



**UNIVERSIDAD NACIONAL
AUTÓNOMA DE MÉXICO**

**FACULTAD DE ESTUDIOS SUPERIORES
ACATLÁN**

**GUÍA METODOLÓGICA PARA LA ADMINISTRACIÓN DE
PROYECTOS DE SOFTWARE ORIENTADO AL MODELO DE
MADUREZ DE CAPACIDAD INTEGRADO NIVEL DOS**

SEMINARIO DE TALLER EXTRACURRICULAR

QUE PARA OBTENER EL TÍTULO DE

LICENCIADO EN MATEMÁTICAS APLICADAS Y COMPUTACIÓN

PRESENTA

OMAR RAMIREZ VILCHIS

Asesor: JUAN TORRES LOVERA

Febrero 2009



Universidad Nacional
Autónoma de México



UNAM – Dirección General de Bibliotecas
Tesis Digitales
Restricciones de uso

DERECHOS RESERVADOS ©
PROHIBIDA SU REPRODUCCIÓN TOTAL O PARCIAL

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

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



Contenido

Dedicatorias	4
Agradecimientos	5
Introducción.....	6

Capítulo 1. El modelo de madurez en la administración de proyectos de software.

1.1 La administración de proyectos de software	8
1.2 El modelo de madurez de capacidad integrado	22
1.3 Evolución de CMMI	26
1.4 Niveles de capacidad y de madurez	29
1.5 Metodologías de desarrollo	34
1.5.1 Marco de trabajo de soluciones de Microsoft.....	36
1.6 Caso práctico FINASIST	38
Conclusiones.....	42

Capítulo 2. Planificación estratégica de las áreas de proceso.

2.1 Planeación estratégica.....	43
2.2 Áreas de proceso	47
2.2.1 Organización del equipo de proyecto.....	48
2.2.2 Administración de requerimientos.....	51
2.2.3 Planificación de proyectos	62
2.2.4 Monitoreo y control	68
2.2.5 Medición y análisis.....	71
2.2.6 Administración de la configuración	75
2.3 Indicadores.....	77
Conclusiones.....	79



Capítulo 3. Implementación y evaluación del modelo.

3.1 Guía metodológica para la administración de proyectos de software	80
3.2 Apego al proceso	91
3.3 Impacto del modelo	111
3.4 Implementación del modelo	121
3.5 Satisfacción del cliente.....	126
Conclusiones.....	129
Conclusiones finales	130
Anexo planeación de escenarios futuros	133
Bibliografía	154



Dedicatorias.

A mis padres: Armando Ramírez Colín y Amalia Vilchis de Ramírez, por enseñarme a siempre esforzarme y a nunca darme por vencido, a ellos que siempre estuvieron a mi lado y me dieron ánimo para seguir adelante. A quienes nunca podré pagar todos sus desvelos.

A mi esposa por que vive conmigo mis triunfos y mis fracasos. A ella que me ha comprendido y apoyado durante mi carrera.

A mis hermanos por creer en mí y por su cariño.

Porque con sus esfuerzos y sacrificios, logré el triunfo que hoy les brindo.



Agradecimientos.

*A mi Universidad Nacional Autónoma de México,
por darme la oportunidad de aprender y forjarme
como profesional.*

*A mi asesor: Juan Torres Lovera, por
su paciencia y dedicación para la
realización de esta Tesis.*

*A mis sinodales: Luz María Lavín Alanís, Hugo
Reyes Martínez, Gerardo Roldán Ceballos, Liliana
Sandoval Luna, por sus valiosas aportaciones y
comentarios.*

*A mi jefe: Por brindarme la oportunidad
de concluir una etapa más de mi vida.*



Introducción.

La industria del desarrollo de software ha presentado desde sus orígenes, problemas durante el ciclo de vida de los proyectos, los más comunes son: Que no se entregan a tiempo, no se ajustan al presupuesto inicial, no cumplen con las especificaciones, el código es confuso lo que dificulta la gestión y evaluación. Las principales causas son: problemas en los requerimientos, falta de planificación, no se administran los riesgos y no existe un monitoreo, supervisión y control de los proyectos. A lo largo del tiempo han surgido distintas áreas: la ingeniería y arquitectura del software, y metodologías o modelos de desarrollo, que de forma aislada pretenden resolver estos problemas.

El presente trabajo establece una guía, para administrar proyectos de software, que implementa las mejores prácticas administrativas, definidas en las áreas de proceso del nivel dos del Modelo de Madurez de Capacidad Integrada. El modelo establece la guía a través de metas, prácticas y actividades específicas de cada una de sus áreas y la metodología de Marco de trabajo de Soluciones de Microsoft, proporciona el sustento para ejecutar las prácticas. La integración entre el modelo y la metodología proporciona un aspecto innovador en la administración de proyectos de software.

Un proyecto administrado es un proyecto con una mejor calidad, que minimiza al máximo los errores, identifica desviaciones a tiempo, lo que permite tomar acciones correctivas, logra entregas a tiempo, establece métricas que apoyan la toma de decisiones y la identificación de mejoras, permite repetir los éxitos, incrementa la satisfacción del cliente, apoya la rentabilidad de las organizaciones y establece las bases para un futuro exitoso.



El principal motivo de la selección del modelo radica en que cuenta con un enfoque de sistemas y es integrado, lo que implica impactos no solo en una parte de la organización si no en toda. El modelo tiene la premisa “La calidad de un sistema o producto es influenciada por la calidad de los procesos usados para desarrollarlo y mantenerlo”.

La guía está dirigida tanto a personas que tienen la oportunidad de administrar por primera vez proyectos de desarrollo de software, como a personas con experiencia interesadas en contar con un marco de referencia que permita aplicar estándares internacionales, que deseen mejorar la calidad de los proyectos de software.

En el capítulo uno se define la administración de proyectos de software y el modelo de madurez, su evolución, sus distintos niveles, una breve introducción a los diferentes tipos de metodologías de desarrollo tanto formales e informales, se exponen los aspectos básicos de la metodología seleccionada y presenta el caso práctico FINASIST una empresa dedicada al desarrollo de un producto de software orientado a la industria financiera en donde se aplicará el modelo. En el siguiente capítulo se elabora una guía, a través de la planeación estratégica de las áreas de proceso del modelo, que incluye: La organización del equipo de proyecto, la administración de requerimientos, la planificación de los proyectos, el monitoreo y control, la medición y análisis, así como la administración de la configuración. Por último se aborda la parte de implementación de la guía, se define el mecanismo de evaluación, al cuantificar el apego al proceso, se establece el impacto directo del modelo a la problemática presentada y se muestran los resultados de la implementación del modelo dentro del caso práctico.



Capítulo 1.

El Modelo de madurez en la administración de proyectos de software.

1.1 La administración de proyectos de software.

La palabra **Administración** se forma del prefijo *ad*, entre cuyos múltiples significados están los de participación en, la causa o el motivo por el cual se hace algo, sirve para indicar el oficio o cargo, y el verbo *Minister* vocablo compuesto de *Minus*, comparativo de inferioridad, y del sufijo *ter*, término de comparación; expresa subordinación u obediencia; el realiza una función bajo el mando de otro; el que presta un servicio a otro.

La etimología da la idea de que ésta se refiere a una función que se desarrolla bajo el mando de otro; de un servicio que se presta.

