recibir comisiones o remuneraciones indebidas y, solicitar influencias o usar de ellas para la obtención u otorgamiento de trabajos profesionales, o para crear situaciones de privilegio en su actuación. **(Influencia)**

12. Usar de las ventajas inherentes a un cargo remunerado para competir con la práctica independiente de otros profesionales. **(Ventajas)**

13. Atentar contra la reputación o los legítimos intereses de otros profesionales, o intentar atribuir injustificadamente la comisión de errores profesionales a otros colegas. **(Reputación)**

14. Adquirir intereses que, directa o indirectamente colidan con los de la empresa o cliente que emplea sus servicios o encargarse sin conocimiento de los interesados de trabajos en los cuales existan intereses antagónicos. **(Intereses)**

15. Contravenir deliberadamente a los principios de justicia y lealtad en sus relaciones con clientes, personal subalterno y obreros, de manera especial, con relación a estos últimos, en lo referente al mantenimiento de condiciones equitativas de trabajo y a su justa participación en las ganancias. **(Justicia)**

16. Intervenir directa o indirectamente en la destrucción de los recursos naturales u omitir la acción correspondiente para evitar la producción de hechos que contribuyan al deterioro ambiental. **(El ambiente)**

17. Actuar en cualquier forma que permita o facilite la contratación con profesionales o empresas extranjeras, de estudios o proyectos, construcción, inspección y supervisión de obras, cuando a juicio del Colegio de Ingenieros de Venezuela, exista en Venezuela la capacidad para realizarlos. **(Extranjeros)**

18. Utilizar estudios, proyectos, planos, informes u otros documentos, que no sean el dominio público, sin la autorización de sus autores y/o propietarios.

19. Revelar datos reservados de índole técnico, financiero o profesionales, así como divulgar sin la debida autorización, procedimientos, procesos o características de equipos protegidos por patentes o contratos que establezcan las obligaciones de guardas de secreto profesional. Así como utilizar programas, discos, cintas u otros medios de información, que no sean de dominio público, sin la debida autorización de sus autores y/o propietarios, o utilizar sin autorización códigos de acceso de otras personas, en provecho propio. **(Secreto)**

20. Someter a su cliente a su empleador a la aplicación de materiales o métodos en experimentación, sin su previo y total conocimiento y aprobación o recomendarle servicios no necesarios. **(Experimentación y servicios no necesarios)**

21. Hacer o permitir cualquier publicidad no institucional, dirigida a atraer al público hacia la acción profesional, personal o participar en programas de televisión, radio y otros medios de carácter divulgativo profesional, o que en cualquier forma, atenten contra la dignidad y seriedad de la profesión. Así como, valerse de su posición para proferir declaraciones en los medios o hacer propaganda de materiales, equipos y tecnologías. **(Publicidad)**

22. Incumplir con lo dispuesto en las "Normas de Actuación Gremial del CIV". **(Actuación Gremial)**

Profesional del Project Management Institute, Código de Ética y Conducta
Fuente: (PMI, 2006).

El código de ética y conducta del PMI está sustentado en una visión y propósito que como profesionales de la dirección de proyectos, nos comprometemos a actuar de manera correcta y honorable. Nos fijamos un alto nivel de exigencia, que aspiramos alcanzar en todos los aspectos de nuestras vidas: en el trabajo, en el hogar y al servicio

de nuestra profesión. Los valores más importantes que sustentan este código son: responsabilidad, respeto, equidad y honestidad.

Responsabilidad: Por responsabilidad se hace referencia a nuestra obligación de hacernos cargo de las decisiones que tomamos y de las que no tomamos, de las medidas que tomamos y de las que no, y de las consecuencias que resultan.

Respeto: Respeto es nuestro deber de demostrar consideración por nosotros mismos, los demás y los recursos que nos fueron confiados. Estos últimos pueden incluir personas, dinero, reputación, seguridad de otras personas y recursos naturales o medioambientales. Un ambiente de respeto genera confianza y excelencia en el desempeño al fomentar la cooperación mutua: un ambiente en el que se promueve y valora la diversidad de perspectivas y opiniones.

.**Equidad**: Equidad se refiere a nuestro deber de tomar decisiones y actuar de manera imparcial y objetiva. Nuestra conducta no debe presentar intereses personales en conflicto, prejuicios ni favoritismos.

Honestidad: Honestidad es nuestro deber de comprender la verdad y actuar con sinceridad, tanto en cuanto a nuestras comunicaciones como a nuestra conducta.

CAPÍTULO IV. VENTANA DE MERCADO

La propuesta de aplicación del modelo está basado en un modelo iterativo para la gestión del tiempo en proyectos de estabilización de software, permitirá gestionar proyectos que se están desarrollando o se le están aplicando mejoras y tienden a desestabilizarse perdiendo funcionalidades y haciendo menos productivas a las unidades de negocio que lo utilizan dentro de las organizaciones.

El paquete que acompañará a este producto estará compuesto por el Documento que describe los procesos definidos por la propuesta de aplicación del modelo iterativo, y un instrumento para la gestión de las prioridades a manejar dentro del proyecto.

El ámbito de mercado de esta propuesta de aplicación del modelo cubre a todos los equipos de desarrollo de software, que fabriquen software a la medida de sus necesidades y que se encuentren ubicados en cualquier organización, institución, actividad económica o zona geográfica.

Este producto, podrá ser utilizado para ser ofrecido como un servicio por empresas de consultoría que se dedique a dar apoyo en este sentido a empresas, o ser utilizado por una organización que desarrolle software a la medida, y que cuente con líderes o gerentes de proyectos con la experiencia suficiente para implementarla.

El costo estimado que tendrá la aplicación de éste modelo por consultoría, se encuentra alrededor de 500 Unidades tributarias (Venezuela), o $US7000,00, y se basa en las siguientes grupos de trabajo a llevar a cabo para implantarla:

- Inducción sobre la aplicación del modelo al equipo de desarrollo de software.
- Aplicación del modelo en la organización.
- Acompañamiento 8 horas por semana, por 8 semanas.

Los agentes reguladores que podrían reconocer, avalar y auditar la aplicación de este modelo propuesto podrían ser (entre otros): Colegio de Ingenieros de Venezuela, PMI, IEEE, SEI, Sociedades de computistas e informáticos de cualquier país.

CAPÍTULO V. DESARROLLO

A continuación se presenta el desarrollo de la propuesta de investigación planteada en este trabajo especial de grado.

Interpretando a Boehm (1988): El modelo en espiral puede adaptarse a casos especiales, y proporciona una orientación en cuanto a la combinación de otros modelos que mejor se adaptan a una situación dada del software.

El foco de esta propuesta de aplicación del modelo, se centra en un modelo iterativo-incremental como lo es el modelo espiral, debido a que para cada iteración se irán resolviendo, o agregando nuevas funcionalidades, de acuerdo al resultado de las revisiones de requerimientos originales y funcionalidades necesarias que definan las partes interesadas.

La propuesta metodológica se hace robusta debido a que se centra en las actividades relacionadas en una secuencia lógica, que permite ubicar las actividades a realizar dentro del marco de los procesos definidos por el PMBOK (2008), para la gestión del tiempo.

El software se considera inestable cuando por distintos motivos deja de ejecutar o no ejecuta las funcionalidades para las que fue desarrollado, como se comenta en el capítulo uno, las causas y los síntomas son diversos.

Antes de desestabilizarse el software existieron riesgos que en la puesta en producción se activaron, y ya estando desestabilizado, se puede referir a ellos como "puntos de atención" ó "asuntos" a ser atendidos, por lo que en ese estado, un análisis profundo de riesgos no aplica, lo que sí aplica, es un análisis de las prioridades a asignar en cuanto a la atención a esos puntos pendientes, y así poder realizar una secuencia optima de las actividades a desarrollar en el proceso de estabilización.

Bajo esta taxonomía de proyecto, los ciclos de la espiral deben ser mucho más cortos, y por lo tanto se harán menos iteraciones, la mayor cantidad de trabajo se verá en la fase de desarrollo.

Las nuevas versiones de producto a entregar, contendrán las funcionalidades que hayan acordado las partes interesadas, y no se continuarán las iteraciones cuando éstas determinen que no es necesario continuar debido a que el producto se considera estable.

La Propuesta de la Aplicación del Modelo

A continuación se presenta la secuencia de procesos que conforman la aplicación que se propone en esta investigación.

Existen dos procesos que se presentan en la Gestión del Proyecto de manera particular (i e ii):

i. Revisión de la documentación de los procesos automatizados por el software

Estas actividades se realizarán sólo al inicio del Proyecto de Estabilización de Software.

Para quienes se encargarán de la estabilización del software, es necesario conocer muy bien los procesos que fueron automatizados, así como los requerimientos aprobados a ser cubiertos por el software desarrollado, si no existe la documentación correspondiente, se debe validar la lógica existente en el código fuente con los usuarios funcionales y proceder a documentar.

Entradas

- Documento de Visión y Alcance
- Documento de requisitos funcionales

Herramientas

- Juicio Experto
- Entrevistas con los clientes funcionales
- Entrevistas con los ingenieros de requisitos iniciales del proyecto
- Entrevistas con los desarrolladores del software

Salidas

- Entendimiento de los procesos involucrados en el software
- Entendimiento de los requisitos funcionales iniciales
- Actualización de los Documento de los Procesos automatizados y Documento de requisitos funcionales, o documento elaborado del mismo, con firmas de aceptación por las partes interesadas.

Cabe destacar, que los dos primeros puntos de la salida, son imprescindibles para poder continuar con el proceso siguiente, mientras el último punto, si no es concluido rápidamente, puede llevarse en paralelo con las actividades siguientes pero debe quedar cerrado antes de entrar al tercer cuadrante de la espiral (ver Gráfica 21).

ii. Controlar la planificación de la iteración

Se trata de una actividad indispensable. En proyectos de este tipo, hacer el seguimiento y control de cerca a las actividades es muy necesario. La persona que lleve el rol de Gerente de Proyecto, Líder de Proyecto, Líder de Desarrollo, Líder técnico, etc., debe llevar a cabo de manera permanente el control de la planificación de todo el proyecto de estabilización.

Entradas

- Salidas de todos los procesos del modelo
    - Planificación del proyecto
    - Cronograma del proyecto

Herramientas

- Software de administración de proyectos
- Software para "bug-tracking"
- Análisis de variaciones
- Nivelación de recursos (hardware, software, presupuesto, capital humano)
- Ajuste de adelantos y retrasos

Salidas

- Indicadores de desempeño del proyecto

  Entre otros se pueden citar:

  - Cantidad de Errores activos
  - Cantidad de Errores resueltos
  - Cantidad de Mejoras activas
  - Cantidad de Mejoras resueltas
  - Tiempos Estimados
  - Tiempos Reales
- Actualización de la documentación de los procesos y requisitos funcionales del proyecto
- Documento de Identificación de funcionalidades o mejoras pendientes
- Ajustes al proyecto

ITERACIÓN 1.

Los procesos a continuación, se presentan en casi todos los ciclos e iteraciones de la espiral.

1. Recopilación de "puntos de atención" presentados o requerimientos no cubiertos

En este proceso se colocará en un solo documento o sistema, todos los puntos de atención o requerimientos no cubiertos detectados.

Entradas

- Listado de "puntos de atención" detectados, que contiene la Identificación de funcionalidades o mejoras pendientes
- Documento de los procesos automatizados y Documento de requisitos funcionales, nuevos o actualizados

Herramientas

- Juicio experto

Salidas

- Listado de "puntos de atención" del proyecto

2. Evaluar y categorizar los "puntos de atención" detectados o requerimientos no cubiertos

Se deben evaluar los "puntos de atención" según una categoría o subcategorías previamente definidas.

Luego de categorizar, se debe analizar en cual(es) categoría(s) existe la mayor cantidad de "puntos de atención", lo que permitirá tomar la decisión acerca de por dónde iniciar a estabilizar.

Entradas

- Listado de "puntos de atención" del proyecto

Herramientas

- Juicio Experto
- Principio de Pareto
- Etiquetas de categorías. Dentro de evaluación se deberá identificar en qué Dentro de la clasificación podría utilizarse las siguientes categorías:
    - Nueva característica o funcionalidad
    - Mejora
    - Reporte
    - Rendimiento
        - BD.Bloqueos
        - BD.Lentitud
    - Error de funcionamiento
    - Excepción
    - Seguridad
    - Entre otras
- Mapas Mentales

Salidas

- Listado de "puntos de atención" del proyecto, actualizado
    - Ítems categorizados
    - Cantidad de "puntos de atención por categoría

3. Diagnóstico

Después de realizar los procesos anteriores, se debe emitir el diagnóstico técnico que permitirá, entre otras cosas:

- Decidir si técnicamente es factible estabilizar el software
- Establecer un primer estimado de tiempo para el proyecto, contrastándolo con los exigidos por el negocio

- Decidir si es más viable la adquisición de un software existente y adaptarlo a la empresa
- Analizar si financieramente es posible llevar a cabo el proyecto
- Revisar la posible utilización de los recursos existentes o adquirir por alguna vía nuevos (hardware, software, capital humano, bibliografía, etc.)