De las definiciones dadas por los principales autores en Administración, se pueden deducir sus elementos básicos:

V. Clushkov: “Es un dispositivo que organiza y realiza la transformación ordenada de la *información*, recibe la información del objeto de dirección, la procesa y la transmite bajo la forma necesaria para la gestión, realizando este proceso continuamente”.¹

¹ <http://www.monografias.com/trabajos7/admi/admi.shtml>, 21 de Diciembre del 2007



Guzmán Valdivia I: “Es la *dirección* eficaz de las actividades y la colaboración de otras personas para obtener determinados resultados”.²

J. D. Money: “Es el *arte* o técnica de dirigir e inspirar a los demás, con base en un profundo y claro *conocimiento* de la naturaleza humana”.³

Peterson and Plowman: “Una técnica por medio de la cual se determinan, clarifican y realizan los propósitos y *objetivos* de un grupo humano particular”.⁴

Henry Fayol dice que “Administrar es prever, organizar, mandar, coordinar y controlar”.⁵

A Reyes Ponce: “Es un conjunto sistemático de reglas para lograr la máxima *eficiencia* en las formas de estructurar y manejar un organismo social”.⁶

Para los fines del presente, se usará la siguiente definición de Administración: “La acción de planear, proveer recursos y coordinar esfuerzos organizada y sistemáticamente”.⁷

Proyecto proviene del latín *projectum* “Lanzarse hacia adelante”, viene de pro, que denota algo que precede la acción y de jacere, tirar.

² IDEM

³ IDEM

⁴ IDEM

⁵ IDEM

⁶ IDEM

⁷ Colunga Dávila Carlos, Modelos Administrativos, México, Panorama, 1995, Pág. 22



Un proyecto es un conjunto de esfuerzos temporales dirigidos a lograr un objetivo específico por medio de una serie particular de tareas interrelacionadas y la utilización eficaz de recursos.

En el área de computación, existe la palabra inglesa **Software** para referirse a los programas de cómputo *Soft* en inglés significa blando y *Ware* artículo u objeto, por lo tanto el Software, es el elemento suave, blando que permite a una computadora funcionar. La traducción es la de programa de cómputo. La palabra *Programa* proviene del griego **Pro** significa delante y **gramma** escritura, un escrito que indica los pormenores de algo, o que busca los resultados de algo. Por otra parte computación proviene del latín *computare* que significa calcular o contar. Ambos programa de cómputo o de computación es un documento en donde se encuentran los escritos que permiten calcular o contar a efecto de que, a través de determinadas operaciones se puedan obtener ciertos resultados.

La administración de proyectos significa “Planear el trabajo y después trabajar el plan”⁸; Es la aplicación de conocimientos, habilidades, técnicas y herramientas a las actividades de un proyecto, lo que implica primero establecer un plan y después llevarlo a cabo con el fin de satisfacer, cumplir el objetivo del proyecto.

El ciclo de vida de los proyectos de desarrollo de software según Microsoft está compuesto de las siguientes etapas:

Concepción. Descripción de los objetivos y limitaciones del proyecto, se identifica el equipo. El objetivo es la creación de una visión compartida del proyecto entre todos los actores del proyecto.

⁸ Gido Jack, Administración exitosa de Proyectos, México, Thomson, 2007, Pág. 10.



Definición de las metas del negocio, análisis de la empresa y el problema, se identifica el alcance y lo que no se incluirá en la solución. Se desarrolla un concepto de solución de acuerdo a los requerimientos identificados. Evaluación del riesgo y plan de mitigación.

Planificación. El equipo determina que va a desarrollar y los planes de cómo crear la solución, se prepara la especificación funcional, se crea el diseño de la solución, planes de trabajo, estimaciones de costos, involucra análisis de los requerimientos, arquitectura, cronograma del proyecto, administración del plan de riesgo y plan de proyecto general.

Desarrollo. El equipo del proyecto crea la solución, elaboración y documentación del código que implementa la solución, y la infraestructura de solución, creación de un prototipo, componentes de la solución y la solución, especificación de pruebas y casos de pruebas.

Estabilización. El equipo realiza la integración, la carga, y pruebas de la versión beta de la solución, despliegue de la solución en un ambiente de pruebas.

Despliegue. El equipo despliega la solución tecnológica, y componentes de sitio, estabiliza el despliegue, transfiere el proyecto al área de operaciones y soporte y obtiene la aprobación final del proyecto por parte del cliente.

La falta de entendimiento de los requerimientos y necesidades de negocio del cliente, la falta de documentación de los acuerdos adquiridos, el no establecer claramente los alcances del proyecto y el no dejar documentado lo que no incluye la solución son los problemas clásicos durante la etapa de concepción.



Durante la fase de desarrollo es común que no se tome el tiempo para diseñar un prototipo de la solución, en los casos que si se llega a desarrollar este no es presentado al cliente, por lo que no existe una retroalimentación por parte de este y el cliente solo conoce la solución hasta que es entregada; Adicionalmente los desarrolladores no se toman el tiempo para elaborar un diseño de la solución.

Se inicia de inmediato el proceso de codificación sin pensar con anterioridad, la arquitectura de la solución, o la mejor forma de desarrollarlo, ni los elementos que pueden ser reutilizados, otro problema en la fase de desarrollo es la falta de documentación del código, por lo que solo la persona que lo desarrolla, es la que lo conoce, y es capaz de mantenerlo, en muchas ocasiones se asume que el desarrollador tiene conocimientos de negocio.

Debido a no aplicar tareas de monitoreo, control, seguimiento y auditorias por parte del responsable del proyecto durante el proceso de desarrollo, no se identifica si el proyecto está en tiempo, o las desviaciones que puedan existir, se identifica al final del proyecto que no se va a terminar en tiempo pero ya es demasiado tarde para tomar acciones que puedan ayudar.

La falta de documentación que defina todos los requerimientos, el prototipo, la funcionalidad, ocasiona que las pruebas sean realizadas sin un conocimiento claro de lo que se debe de probar.

Debido a lo antes mencionado existe una dificultad de escribir programas libres de errores, fácilmente comprensibles, verificables.

Los cambios a los que se someten un programa para ser adaptado a las necesidades de los usuarios, la falta de herramientas que permitan estimar de manera exacta, antes de comenzar el proyecto, el esfuerzo que se necesitara para desarrollar un programa.



Provoca complicaciones al estimar cuánto tiempo llevará un proyecto, cuánto personal será necesario. Debido a esto último los plazos que se fijan normalmente no se cumplen. Del mismo modo, en muchas ocasiones el personal asignado a un proyecto se incrementa con la esperanza de disminuir el plazo de ejecución. Las aplicaciones de hoy en día son programas muy complejos, inabordables por una sola persona.

La industria del software experimenta una crisis creciente de problemas, los principales son:

- Los proyectos no se terminan en plazo.
- Los proyectos no se ajustan al presupuesto inicial.
- Software que no cumple con las especificaciones.
- Código confuso que dificulta la gestión y evolución del proyecto.
- El software no es fiable y necesita de un mantenimiento permanente.
- El software es imposible de mantener, carece de claridad y no se puede modificar ni mejorar.
- Asignación de recursos inexacta.
- Escasez de personal calificado en un mercado laboral de alta demanda.
- Tendencia al crecimiento del volumen y complejidad de los productos.
- Errores de codificación.