Aunque algunos puntos tienen cierto nivel de incertidumbre, el juicio experto del equipo puede determinar una estimación cercana a la realidad para cada uno, para poder emitir el diagnóstico.

Entradas

- Documento de Visión y Alcance
- Documento de requisitos funcionales
- Documento de los Procesos automatizados
- Listado de "puntos de atención" del proyecto
- Especificaciones técnicas de los recursos (hardware, software, capital humano)
- Presupuesto asignado

Herramientas

- Juicio Experto del equipo de Desarrollo

Salidas

- Factibilidad técnica del proyecto
- Factibilidad financiera del proyecto

Si es factible:

- Especificaciones de las habilidades necesarias del equipo de desarrollo
- Listado de recursos necesarios
- Estimaciones iniciales de tiempo

4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional

Si se llega a este hito dentro del ciclo, es porque el diagnóstico determinó que el proyecto es factible de llevar a cabo.

En este proceso se deberán establecer las relaciones entre las actividades del proyecto, así como las prelaciones entre ellas, siguiendo un orden lógico que permita al proyecto tener el mejor rendimiento en la relación "tiempo, costo y calidad"

Entradas

- Documento de requisitos funcionales
- Documento de los Procesos automatizados
- Listado de "puntos de atención" del proyecto (categorizados)

Herramientas

Se toman en cuenta las categorías para:

- Diagramación por precedencia
- Determinación de las dependencias

Salidas

- Actualización del Listado de "puntos de atención" del proyecto (categorizados, y secuenciados)

4.1. Evaluación de prioridades

En este proceso de deben tomar en cuenta dos aspectos:

Prioridades funcionales, son actividades a desarrollar que son fijadas por los usuarios funcionales, las cuales consideran necesarias para llevar a cabo su trabajo.

Prioridades técnicas, son actividades que deben llevarse a cabo debido a que sin ellas no podría avanzar el proyecto. Estas actividades podrían tener mayor peso que las prioridades funcionales.

Los criterios a seguir como un mínimo, deberían ser los siguientes:

1. Funcionalidad
2. Rendimiento
3. Interface usuario
4. Mejores prácticas
5. Rediseño

Entradas

- Listado de "puntos de atención" del proyecto (categorizados, y secuenciados)

Herramientas

- Matriz de gestión de Prioridades (Anexo 1)

Salidas

- Listado de "puntos de atención" del proyecto (categorizados, secuenciados y priorizados)

5. Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos

Este proceso es de suma importancia debido a que se debe determinar si el equipo de desarrollo de software con el que se dispone, realmente tiene las capacidades de afrontar un proyecto de éste tipo.

Debido a que se cuenta con un listado de "puntos de atención" del proyecto (categorizados, secuenciados, priorizados), y que se revisará con más detalle las habilidades de cada recurso sobre las categorías de las actividades y conociendo cuáles son las más críticas, este proceso determinará los cambios y acciones necesarias a efectuar en el equipo respecto a las habilidades del equipo y cómo sería la participación de éste.

De ser necesario se deberá proceder a incorporar por algún medio (bien sea por contratación directa o "outsourcing" de nuevos recursos. La cantidad dependerá de las habilidades técnicas con las que cuente el equipo, presupuesto disponible, y la rapidez con la que la organización necesite contar con el software estable. También debe revisarse la cantidad de equipos (hardware) y licencias de software disponible o a adquirir, como entrenamientos o capacitaciones necesarias.

Entradas

- Listado de "puntos de atención" del proyecto (categorizados, secuenciados y priorizados)

Herramientas

- Mapa de Calor de Habilidades del Equipo de Desarrollo
- Juicio Experto

Salidas

De acuerdo a los resultados del análisis, podrían ejecutarse algunas de las siguientes alternativas:

- Listado de acciones a tomar en cuanto a lograr compensar o reforzar las habilidades necesarias en el equipo
- Entrenamientos o capacitaciones necesarias para cerrar brechas detectadas
- Contratación de nuevo personal a incorporar al equipo
- Desincorporaciones de personal en el equipo
- Implementar estrategias de "outsourcing"
- Contratación de Consultoría externa para atención de actividades puntuales del proyecto
- Listado de "puntos de atención" del proyecto (categorizados, secuenciados, priorizados), incluyendo dentro de las secuencias, las actividades que se determinen en este punto que afectarán el cronograma del proyecto.

6. Estimación de los tiempos de cada actividad

En este proceso se obtendrá el tiempo aproximado de desarrollo de cada actividad, y ya habiéndose determinado previamente las secuencias y prioridades de las mismas, como resultado se podrá contar con el tiempo total que se requerirá para estabilizar el software con una incertidumbre muy baja.

Entradas

- Tiempos de entrega requeridos por el negocio
- Listado de "puntos de atención" del proyecto (categorizados, secuenciados y priorizados)

Herramientas

- Juicio experto para la estimación del tiempo de cada actividad, según desarrollador asignado
    - Mesa de trabajo con el equipo de desarrollo para estimar tiempos de actividades específicas
- Asignación de recursos
    - "Crashing" (asignar más recursos a las actividades)
    - "Fast-Tracking" (asignar actividades en paralelo, adición de desarrolladores)

Salidas

- Listado de "puntos de atención" del proyecto (categorizados, secuenciados, priorizados y con desarrolladores asignados)

7. Identificar las actividades que se entregarán en cada versión

Bajo un acuerdo entre las partes interesadas, se definirá el contenido de actividades de cada versión a ser entregada en cuanto a correcciones o mejoras que se desarrollarán, definiendo los "puntos de atención" que se incluirán en la versión en cuanto funcionalidad (de parte de los usuarios), y los que deben ser atendidos en cuanto a criterios técnicos (de parte de los desarrolladores de software)

Entradas

- Listado de "puntos de atención" del proyecto (categorizados, secuenciados, priorizados)

Herramientas

- Mesa de trabajo

Salidas

- Cantidad de versiones necesarias previas a la estabilización del software
- Listado de "puntos de atención" del proyecto indicando a qué versión pertenece cada uno

8. Desarrollo de la estabilización

Este proceso es el más laborioso y clave para el éxito de la estabilización, por lo tanto: El más crítico, así que el proceso de "Control de la Planificación" (ii), se deberá llevar a cabo de forma muy cercana al desenvolvimiento del proyecto. Identificar desviaciones, tomar decisiones y acciones para corregirlas será muy frecuente.

Al inicio de este proceso, se deberían haber terminado o se deberían estar llevando a cabo de manera paralela las actividades correspondientes a:

- Procesos de selección y reclutamiento de personal
- Negociación de personal de "outsourcing"
- Negociación de personal para consultorías
- Adquisición de otros tipos de recursos

En esta fase los desarrolladores podrían identificar nuevos "puntos de atención", los cuales deben agregarse al listado de éstos, para ser revisados en la próxima iteración, o ser atendidos de una vez debido a necesidades técnicas para poder continuar estabilizando.

Entradas

- Listado de "puntos de atención" del proyecto (categorizados, secuenciados, priorizados y con desarrolladores asignados)

- Documento de Visión y Alcance
- Documento de requisitos funcionales
- Documento de los Procesos automatizados

Herramientas

- Equipo de desarrolladores calificados
- Desarrollo de Software
- Herramientas de desarrollo de software
- Mesas de trabajo
    - Tormentas de ideas
- Discusiones técnicas
- Revisiones de bibliografía
- Pruebas Unitarias
- Pruebas de integración y calidad

Salidas

Incluye:

- Versión (grupo de funcionalidades) de software estabilizado y listo para ser probado
- Listado de "puntos de atención" del proyecto, se debe actualizar indicando cuales fueron atendidos

8.1. Pruebas Unitarias

Son llevadas a cabo por los propios Desarrolladores, deberán probar unitariamente cada uno de los "puntos de atención" que van cerrando para cerciorarse de su correcto funcionamiento. Se hacen pruebas de secciones específicas del código fuente del software. Si no pasan las pruebas deben ser atendidas inmediatamente.

Entradas

- Documento de Visión y Alcance
- Documento de requisitos funcionales

- Documento de los Procesos automatizados
- Listado actualizado de "puntos de atención" a ser atendidos del proyecto de la versión actual
- Versión (grupo de funcionalidades) de software estabilizado

Herramientas

- Desarrolladores
- Pruebas Unitarias

Salidas

- Listado actualizado de "puntos de atención" del proyecto, que incluirá:
    - Nuevos "puntos de atención" detectados
- Versión (grupo de funcionalidades) de software estabilizado

8.2. Pruebas de integración y calidad (QA)

Deben ser realizadas por personal técnico que no haya estado involucrado en el desarrollo del software. Involucran componentes, módulos, interfaces de usuarios, las relaciones entre éstos etc. El resultado de las pruebas no exitosas, debe ser retroalimentado al equipo de desarrollo para que las atienda y de solución.

No deberían realizarse más de dos iteraciones de estas pruebas.

Entradas

- Documento de Visión y Alcance
- Documento de requisitos funcionales
- Documento de los Procesos automatizados
- Listado actualizado de "puntos de atención" a ser atendidos del proyecto de la versión actual
- Versión (grupo de funcionalidades) de software estabilizado

Herramientas

- Personal técnico especializado en Pruebas de Calidad de Software
- Guiones de prueba

- Pruebas de caja blanca
- Pruebas de caja negra
- Pruebas de caja gris

Salidas

- Listado actualizado de "puntos de atención" del proyecto, que incluirá:
    - Nuevos "puntos de atención" detectados
- Documento técnico que indica los pasos a seguir para la puesta en producción (implementación)
- Versión (grupo de funcionalidades) de software estabilizado

9. Pruebas de calidad y aceptación por usuarios funcionales

Deben ser realizadas por los usuarios funcionales quienes son los que utilizan el software, preferiblemente quienes realizaron los requerimientos funcionales.

Este proceso tiene una importancia relevante, ya que al finalizar, se aceptaran diversos "puntos de atención" como resueltos y se dará el visto bueno a la versión que pasará a producción.

Cabe destacar, que los "puntos de atención" que no pasen esta prueba de calidad, se mantendrán activos en el Listado de "puntos de atención", para ser atendidos y revisar su prioridad en el próximo ciclo.

No deberían realizarse más de dos iteraciones de estas pruebas.

Entradas

- Documento de Visión y Alcance
- Documento de requisitos funcionales
- Documento de los Procesos automatizados
- Listado actualizado de "puntos de atención" a ser atendidos del proyecto de la versión actual
- Versión (grupo de funcionalidades) de software estabilizado

Herramientas

- Usuarios funcionales
- Guiones de prueba
- Pruebas de caja negra

Salidas

- Listado actualizado de "puntos de atención" del proyecto, que incluirá:
    - Nuevos "puntos de atención" detectados
    - "Puntos de atención" que no fueron cerrados por algún motivo, y que deben mantenerse en el listado para el próximo ciclo (versión)
- Documento de aceptación firmado por los funcionales, donde especifican cuales "puntos de atención" fueron cerrados, y que están de acuerdo en que pasen a producción en la versión actual.
- Versión (grupo de funcionalidades) de software estabilizado

10. Pase a producción de la versión

En este proceso se realiza la implementación de la versión en producción, y se considera a su vez como el cierre del ciclo actual de la espiral.

Entradas

- Versión (grupo de funcionalidades) de software estabilizado, probado y con visto bueno de parte de los usuarios funcionales para pasarlo a producción
- Visto bueno y ventana de tiempo aprobada por el comité de Control de Cambios para hacer el pase a producción
- Documento técnico que indica los pasos a seguir para la puesta en producción (implementación)
    - Incluye pasos para regresar el cambio, entre otros.

Herramientas

- Analistas de Sistemas que harán el pase a producción

Salidas

- Versión en producción lista para ser utilizada por usuarios funcionales

ITERACIÓN 2 A LA N

En las iteraciones de la espiral a partir de la número 2, se encuentran presentes todos los procesos de la iteración 1, excepto (i), (3) y (5):

i. Revisión de la documentación de los procesos automatizados por el software

3. Diagnóstico

5. Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos.

Por lo que la secuencia de procesos sería la siguiente, manteniendo las especificaciones dadas en este capítulo para cada uno de ellos:

1. Recopilación de "puntos de atención" presentados o requerimientos no cubiertos

2. Análisis y categorización de los "puntos de atención" detectados o requerimientos no cubiertos

Variante: sólo de nuevos "puntos de atención" identificados.