Esto no es nuevo existe desde los inicios del desarrollo de software, al principio era desarrollado por solo una persona, la cual conocía la necesidad, la forma de resolverla y contaba con los conocimientos técnicos para desarrollar la solución. Con el transcurso del tiempo y debido a la complejidad de los proyectos, los equipos de desarrollo se han conformado por un grupo de personas interdisciplinarias con distintos conocimientos, habilidades y formaciones.

Las organizaciones o equipos de desarrollo centran sus esfuerzos en el proceso de construcción del software, preocupándose por los elementos tecnológicos a implementar, dentro del producto.

Estas organizaciones son capaces de producir productos funcionales que pueden ser comercializados y que permiten la subsistencia de la organización. Generalmente estos logros se obtienen por esfuerzos aislados, por el sacrificio de uno o un grupo de individuos comprometidos. Tales organizaciones aplican procesos informales. Tienen poco énfasis en la definición de sus procesos e institucionalización de estos. Dejan en segundo plano las herramientas y técnicas de administración, las metodologías, modelos y procesos existentes para el desarrollo de software. Esto se debe a la presión con respecto a los tiempos de entrega de los proyectos, y a un constante cambio en los requerimientos y prioridades.

Si estas organizaciones sufren modificaciones en sus ambientes: crecimiento acelerado de clientes o proyectos, incorporación de nuevos recursos, salida de personal clave, se agudiza la problemática antes mencionada. Lo que lleva a la dirección a preguntarse, el porqué de esta situación, si antes con menos personal no sucedía lo mismo, los proyectos se entregaban con mejores tiempos, una simple junta era capaz de transmitir la necesidad y de dejar claro los requerimientos y necesidades del cliente y el software generado no tenía tantos errores.



Dentro de los elementos que son capaces de controlar los responsables de la administración de proyectos se encuentran juntas con los expertos del negocio, las estimaciones de desarrollo y cronogramas, los diseños funcionales y técnicos, las pruebas unitarias, las políticas, procedimientos, planeación, monitoreo, control, medición e historial de cambios.

Aquellos elementos que quedan fuera del alcance de los administradores son la calidad de los requerimientos, cronograma de otras áreas, conocimiento del negocio por parte de la gente de desarrollo, la cantidad de recursos, errores latentes, prioridades, necesidades del cliente, comunicación.

Existen restricciones por parte de los clientes: Tiempos de entregas, procesos de liberación, restricciones de costo, grado de definición de procesos operativos y necesidades descritas en los requerimientos, elementos legales operativos que deban incorporarse. Normalmente dentro de las organizaciones no se cuenta con información histórica cuantificable de las desviaciones de los proyectos, el tipo de desviación, la causa o responsable lo que afecta la toma de decisiones.

Uno de los objetivos es contar con un proceso de desarrollo, que incluya las actividades más comunes en cualquier organización de este giro como son: Nuevos proyectos, modificación o mantenimiento a productos existentes y la corrección de errores. La definición de este proceso debe abarcar el ciclo completo de desarrollo que involucra la afectación de todas las áreas dentro de la organización. Es de suma importancia el involucrar a los responsables de cada área durante la definición de este proceso, con la finalidad de generar la participación de cada uno de ellos y la adquisición de los compromisos particulares. A través de la participación se logra un menor rechazo al proceso y una motivación a la integración de las áreas de la organización.



Durante la definición del proceso se deben de seguir las siguientes disposiciones:

- Especificación de roles y responsables.
- Establecer el flujo.
- Definición de actividades.
- Entradas y salidas.

Otro de los objetivos a perseguir en la administración de los proyectos de desarrollo de software, es lograr terminar los desarrollos en tiempo y costo de acuerdo a lo definido con el cliente, mediante la implementación de prácticas administrativas como:

- Planeación y seguimiento del proyecto.
- Creación de un cronograma.
- Revisiones del estado del proyecto, avance planeado contra avance real.
- Llevar a cabo acciones correctivas.

Un tercer objetivo es la generación de información que apoye la toma de decisiones, mediante el seguimiento a los proyectos se podrá documentar las desviaciones que se presenten, con la finalidad de identificar posibles áreas de mejora. Se deberá de realizar las siguientes actividades:

- Documentar todas las desviaciones a la planeación del proyecto.
- Determinar la naturaleza de la desviación.
- Documentar los errores del proyecto, durante su construcción, liberación y uso por parte del cliente.



La mala calidad de los requerimientos puede ser mitigada mediante la definición e institucionalización de los procesos de desarrollo, en los cuales se debe de establecer claramente: Las actividades, los responsables, la documentación de entrada y salida de cada actividad y los elementos mínimos de cada requerimiento, el análisis permite profundizar en los requerimientos, es donde surgen dudas o inconsistencias y es aquí donde se puede tener una última retroalimentación por parte del cliente con el objetivo de que el producto final cubra sus necesidades y le sea de utilidad.

La entrega al cliente depende de varias áreas adicionales al desarrollo, debido a la falta de injerencia sobre estas, se requiere la implementación de prácticas administrativas como: Planeación, monitoreo, control y medición no solo durante las etapas de las cuales se es responsables, si no dentro de todo el ciclo de vida del desarrollo, lo que permite coordinar la fecha de entrega a los clientes. Mediante la implementación de herramientas de control y seguimiento se genera una base de conocimientos con las causas más comunes de las desviaciones de los proyectos lo que generara información para la toma de decisiones.

Las estrategias a seguir de acuerdo a todos los temas antes mencionados son:

- Definición e institucionalización de los procesos de desarrollo.
- Implementación de prácticas administrativas.
- Recopilación de información de las desviaciones de los proyectos.

La solución planteada en el presente trabajo a la problemática expuesta que da continuidad a las estrategias identificadas por lo elementos controlables y no controlables es. **La implementación de prácticas administrativas**, en los procesos de desarrollo de software.



La implementación de prácticas administrativas apoya el desarrollo de software de calidad. Mediante la administración de requerimientos, planeación del proyecto, monitoreo, control, medición y análisis del proyecto. Al desarrollar software de calidad se incrementara la rentabilidad de la organización.

El beneficio fundamental de la implementación de técnicas de administración de proyectos es tener un cliente satisfecho. Terminar todo el alcance del proyecto con calidad, a tiempo y dentro del presupuesto.

El concepto primordial en este trabajo es la administración; Centrada en el desarrollo de proyectos de software, que pretende resolver los problemas antes expuestos que afectan la calidad del software. Esto es un problema de tipo de control y seguimiento.

Este trabajo está dirigido a aquellas personas involucradas en la administración de proyectos de software, ya se han líderes de proyectos, gerentes ó directores de sistemas, a todos aquellos que participen en la toma de decisiones, encaminadas a la disolución de los problemas antes planteados. Tanto para personas con experiencia que desean implementar prácticas administrativas o para aquellos profesionistas que tienen la oportunidad de dirigir por primera vez a un equipo de desarrollo de software. Proporciona un marco de trabajo que sirve de guía para la administración de proyectos de software.

Si los proyectos son exitosos ¡Todos ganan!, el responsable tiene la satisfacción de saber que su esfuerzo fue exitoso, lo que mejora su imagen. El ser miembro de un equipo exitoso, tiene la satisfacción de estar en uno ganador. No sólo se contribuyó al éxito, sino que además se amplió el conocimiento y mejoró sus habilidades. Si está interesado en administrar será capaz de tomar responsabilidades adicionales.