4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional

Variante: sólo si existen nuevos "puntos de atención" identificados

4.1. Evaluación de prioridades

6. Estimación de los tiempos de cada actividad

Variante: sólo de nuevos "puntos de atención" identificados

7. Identificar las actividades que se entregarán en cada versión

Variante: sólo de nuevos "puntos de atención" identificados

8. Desarrollo de la estabilización

9. Pruebas de aceptación por usuarios funcionales

10. Pase a producción de la versión

Criterios para finalizar del Proyecto de Estabilización del Software

Dentro de la gestión del proyecto de estabilización se deben contestar dos de las preguntas fundamentadas por Boehm (1988) en su modelo:

1. ¿Cómo salir de la espiral cuando conviene dar por concluido un proyecto antes de tiempo?
2. ¿Por qué la espiral finaliza tan abruptamente?

Ambas preguntas quedan cubiertas con la siguiente respuesta:

No se continuará con una próxima iteración si las partes interesadas se encuentran satisfechas con los resultados obtenidos, y pueden convivir con los "puntos de atención" que faltan por estabilizar (si es el caso).

Cómo se mide el costo según este modelo

Al igual que el Modelo Espiral Boehm (1988), el costo se mide en dos dimensiones:

- **Angular:** Indica el avance del proyecto de estabilización de software dentro de un ciclo.
- **Radial:** se refiere al incremento del costo del proyecto, ya que con cada nueva iteración se invierte más tiempo desarrollando.

Matriz de asignación de prioridades

Dentro de las estrategias a seguir esta la evaluación de prioridades (4.1), que se apoya en la siguiente "Matriz de Asignación de prioridades".

Por medio de este instrumento, se pueden evaluar las prioridades de los "puntos de atención" a ser atendidos, que se encuentra en el *Listado de "puntos de atención" del proyecto.*

Dicha matriz, luce de la siguiente manera:

Módulo	Punto de Atención / Requerimiento	Categoría	Capa	Impacto	Usabilidad	PRIORIDAD	Estado

Está compuesta por los siguientes campos:

- **Módulo**: El módulo del software al cual pertenece un "punto de atención"
- **"Punto de Atención/ Requerimiento"**: Falla identificada en el *Listado de "puntos de atención" del proyecto.*
- **Categoría**: La naturaleza que identifica la falla.
- **Capa**: Donde se encuentra la falla según el diseño de capas.
- **Impacto**: El nivel de importancia que tiene para el negocio el "punto de atención", Escala del 5 al 1, donde 5 es la mayor magnitud de impacto.
- **Usabilidad**: El porcentaje de uso dado por los usuarios funcionales. Incrementa de 0% a 100% de acuerdo a cuanto se use.
- **PRIORIDAD**: Campo calculado, definido por la siguiente formula:

$$Prioridad = (Impacto * Usabilidad)$$

- **Estado**: Es el estatus en cuanto a atención o gestión se le está dando a un "punto de atención".

Valores usados por cada campo de la Matriz de Prioridades

Tabla 3: Valores a utilizar en los campos de la matriz de prioridades

Impacto	Usabilidad	Estado	Categoría	Modulo	Capa
5	0%	1-Nuevo PA o Func	Nuevo	Mod1	IU
4	5%	2-En progreso	Mejora	Mod2	Log Negocios
3	10%	3-Rechazado	Reporte	Mod3	BD
2	15%	4-Feedback	Rendimiento	Mod4	
1	20%	5-Cerrado	Error		
	25%	6-Cancelado	Excepción		
	30%		Seguridad		
	35%				
	40%				
	45%				
	50%				
	55%				
	60%				
	65%				
	70%				
	75%				
	80%				
	85%				
	90%				
	95%				
	100%				

Criterio de Prioridades

El criterio para priorizar los "puntos de atención" a ser atendidos lo define el valor del campo "Prioridad", el "punto de atención" que tenga mayor valor debe ser atendido antes que el resto, pero tomando en cuenta las secuencias definidas en el proceso número cuatro del modelo (4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional).

Ejemplo de uso

Módulo	Punto de Atención /Requerimiento	Categoría	Capa	Impacto	Usabilidad	PRIORIDAD	Estado
Mod1	Eliminar "hardcode" en pag1	Error	Log Negocios	5	35%	1,75	2-En progreso
Mod2	Error en Stored Procedure AgregarProductoSucursal()	Excepción	BD	5	10%	0,5	2-En progreso

Figura 21: Ejemplo del uso de la Matriz de Prioridades

En el ejemplo se puede ver un listado de "puntos de atención" a ser atendidos que han sido clasificados, y asignado un "Estado".

Como se ve en el ejemplo el primer punto a atender, es el que tiene mayor prioridad, debido a que el producto del impacto y usabilidad definido para él (1.75), es mayor al segundo (0.5).

Nota: El archivo donde se ha creado este instrumento, lleva por nombre *Matriz_de_prioridades.xls*, y va adjunto a éste modelo propuesto.

CAPÍTULO VI. ANÁLISIS DE LOS RESULTADOS

El resultado del presente Trabajo Especial de Grado está representado en la aplicación de un modelo iterativo para la gestión del tiempo en proyectos de estabilización de software, obteniéndose la descripción de un modelo que identifica los procesos a seguir en una secuencia lógica dentro de iteraciones incrementales basadas en el modelo en espiral de Boehm (1988) (ver Figura 22: Modelo iterativo para la gestión del tiempo en proyectos de estabilización de software), proponiendo documentos a utilizar, así como herramientas que son citados en las entradas y salidas de cada proceso, y a su vez enmarcado dentro del proceso de la Gestión del Tiempo del PMBOK (2008), como se muestra en la siguiente tabla:

Tabla 4: Asociación entre los procesos definidos por la Gestión del Tiempo del PMBOK (2008) y la Aplicación del Modelo Iterativo propuesto

Asociación de los procesos definidos por		
Gestión del Tiempo (PMBOK2008)	**Modelo iterativo propuesto**	**Cua-drante**
No disponible	i. Revisión de la documentación de los procesos automatizados por el software	1
6.1 Definir las Actividades	1. Recopilación de "puntos de atención" presentados o requerimientos no cubiertos 2. Análisis y categorización de los "puntos de atención" detectados o requerimientos no cubiertos 3. Diagnóstico	1 2 3
6.2 Secuenciar actividades	4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional 4.1. Evaluación de prioridades	2 3
6.3 Estimar los recursos de las actividades	5. Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos.	3
6.4 Estimar la Duración de las Actividades	6. Estimación de los tiempos de cada actividad	3
No disponible	7. Identificar las actividades que se entregarán en cada versión	3
6.5 Desarrollar el Cronograma	8. Desarrollo de la estabilización 8.1.Pruebas Unitarias 8.2.Pruebas de integración y calidad 9. Pruebas de aceptación por usuarios funcionales 10. Pase a producción de la versión	4 4 4 4 4
6.6 Controlar el Cronograma	ii. Controlar la planificación de la iteración	1,2, 3,4

Siguiendo dicho modelo se podrá garantizar la gestión del tiempo de un proyecto de éste tipo, de manera metódica, ordenada, y manteniendo a las partes interesadas muy involucradas con el proyecto, hasta conseguir la estabilización del software.