La planeación y la comunicación son cruciales para la administración. Evitan que ocurran problemas o reducen al mínimo su impacto sobre el logro del objetivo.

Tomarse el tiempo para desarrollar un plan bien diseñado antes de iniciar el proceso es vital para el éxito del mismo. Se tiene que tener un objetivo definido, un resultado o producto esperado, un alcance, planeación y costos aceptados por el cliente.

La clave para el control eficaz es el medir el avance real y compararlo con el avance planeado de manera regular, y aplicar acciones correctivas de inmediato. Una vez que concluye el ciclo de vida, debe evaluarse su desempeño para saber qué podría mejorarse si se realizara algo similar en el futuro.

Actualmente las compañías quieren desarrollar productos y servicios mejores, más rápidos y baratos. Casi todas las organizaciones han encontrado cada vez más compleja la construcción de estos.

Las organizaciones tienen que ser capaces de administrar y controlar estos complejos desarrollos y procesos de mantenimiento. La solución de estos problemas requiere la participación integrada de toda la empresa. El manejo efectivo de los recursos de la organización es crítico para el éxito del negocio.

En el mercado actual, existen modelos de madurez, estándares, metodologías y directrices que pueden ayudar a las organizaciones a mejorar su manera de trabajar. Sin embargo, la mayoría de las mejoras se centran en una parte de la empresa y no tienen un enfoque de sistemas. Al centrarse, en la mejora de un área de una empresa, estos modelos promueven la división y los obstáculos que existen en las organizaciones.



El siguiente paso es identificar que prácticas administrativas, debe de ser implementadas, la respuesta es CMMI, el modelo de madurez de capacidad integrado, el cual contiene las mejores prácticas definidas para los proyectos de desarrollo de software.

El CMMI, parte de la premisa que la calidad de un sistema depende, de la calidad del proceso utilizado en su elaboración. El modelo se basa en la definición y mejora de procesos, tiene un retorno de inversión alto. “Es posible lograr un retorno de inversión de 5:1 a partir de las horas invertidas ”⁹.

El CMMI es un conjunto de modelos integrados, que afectan a todos los procesos de desarrollo de la organización, lo que permite mejoras a nivel organización.

El modelo de madurez cuenta con varios niveles, el nivel seleccionado es el dos, el definido como administrado, en el cual el proceso se institucionaliza, es un proceso en ejecución, que se ha planificado y ejecutado de acuerdo a una política, utiliza recursos adecuados para producir salidas controladas, es monitoreado, controlado, revisado y evaluado en términos de su adherencia a su descripción de proceso.

Mediante la implementación del nivel dos del modelo de madurez se obtendrán los siguientes beneficios:

- Entrega en plazos de los proyectos, por lo que los costos son los definidos con el cliente.

⁹ Dennis R. Goldenson, Demonstrating the impact and Benefits of CMMI, United States, SEI, 2003, Pág. 11



- La solución desarrollada cubre los requerimientos del cliente, satisface sus necesidades y es funcional para él.
- Favorece la definición de la arquitectura de la aplicación, desarrolla código libre de errores, estructurado que facilita su mantenimiento.
- La definición, apego al proceso y la documentación promueve el trabajo eficiente y eficaz, además de que permite la incorporación ágil de nuevos recursos.
- El CMMI es un estándar internacional para la industria de desarrollo de software, promueve la confianza en los clientes actuales y apoya la adquisición del producto por nuevos clientes.
- Utilidad en costos al evitar el gasto que representa la corrección de errores, que implica tiempo de gente de todas las áreas, atención a clientes, analistas, desarrolladores, probadores.
- Minimiza el riesgo de no terminar otros proyectos por desviaciones causadas por errores e incremento en proyectos al destinar menos tiempo a errores, lo que genera mayores ingresos.
- Satisfacción total del cliente, al entregar productos fiables, en tiempo y costo.

La implementación del modelo de madurez, requiere de la toma de decisiones a los niveles más altos de la dirección:

- Apoyo en la institucionalización y definición del proceso, por la afectación a nivel organizacional.
- Asignación de recursos que implementen las actividades administrativas.



1.2 El modelo de madurez de capacidad integrada.

El Modelo de Madurez de Capacidad Integrada, CMMI por sus siglas en inglés Capability Maturity Model Integration, es un modelo de madurez de mejora de procesos para el desarrollo de productos y servicios. CMMI para desarrollo consiste de las mejores prácticas dirigidas al desarrollo y mantenimiento aplicado a productos y servicios. Estas prácticas cubren el ciclo de vida del producto desde su concepción hasta su liberación y mantenimiento. El énfasis es sobre el trabajo necesario para construir y mantener el producto total.



Figura 1. "Las tres dimensiones críticas"¹⁰

Los tres aspectos críticos en el que las organizaciones normalmente se centran: Gente, procedimientos y métodos, y herramientas y equipos.

¹⁰ Chrissis Mary Beth, CMMI Guidelines for Process Integration and Product Improvement, United States, Addison Wesley, 2007, Pág. 4



Lo que mantiene todo integrado, es el proceso, usado en la organización. Este permite adaptar la forma de hacer negocios, dirigir el crecimiento, establecer una forma de cómo hacer las cosas y asignar los recursos eficientemente. Este no dice que la gente y la tecnología no son importantes. La tecnología avanza a una velocidad vertiginosa, en un mundo dinámico, en constante cambio, centrado en los procesos de infraestructura que permitan afrontar los retos futuros. El proceso incrementa la productividad de las personas y permite ser más competitivos.

La importancia de la calidad de los procesos y que estos sean eficientes y eficaces ayudan a las organizaciones y al personal a que cumpla con los objetivos para trabajar inteligentemente. La eficacia de los procesos proporciona un medio para la implementación de nuevas tecnologías que se ajusten a los objetivos de la organización.

En 1930, Walter Shewhart comenzó a trabajar en la mejora de procesos con sus principios de control estadístico de calidad, estos principios fueron refinados por W. Edwards Deming, Phillip Crosby, Joseph Juran, Watts Humphrey, Ron Radice, y otros, lo que extendió estos principios aun más e iniciaron a aplicarlos a software en su trabajo en IBM y en el SEI (Software Engineering Institute, Instituto federal estadounidense de investigación y desarrollo fundado en 1984 para desarrollar modelo de evaluación y mejora en el desarrollo del software).

Humphrey's book, Administración de los procesos de software, provee una descripción de los principios básicos y conceptos en los cuales el modelo de madurez de capacidad está basado.



El SEI ha tomado la premisa en la administración de procesos, “La calidad de un sistema o producto es influenciada por la calidad del proceso usado para desarrollarlo y mantenerlo”¹¹.

Las cinco principales ideas que inspiraron el modelo CMMI y el proceso de evaluación son:

- 1.- Planeación, seguimiento, y administración del cronograma.
- 2.- Definición de requerimientos y control de configuración.
- 3.- Proceso de evaluación.
- 4.- Aseguramiento de calidad y mejora continua.
- 5.- Mejora evolutiva.

La única manera para administrar complejas operaciones es con una descripción detallada y planes precisos.

La primera asignación de Watts S. Humphrey dentro del SEI fue la de mejorar el Software para la fuerza aérea de los Estados Unidos de Norte América. Definir una manera para evaluar la capacidad de las organizaciones, para determinar si usan las mejores prácticas y técnicas de administración en sus proyectos de desarrollo. Esta consistió en elaborar un cuestionario de 85 preguntas que cubrían:

¹¹ Chrissis Mary Beth, CMMI Guidelines for Process Integration and Product Improvement, United States, Addison Wesley, 2007, Pág. 5



- Planificación de proyectos.
- Seguimiento de proyectos.
- Administración de cronograma.
- Administración de requerimientos.
- Control de la configuración.
- Aseguramiento de la calidad.
- Procesos continuos de mejora.

Para clasificar los resultados, se agruparon las preguntas en un marco de madurez. Esto se convirtió en la primera versión de CMMI. Los niveles indican las zonas de riesgo, guían el proceso de mejora al describir un conjunto mínimo de actividades necesarias. El marco de madurez establece las directivas a seguir pero no la forma en que deben de ser ejecutadas. Para controlar el trabajo complejo y preciso, cada uno debe administrar detalladamente y realizar planes precisos y cada uno debe de medir y gestionar la calidad. CMMI no define la forma detallada de hacer el trabajo de desarrollo, se centra en lo que se debe de hacer, las actividades que se deben de realizar y las áreas de proceso a ejecutar. La necesidad es la de comprender estos principios y prácticas en las organizaciones de desarrollo. CMMI se centra en la mejora de los procesos en una organización. El SEI creó el primer CMM diseñado para organizaciones de Software en 1995. El valor del enfoque en los procesos de mejora ha sido confirmado a través del tiempo. Las organizaciones han experimentado un aumento en la productividad, calidad, tiempos de ciclo, apego a planes y presupuestos más exactos y predecibles, lo que afecta directamente la calidad del software desarrollado.



1.3 Evolución de CMMI.

Poco después de la creación del SEI, la fuerza aérea de los EE.UU. solicitó la identificación de las principales prácticas que las organizaciones de desarrollo tenían que realizar para entregar sistemas fiables.

Para 1991, estas prácticas y mediciones reforzadas a través de un enfoque de mejora, maduro en el modelo de madurez de capacidad de software (SW-CMM).

El éxito de este modelo para una disciplina de ingeniería llevó a similares esfuerzos para otros elementos de desarrollo.

El interés en este tipo de modelos para procesos de ingeniería de sistemas llevó a la creación de dos modelos en 1994. El primero fue el de ingeniería de sistemas CMM creado por Enterprise Process Improvement Collaboration (EPIC), con la participación del SEI. El segundo modelo, capacidad de sistemas de ingeniería y métodos de evaluación o SECAM, fue creado por International Council on Systems Engineering (INCOSE). Cuatro años después estos dos modelos se fusionaron exitosamente en Electronic Industries Alliance (EIA), el resultado de un esfuerzo conjunto de EIA, EPIC, e INCOSE se generó el estándar provisional 731.

En 1996, un hermano del SW-CMM fue creado para cubrir las principales prácticas en adquisición de Software, el modelo de madurez de capacidad de adquisición de Software o SA-CMM. Asimismo, comenzaron los trabajos en el SEI para producir una actualización para el SW-CMM, y para producir un modelo que contenga las prácticas de ingeniería en un producto de desarrollo integrado. El departamento de defensa de los EE.UU. determinó que esos esfuerzos debían fusionarse en un modelo integrado, que se conoce con el nombre del modelo de Madurez de Capacidad Integrado (CMMI).



La misión inicial del equipo de CMMI fue la de combinar tres modelos:

- El modelo de madurez de capacidad para Software (SW-CMM).
- El modelo de capacidad de ingeniería de sistemas (SECM).
- El modelo de madurez de capacidad integrada para productos de desarrollo (IPD-CMM).

La combinación de estos modelos dentro de un único marco se destino para la búsqueda de los procesos de mejora en las organizaciones. Estos tres modelos fueron seleccionados debido a su amplia difusión en las organizaciones de software. CMMI es el resultado de la evolución del SW-CMM, el SECM y el IPD-CMM.

La versión v1.02 se libero en el 2000 la cual incluía ingeniería de sistemas, ingeniería de software y procesos de desarrollo integrados. Para diciembre del mismo año se liberaba una versión inicial para los procesos de adquisición denominada v1.02d. Tomo un año reunir los resultados de la liberación inicial antes de refinar el material que se deseaba estabilizar. El primero de estos modelos fue liberado en diciembre del 2001 la versión v1.1, a esta se agregaron prácticas de adquisición en el 2002. Esta versión marco las bases. Véase Fig.2

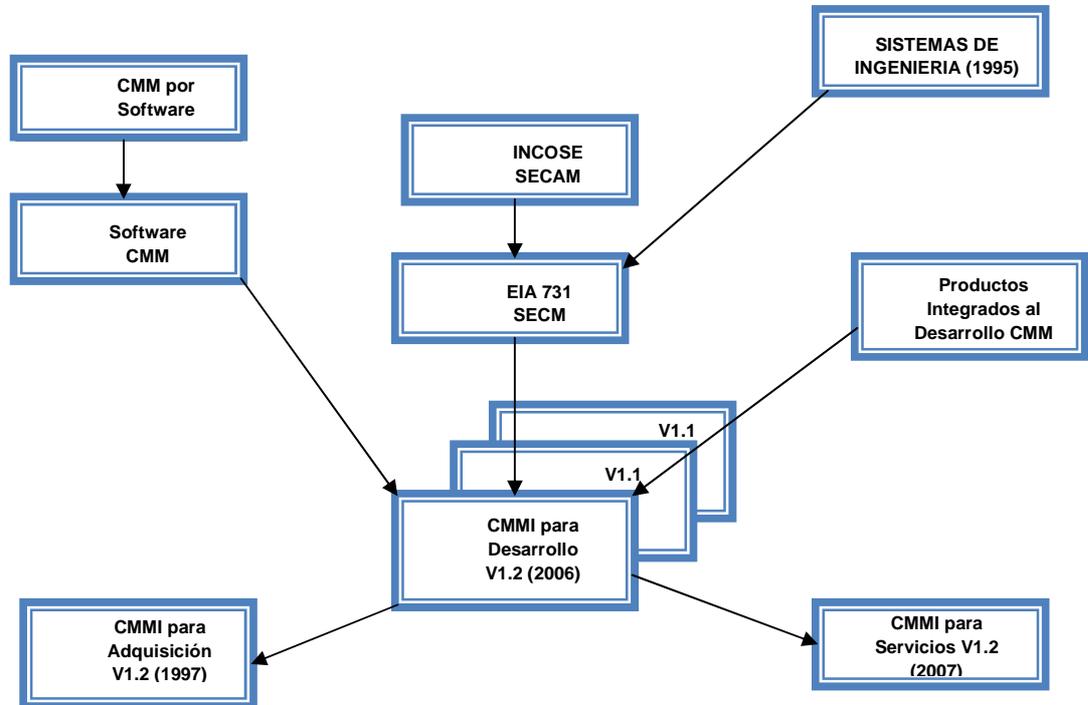


Figura 2. "La Historia de CMMs"¹².