Representación gráfica del modelo iterativo para la gestión del tiempo en proyectos de estabilización de software

El modelo de espiral de Boehm (1988) adaptado a proyectos de estabilización de software, queda representado de la siguiente manera:

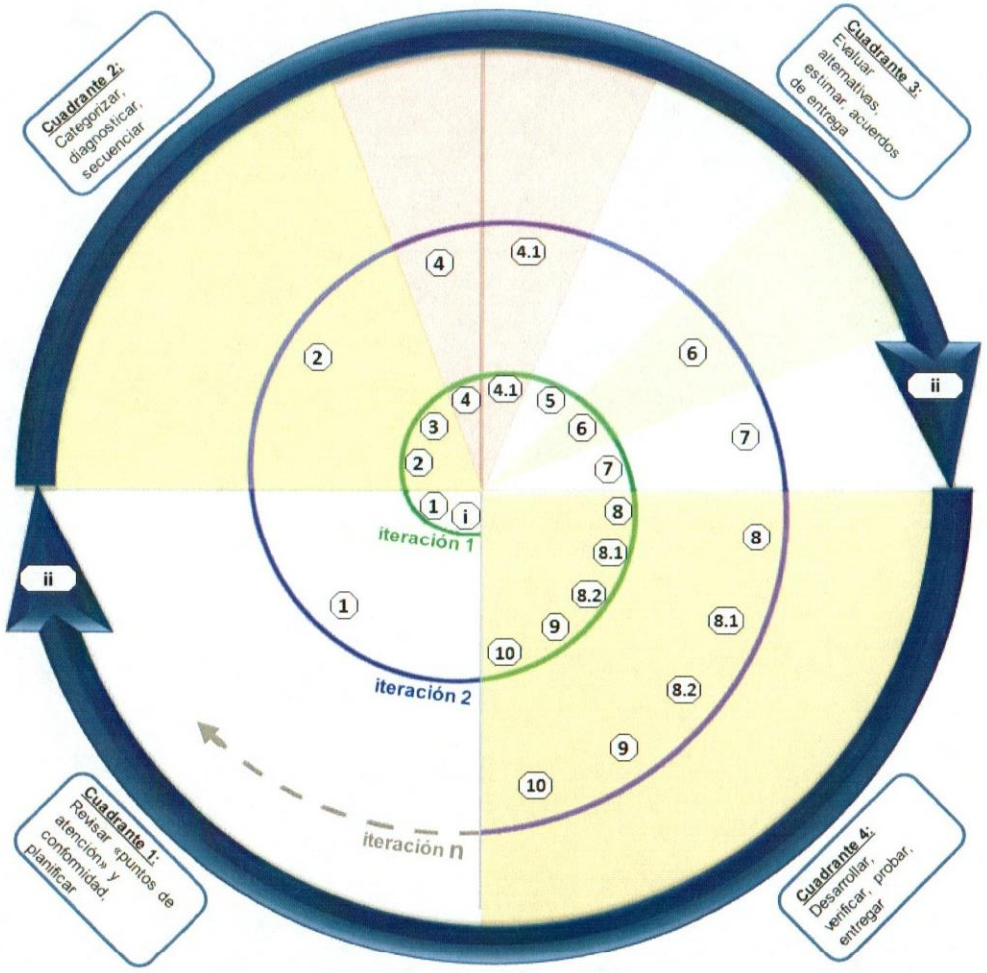

Figura22: Modelo iterativo para la gestión del tiempo en proyectos de estabilización de software.

Fuente: Gregorio Márquez (Propia)

Leyenda

Modelo iterativo propuesto
i. Revisión de la documentación de los procesos automatizados por el software
1. Recopilación de "puntos de atención" presentados o requerimientos no cubiertos
2. Análisis y categorización de los "puntos de atención" detectados o requerimientos no cubiertos
3. Diagnóstico
4. Secuenciar las actividades a desarrollar según un orden de prelación, bien sea técnico o funcional
4.1. Evaluación de prioridades
5. Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos.
6. Estimación de los tiempos de cada actividad
7. Identificar las actividades que se entregarán en cada versión
8. Desarrollo de la estabilización
8.1.Pruebas Unitarias
8.2.Pruebas de integración y calidad
9. Pruebas de aceptación por usuarios funcionales
10. Pase a producción de la versión
ii. Controlar la planificación de la iteración

Iteración 1

Iteración 2

Inicio próxima iteración

Gestión del tiempo(PMI, 2008)

- 6.1 Definir las Actividades
- 6.2 Secuenciar actividades
- 6.3 Estimar los recursos de las actividades
- 6.4 Estimar la Duración de las Actividades
- 6.5 Desarrollar el Cronograma
- 6.6 Controlar el Cronograma

En la figura anterior, puede observarse que la espiral es dividida en cuatro cuadrantes, que definen bien qué tipo de actividades se realizan en cada uno de ellos:

- Cuadrante 1: Revisar «puntos de atención» y conformidad, planificar.
- Cuadrante 2: Categorizar, diagnosticar, secuenciar.
- Cuadrante 3: Evaluar alternativas, estimar, acuerdos de entrega
- Cuadrante 4: Desarrollar, verificar, probar, entregar.

Realizándose sólo en el primer cuadrante y sólo en la primera iteración la "Revisión de la documentación de los procesos automatizados por el software" (Proceso "i"), y de igual forma sólo en la primera iteración se realizará el "Diagnóstico" General (Proceso "3"). En esta primera iteración, y no en otras iteraciones, solo se realiza la "Revisión de las habilidades de los desarrolladores disponibles, y se estiman los recursos necesarios asociados a éstos" (proceso "5").