El modelo se ha convertido en un estándar para el desarrollo de sistemas de Software, y demostró su valor en el proceso de auditorías de la gestión pública, un ejemplo de esto es la ley Sarbanes-Oxley. Para obtener el máximo beneficio de la adopción y la aplicación de CMMI, debe aplicarse a toda la estructura de desarrollo de la organización y para esto la última versión v1.2 se denominó CMMI para desarrollo (CMMI-DEV). La arquitectura de la versión v1.2 ha sufrido una ligera modificación para dar cabida a dos adicionales aplicaciones, CMMI para adquisición (CMMI-ACQ) y CMMI para servicios (CMMI-SVC).

¹² Chrissis Mary Beth, CMMI Guidelines for Process Integration and Product Improvement, United States, Addison Wesley, 2007. Pág. 15



1.4 Niveles de capacidad y de madurez.

Todos los modelos de CMMI reflejan niveles de capacidad en su diseño y contenido. Un nivel de capacidad consiste de metas genéricas, relacionadas a prácticas genéricas de áreas de proceso, las cuales pueden mejorar los procesos organizacionales. Al satisfacer las metas y prácticas genéricas para cada nivel de capacidad, se obtienen los beneficios de mejora de procesos.

Existen seis niveles de capacidad que son las siguientes:

0. Incompleto.
1. Ejecutado.
2. Administrado.
3. Definido.
4. Administrado Cuantitativamente.
5. Optimizado.

Nivel cero. Incompleto.

Un proceso incompleto, es un proceso que no se realiza o se realiza parcialmente. Una o más de las metas específicas de las áreas de proceso no son cubiertas y no existen metas genéricas para este nivel y no hay razón para institucionalizar el cumplimiento parcial de procesos.



Nivel uno. Ejecutado.

Un proceso realizado, es un proceso que cumple con los objetivos específicos del área de procesos. Apoya y realiza el trabajo necesario para producir productos. Las mejoras pueden ser perdidas con el tiempo si estas no son institucionalizadas.

Nivel dos. Administrado.

Un proceso administrado, es algo ejecutado que cuenta con la infraestructura básica para soportar el proceso. Planeado y ejecutado de acuerdo con una política. Emplea a personas que tengan habilidades adecuadas para producir salidas controladas, afecta a involucrados relevantes, es monitoreado, controlado y revisado; es evaluado en términos de su adherencia a su descripción de proceso. La disciplina del proceso reflejado por el nivel 2 ayuda a asegurar la existencia de prácticas que se mantienen durante tiempos de estrés.

Nivel tres. Definido.

Un proceso definido, es algo administrado adaptado a los estándares de la organización, y contribuye con productos de trabajo, mediciones, e información de mejora organizacional.

Una crítica diferencia entre el nivel dos y el tres es el alcance de los estándares, descripciones, y procedimientos. Se establecen claramente criterios de entradas, actividades, roles, mediciones, medidas de verificación, salidas, y criterios de éxito. La administración es proactiva bajo el entendimiento de las relaciones de las actividades y el detalle de las mediciones de los procesos.



Nivel cuatro. Administrado cuantitativamente.

El proceso es caracterizado por ser administrado cuantitativamente, definido, controlado, la calidad y el desempeño son entendidos en términos estadísticos y otras técnicas cuantitativas administradas a lo largo del ciclo de vida.

Nivel cinco. Optimizado.

Es caracterizado por una optimización, administrado cuantitativamente, mejorado con base al entendimiento de las causas comunes de variaciones externas. Es optimización el rango de desempeño a través de mejoras incrementales e innovadoras. Cada nivel de madurez es soportado por las características de las áreas de procesos implementadas en los niveles anteriores. Los niveles de madurez representan una evolución del proceso de mejora para la organización entera.

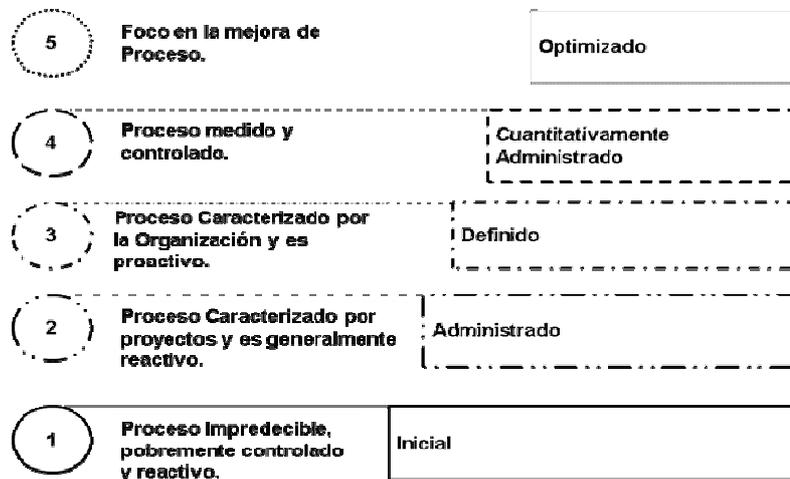


Figura 3. "Modelo de Madurez"¹³



A continuación se describen las áreas de proceso de cada uno de los niveles.

Nivel dos.

- Administración de requerimientos.
- Planeación del proyecto.
- Monitoreo y control del proyecto.
- Administración de acuerdos con proveedores.
- Medición y análisis.
- Aseguramiento de calidad de procesos y producto.
- Administración de la configuración.

Nivel tres.

- Desarrollo de requerimientos.
 - Solución técnica.
 - Integración de producto.
 - Verificación.
 - Validación.
 - Enfoque en el proceso de la organización.
 - Definición del proceso de la organización.
 - Planeación del proyecto.
-



- Formación organizacional.
- Administración integrada de proyectos.
- Administración de riesgo.
- Decisión y análisis de resolución.

Nivel cuatro.

- Desempeño de los procesos de la organización.
- Administración cuantitativa de proyectos.

Nivel cinco.

- Innovación y desarrollo organizacional.
- Análisis causal y de resolución.

El modelo de madurez establece en sus distintos niveles las áreas de proceso a ejecutar, pero no establece la forma de hacerlo, con la finalidad de buscar un sustento metodológico que permita desarrollar dichas áreas y actividades con las mejores prácticas a continuación se presenta las tendencias de las metodologías de desarrollo actuales y una breve descripción de las más relevantes desde mi punto de vista muy personal.



1.5 Metodologías de desarrollo.

Conjunto de procedimientos, técnicas, herramientas y soporte documental que ayuda a los desarrolladores a realizar un nuevo software. La metodología indica cómo hay que obtener los distintos productos parciales y finales.

Desarrollar software de calidad depende de un sin número de actividades y etapas, donde el impacto de elegir la metodología para un equipo en un determinado proyecto, es trascendental, para el éxito del producto.

La metodología debe encajar en el equipo, guiar y organizar las actividades que lleven a cumplir los objetivos.

Existen dos grandes enfoques, metodologías tradicionales y metodologías ágiles.

Las metodologías tradicionales imponen una disciplina de trabajo sobre el proceso de desarrollo del software, con el objetivo de conseguir un software más eficiente y predecible. Promueven la planificación detallada de prácticamente todo el desarrollo de software a largo plazo y recalcan el uso de documentación durante todo el ciclo del proyecto. Son predictivas y basadas en procesos.

Entre las principales metodologías tradicionales están RUP y MSF, centran su atención en la documentación, planificación y procesos (plantillas, técnicas de administración, revisiones) llevan una documentación y el cumplimiento del plan definido en la fase inicial del proyecto. Una característica importante, son los altos costos al implementar un cambio y la falta de flexibilidad en donde el entorno es volátil.