En las iteraciones siguientes se ven bien definidos los procesos a ser llevados a cabo, uno de los más importantes es el proceso "Evaluación de prioridades" (Proceso "4.1"), que es el que define cuales son los entregables de la próxima versión.

Durante todo el proyecto, será permanente y constante el "Control de la planificación de cada iteración" (proceso "ii")

Las iteraciones se mantienen hasta que las condiciones sean favorables y las partes interesadas acuerden que el software está estable y se cierren los ciclos, y a su vez cerrando el proyecto.

Dinámica y desempeño aproximado de un proyecto de estabilización de software

En cuanto al criterio a definir cuándo detener el continuar con la espiral, tiene mucho sentido si se analiza la Figura 22, donde se expone lo que podría ser un ejemplo del desempeño aproximado de un proyecto de estabilización de software, resaltando las fases que definen un proyecto de MSF:

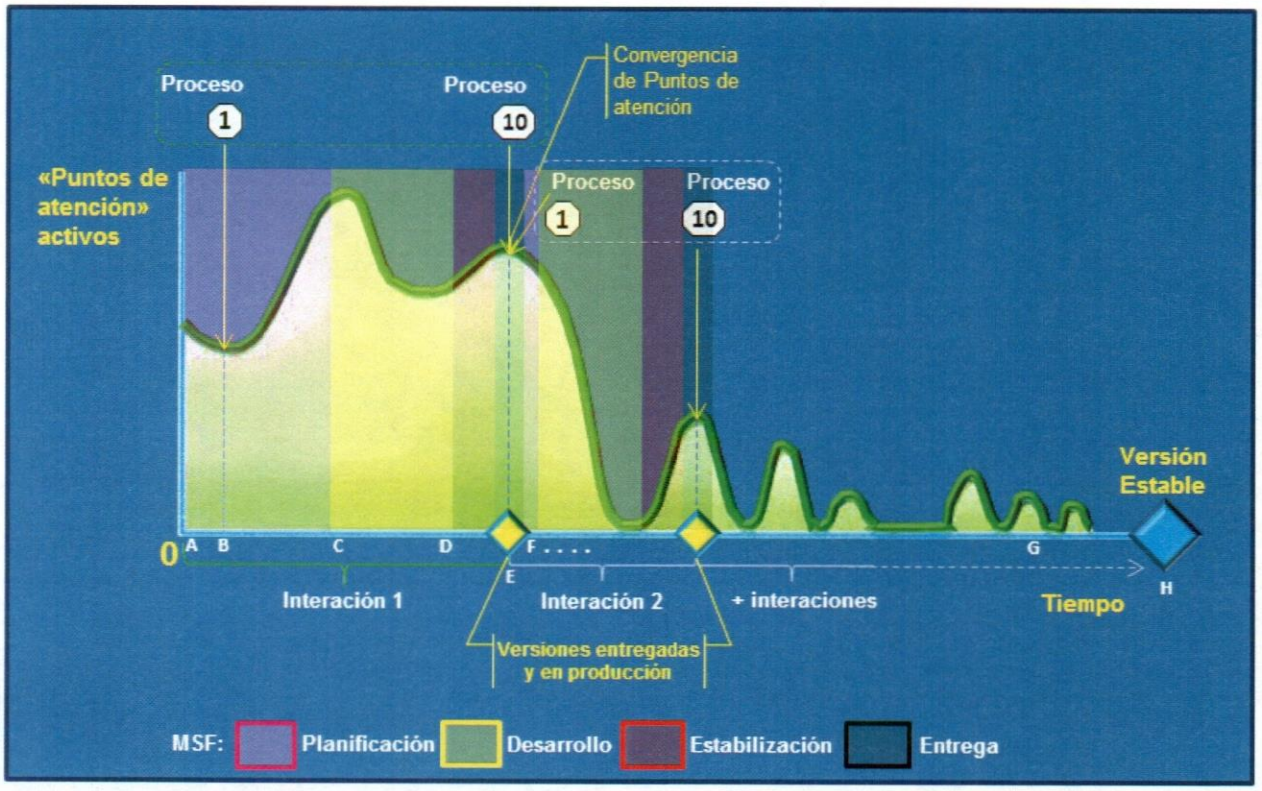

Figura23: Desempeño aproximado de un proyecto de estabilización de software
Fuente: Gregorio Márquez (Propia)

A. Seinicia el proyecto con una cantidad de "puntos de atención".
B. Al llegar al proceso 1, se han descartado algunos "puntos de atención" que no merecían estar dentro del listado.
C. Cuando inicia el desarrollo, se comienzan a identificar "puntos de atención" muy técnicos que no fueron detectados por los usuarios funcionales, y deben agregarse al listado de puntos de atención, lo que explica el crecimiento de la cantidad de puntos activos.
D. Se termina la fase de desarrollo y comienzan las pruebas y estabilización de la versión, donde se detectan nuevos "puntos de atención" que se agregan al listado.
E. Hito: Se entrega la versión acordada.
F. Inicia la nueva iteración, donde no se repiten todos los procesos que se siguieron al inicio del proyecto.

G. Continúan las iteraciones y la cantidad de "punto de atención" activos van disminuyendo.

H. Hasta acordar con las partes interesadas, que la versión entregada es estable, y con ella se cierra la última iteración y saliendo de la espiral.

CAPÍTULO VII. EVALUACIÓN DEL PROYECTO

A continuación, se presenta la evaluación de todas las interrogantes y objetivos que fueron planteados al inicio de este trabajo en el Capítulo I.

En primer lugar se revisa si se les dieron respuesta a las interrogantes:

Tabla 5: Evaluación del cumplimiento del problema

Preguntas		Se respondió en un
Formulación del problema	¿Cuál es la secuencia lógica de procesos que pueden seguirse para lograr una efectiva gestión del tiempo en proyectos de estabilización de software?	100%
Sistematización del problema	¿Se está desarrollando una versión estable única que agrupe todas las funcionalidades requeridas, o se ha planteado la aplicación de un modelo para entrega de versiones?	100%
	¿Se está desarrollando una versión estable única que agrupe todas las funcionalidades requeridas, o se ha planteado un modelo de entrega de versiones?	100%
	¿Se cuenta con una propuesta de aplicación de un modelo definido que permita hacer una correcta gestión del tiempo de proyectos de este tipo?	100%
	¿Cómo debería ser la dinámica y desempeño de un proyecto de estabilización de software?	100%

Se revisan a continuación si el objetivo general y los específicos que se plantearon al inicio de este trabajo fueron alcanzados:

Tabla 6: Evaluación del cumplimiento de los objetivos.