En los últimos años ha aparecido un nuevo grupo de metodologías, las denominadas metodologías ágiles. Proponen procesos que se adaptan y progresan con el cambio, orientadas al personal más que orientadas al proceso, intentan trabajar con la naturaleza del personal asignado al desarrollo. Ponen mayor importancia en la capacidad de respuesta a los cambios y a mantener una buena relación con el cliente. Se basan en dos aspectos puntuales, el retrasar las decisiones y la planificación adaptativa.

Para los clientes esta flexibilidad será una ventaja competitiva y porque estar preparados para el cambio significa reducir sus costos.

La planificación adaptativa permite estar preparados para el cambio ya que se ha introducido en el proceso de desarrollo, consiste en tomar decisiones a lo largo del proyecto, esto para transformarlo en un conjunto de proyectos pequeños. Esta planificación a corto plazo permitirá tener software disponible para los clientes y además aprender a realizar la planificación más sensible.

En el presente trabajo se adoptará una metodología tradicional, se parte de la importancia de la definición de procesos y en la necesidad de predicción de los proyectos, la metodología seleccionada es la de Microsoft, debido a que es esta la que más se apega al proceso actual de la organización que representa el caso de estudio.



1.5.1 Marco de trabajo de soluciones de Microsoft (MSF - Microsoft Solution Framework).

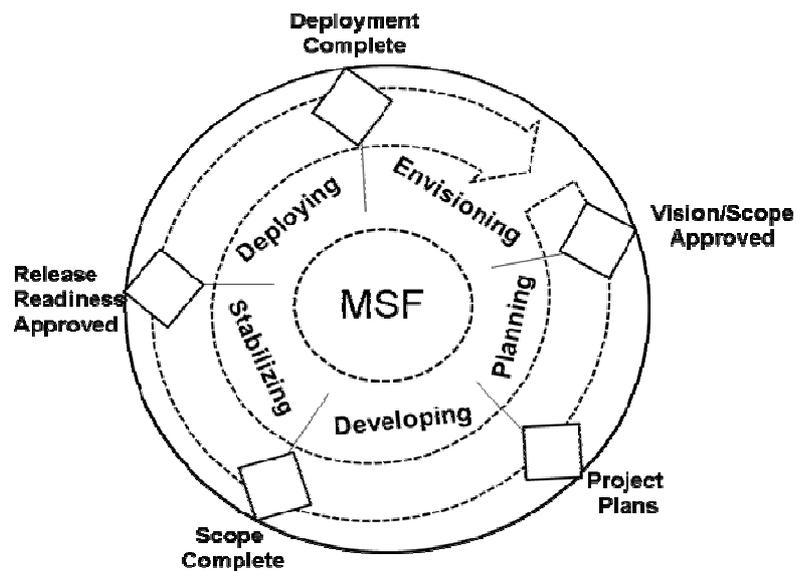


Figura 4. "Ciclo de Vida Microsoft Solution Framework (MSF)"¹⁴.

MSF es un compendio de las mejores prácticas en cuanto a administración de proyectos se refiere. Más que una metodología rígida, MSF es una serie de modelos que puede adaptarse a cualquier proyecto de tecnología de información, es flexible e interrelacionada, se centra en los modelos de proceso y de equipo, y en segundo plano las elecciones tecnológicas.

MSF tiene las siguientes características:

- Adaptable. Puede ser usado como un mapa.

14 Analyzing Requirements and Defining Microsoft .NET Solution Architectures Course 2710 MSDN Training, pág. 13



- Escalable. Puede organizar equipos pequeños 3 o 4 personas, ó también, proyectos que requieren 50 personas o más.
- Flexible. Aplicable en cualquier ambiente de desarrollo.
- Tecnología Agnóstica. Puede usarse para desarrollar soluciones basadas sobre cualquier tecnología.

Fases

- Visión y alcances: Consiste en la unificación del equipo detrás de una visión común.
- Planificación: Se realiza el proceso de diseño de la solución, y prepara los planes de trabajo. Las estimaciones de costo y cronogramas de los diferentes entregables del proyecto.
- Desarrollo: Construcción de los componentes y documentación del código.
- Estabilización: Pruebas sobre la solución.
- Implantación: Implanta la tecnología base.

Modelo de roles.

Los equipos organizados bajo este modelo son pequeños y multidisciplinarios, en los cuales los miembros comparten responsabilidades para mantenerse enfocados en el proyecto que está en desarrollo. MSF se compone de varios modelos encargados de planificar las diferentes partes implicadas en el desarrollo de un proyecto: Modelo de Arquitectura del proyecto, Modelo de equipo, Modelo de proceso, Modelo de gestión de riesgo, Modelo de diseño de proceso y finalmente el modelo de aplicación.



1.6 Caso práctico FINASIST.

FINASIST es una empresa dedicada al desarrollo del software, propietaria de un producto denominado SmartControl ®, una Suite Financiera orientada al sector financiero y a las organizaciones auxiliares del crédito (Arrendadoras, Sofoles, Uniones de crédito y Sofomes).

SmartControl es una solución integral para la administración del crédito, la cual incluye los procesos administrativos y operativos en un solo producto.

FINASIST fue fundada en 2003 con el objetivo de desarrollar un producto innovador, que ofreciera a sus clientes una herramienta compuesta de distintos módulos, que apoyaran en cada una de las fases de la administración de cartera, es una aplicación integral, donde los módulos se encuentran interconectados entre sí, pero que al mismo tiempo el cliente pueda seleccionar la adquisición de los módulos que requiera su operación.

En enero del 2007, se contaba con tres clientes usuarios de la aplicación, para finales del mismo año el número de clientes paso de tres a diez clientes.

Este crecimiento se reflejó en, rotación de personal y en carga de trabajo debido a las adecuaciones al producto, por la forma particular de operar de cada uno de sus clientes, debido a esto se presentaron los siguientes problemas.

- Los proyectos no se terminan en plazo.
- Software que no cumple con las especificaciones.
- Dependencia de los desarrolladores.
- Falta de documentación.



- Código inmantenible que dificulta la gestión y evolución del proyecto.
- El software no es fiable y necesita de un mantenimiento permanente.
- Asignación de recursos inexacta.
- Tendencia al crecimiento del volumen y complejidad de los productos.

FINASIST no cuenta con información histórica que permita determinar las tendencias de las desviaciones o identificar las causas que los generan.

Actualmente existe una creciente inconformidad por parte de los clientes que tienen la percepción de que toda liberación tiene errores, no cumple con sus requerimientos, no les es útil, no es lo que solicitaron y que todos los proyectos se entregan fuera de las fechas planeadas, Y más aún esta percepción es confirmada dentro de las áreas internas que tienen el contacto con el cliente. FINASIST ha tomado algunas acciones con el objetivo de solucionar este problema.

Dentro de las acciones realizadas fue la separación del área de desarrollo en cuatro divisiones, una división encargada de los nuevos requerimientos, otra de la corrección de errores, una de infraestructura, adicionalmente se consolido un área responsable de pruebas.