	Objetivos	Evaluación	Referencia
General	Aplicar el modelo iterativo en la gestión del tiempo en proyectos de estabilización de software.	100%	Capítulo V
Específicos	Definir los criterios de estabilización en proyectos de software.	100%	Capítulo V, Criterios para finalizar del Proyecto de Estabilización del Software
	Analizar el método iterativo como referencia de uso en la gestión de tiempo en proyectos de software.	100%	Inicio Capítulo V Capítulo II (teoría)
	Formular las estrategias de aplicación del método iterativo en la gestión de tiempo en proyectos de software.	100%	Capítulo V: - Item 4.1: Evaluación de prioridades - Matriz de asignación de prioridades

Estos resultados pueden ser comprobados al verificar el producto obtenido, revisando las referencias presentadas en la tabla 5.

CAPÍTULO VIII. CONCLUSIONES Y RECOMENDACIONES

Conclusiones

- Al ser un modelo iterativo orientado a la gestión de prioridades, se concluye que uno de los aspectos fundamentales de su éxito radica en que el equipo que lo utilice tenga la experiencia y habilidad para:
    - Analizar y categorizar de los "puntos de atención" (proceso 2)
    - Evaluar de prioridades (proceso 4.1)

- La introducción de la aplicación de un modelo iterativo de proyecto en una organización requiere una cuidadosa coordinación y control.

- Con la aplicación de éste modelo iterativo propuesto, en la que se abordan los proyectos en colaboración (con todas las partes interesadas), se entregan resultados incrementalmente y reconocibles a un ritmo mucho más rápido, para proyectos de estabilización de software, es más efectivo en comparación con una metodología rígida que obliga a los usuarios a cumplir con todos los documentos y pasos donde no es una prioridad la entrega rápida e incremental de funcionalidades del software.

- El rechazo al cambio es un proceso natural dentro de una organización, no se podrá ganar la aceptación de todo el mundo al implementar una nueva metodología de proyecto, siempre alguien va a identificar las piezas que faltan.

- Realizar una evaluación de la capacidad actual de la organización, identificar qué áreas clave del proceso son necesarias mejorar para obtener el máximo beneficio, a continuación, hacer los cambios apropiados.

- Cuando se trabaja con personal de ventas y marketing que venden modelos metodológicos de los proyectos a los clientes, hay que asegurarse de que el líder

del proyecto esté involucrado antes de comprometer alguna fecha, se conoce cómo muchos proyectos fracasan cuando las promesas se hacen en forma aislada sólo para obtener la aceptación de un cliente, por personas que no tienen el nivel técnico, gerencial y experiencia respecto a este tipo de proyectos.

- El criterio que permite decidir cuándo cerrar el ciclo dejando de iterar debido a que se identifica que el software es estable, es llevado a cabo por todas las partes interesadas.

- Dar acertadamente las prioridades a los "puntos de atención" definidos, es la clave de que se hagan menos ciclos, ahorrando en tiempo y dinero, y hará que se obtengan éxitos tempranos que reforzaran el apoyo en el proyecto y entregar versiones cada vez más estables a la organización, haciendo a ésta más productiva.

Recomendaciones

- Este modelo iterativo propuesto se podría extender a las fases de:
    - Pruebas unitarias,
    - Pruebas funcionales,
    - Puesta en producción y
    - Soporte post producción de un proyecto

- Se debe incluir el apoyo de soluciones informáticas, para el uso de las herramientas propuestas en cada proceso.

- Se recomienda que esta aplicación de modelo iterativo sea por Líderes o Gerentes de Proyectos, quienes tengan experiencia y antecedentes gerenciando proyectos de desarrollo de software.

- Se debe mantener una comunicación muy estrecha entre todos los interesados y mantener negociaciones constantes para llegar a acuerdos en cuanto a los puntos de atención a ser distribuidos en cada versión.

- La motivación del equipo de trabajo y otras partes interesadas en proyectos de este tipo que generan mucho estrés, es necesario compartir los éxitos y destacar los hitos alcanzados y los beneficios que se obtendrán.

REFERENCIAS BIBLIOGRÁFICAS

- Barrios, M. (2006). *Manual de trabajos de grado de especialización y maestría y tesis doctorales.* Caracas: Fondo Editorial de la Universidad Pedagógica Experimental Libertador.
- Bennatan, E. (2006). *Catastrophe Disentanglement: Getting Software Projects Back on Track.* Addison Wesley Professional.
- Bittner, K., & Spence, I. (2006). *Managing Iterative Software Development Projects.* Addison Wesley Professional.
- Boehm, B. W. (Mayo de 1988). A Spiral Model of Software Development and Enhancement. *Computer, vol. 21, no. 5*, 61-72.
- CIV, C. d. (s.f.). *Código de Ética Profesional.* CIV.
- Congreso de la República de Venezuela. (1993). *Ley de derecho de autor.* Caracas: Gaceta Oficial de la República de Venezuela.
- Hernández, R., Fernández, C., & Baptista, P. (2006). *Metodología de la Investigación.* Mexico, DF: McGraw-Hill.
- Méndez, E. M. (2006). *Diseñar un modelo de evaluación de metodologías para el desarrollo de software.* Caracas: UCAB.
- Merchán, N. (2006). *Diseño de un proceso de generación de estimados de tiempo para una empresa de Desarrollo de aplicaciones.* Caracas: UCAB.
- Microsoft. (2003). *Microsoft Solutions Framework v3.0.* Recuperado en Abril de 2011, de Microsoft Download Center: http://www.microsoft.com/downloads/en/details.aspx?FamilyID=a71ac896-1d28-45a4-880c-8b0cc8265c63

- PMI. (2008). *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK).* Newtown Square: Project Management Institute, Inc.
- PMI, J. D. (2006). *Project Management Institute Código de Ética y Conducta Profesional.* PMI.
- Real Academia Española . (2009). *Diccionario de la lengua española.* Real Academia Española.
- Stapleton, J. (1997). *Dynamic Systems Development Methodology, from Dynamic System Development Method .* Addison-Wesley.
- Talavera, M. (2008). *Aplicación de la gestión del tiempo para los proyectos de una empresa desarrolladora de software.* Caracas: UCAB.