Pero se genero una tendencia en desarrollo a destinar a los recursos con mayor experiencia y conocimiento a resolver los errores latentes de la aplicación, con lo cual se descuido los nuevos desarrollos, por lo que estos tenían más errores que después tenían que ser corregidos lo que propicia un circulo vicioso difícil de disolver. Otra de las acciones emprendidas fue la contratación de un responsable de toda el área de desarrollo que coordinara las divisiones generadas anteriormente. Esta acción fracaso debido a la falta de pro actividad e involucramiento del personal contratado en las actividades diarias.



Actualmente FINASIST fusiono las divisiones de desarrollo en una sola área responsable tanto de los nuevos requerimientos y la corrección de los errores de la aplicación.

FINASIST se ha dado cuenta de la necesidad de definir sus procesos y políticas, que guíen al personal hacia los objetivos comunes además de motivar la integración de las distintas áreas, a través de la comprensión de la dependencia que existe entre ellas.

El personal responsable de la administración de proyectos de software ha identificado la necesidad de buscar e implementar las mejores prácticas y herramientas de administración de proyectos que permitan generar software de calidad. Los distintos niveles de la organización están conscientes de las deficiencias administrativas y se promulgan hacia una mayor organización en las distintas áreas.

Debido a la creciente competencia en el mercado de soluciones tecnológicas para la industria crediticia FINASIST necesita incorporar estándares internacionales como CMMI que ayuden a generar confianza en los clientes y permitan ser más competitivos.

La decisión de seleccionar el nivel dos del modelo de madurez es consecuencia del desconocimiento de las mejores prácticas administrativas aplicadas al desarrollo del software. Adicionalmente no se cuenta con información histórica de los proyectos la cual es requerida para las mediciones del nivel tres del modelo de madurez.

El modelo de madurez define las áreas de proceso a cubrir, las actividades a realizar, las metas genéricas y específicas que deben de ser cubiertas.



Pero el modelo no define la forma de realizar esto, con el objetivo de tener una base metodológica para realizar estas actividades el presente trabajo involucra la metodología de desarrollo MSF para dar sustento para realizar dichas actividades.

El uso de metodologías tradicionales, predictivas, orientadas a procesos influyen en la capacidad de la empresa enfocándose en la definición de todas las actividades lo que facilita la integración de personal o la supervisión de actividades. MSF es la metodología de desarrollo seleccionada a seguir ya que es aquella que se asemeja más al modelo actual utilizado dentro de FINASIST, es una metodología libre de lenguajes de desarrollo, orientada a procesos, adicionalmente MSF cuenta con una versión ágil que puede ser integrada en proyectos que lo permitan.

La implementación de las áreas de proceso del nivel dos del modelo de madurez propiciará la generación de software de calidad y con esto incrementará la rentabilidad de la organización.

La definición e institucionalización de los procesos y la implementación de prácticas administrativas, permitirán repetir el éxito de los proyectos, así FINASIST se preparará para futuros retos.

El siguiente paso consistirá en la planeación de la implementación de las áreas de proceso dentro de la organización. La tipología de la planeación es del tipo interactivista debido a que se desea ejecutar hoy los métodos para formar un futuro deseable, en el que el software sea libre de errores, que cumpla las expectativas del cliente, en costo y tiempo. Se deberán de ejecutar una planeación interactiva debido a sus principios operativos, de participación y de continuidad.



Conclusiones.

En la sección 1.1, se define que la administración es la acción de planear, proveer recursos y coordinar esfuerzos; Proyecto es un conjunto de esfuerzos con un objetivo específico y software es el elemento que permite a una computadora funcionar de lo cual se define que la administración de proyectos de software es la acción de planear recursos y esfuerzos con el objetivo de generar el elemento que permite a una computadora funcionar. En la página 12 se describen los principales problemas presentados durante el proceso de generación de programas de computadora dentro de los que encontramos que los proyectos no se terminan en plazo, no se ajustan al presupuesto, no se cumple con las especificaciones, entre otros. La solución es la implementación de prácticas administrativas, las definidas en el nivel dos del modelo de madurez de capacidad integrada, el cual es un modelo de mejora de procesos, que contiene las mejores prácticas dirigidas al desarrollo de software, con un enfoque de sistemas lo que tiene un impacto en toda la organización. La premisa en la administración de procesos es que la calidad de un sistema o producto es influenciada por la calidad del proceso usado para desarrollarlo y mantenerlo. En la sección 1.4 se definen los niveles de modelo de madurez y el nivel dos que corresponde a un proceso administrado, así como las áreas de proceso, las cuales establecen el que se debe de hacer y el cómo se responde a través de la metodología seleccionada en la sección 1.5 que es la de Microsoft. Todo esto con el objetivo de que al administrar proyectos de desarrollo de software siguiendo las áreas de proceso del modelo de madurez la empresa FINASIST genere software de calidad que busque la satisfacción total de sus clientes.



Capítulo 2.

Planificación estratégica de las áreas de proceso.

2.1 Planeación estratégica.

De acuerdo a lo definido en el capítulo anterior la problemática presentada, consiste en el desarrollo de software de baja calidad, debido a los errores durante el proceso de desarrollo, los cuales ya se definieron, causado principalmente por la falta de administración de los proyectos de software; las estrategias definidas, para solucionar el problema son: Definición de los procesos de desarrollo, implementación de prácticas administrativas y la recopilación de información que apoye la toma de decisiones, en caso de no aplicar un plan de acción la organización no sería capaz de enfrentar la incorporación de nuevos clientes, la relación con los clientes actuales se podría llegar a deteriorar al punto de la ruptura, entre otras cosas.

En el presente capítulo se desarrolla una guía para la implementación de las mejores prácticas administrativas, establecidas en el nivel dos del modelo de madurez, para la administración de proyectos de software, con el objetivo de mejorar la calidad del desarrollo de software.

Antes de iniciar el desarrollo de las estrategias, se debe considerar que el esfuerzo que representa el presente trabajo se encuentra orientado a establecer las bases de un futuro en el cual, el software generado satisfaga las necesidades de los clientes, se entregue en tiempo, libre de errores y que sea fácil de mantener, permite a las organizaciones responder efectivamente en los retos futuros. Por lo que el objetivo del presente trabajo corresponde:



Analizar e implementar las áreas de proceso del nivel dos del modelo de madurez para la generación de software de calidad bajo la metodología de desarrollo de Microsoft. Al contar con un proceso definido e institucionalizado, se acortará la distancia entre lo que se planea y lo que se ejecuta, al determinar de forma eficiente las etapas y actividades a realizar. La definición del proceso es una actividad que debe ser realizada en conjunto por todas las áreas involucradas en el ciclo de vida del proyecto, mediante la participación de todos en conjunto, se logra una mayor motivación y compromiso de todos los involucrados, lo que busca evitar el rechazo al proceso.

La mayor parte de las actividades están dirigidas a los administradores, son funciones que deben ejecutar, durante el proceso de desarrollo, aplicándolo sobre analistas, desarrolladores, probadores y las actividades desempeñadas por estos, se deben de cuidar las relaciones entre las áreas involucradas. Los principales riesgos que se pueden presentar durante la implementación y ejecución de las estrategias definidas son: rechazo al proceso, falta de apego al mismo, percepción de burocratización por la documentación que se requiera, falta de integración, no contar con el respaldo de la dirección general, oposición al cambio de cultura, esfuerzos aislados no integrados, medio de almacenamiento para generar historia y consulta eficiente para la información recolectada.

La meta para el nivel dos es:

El proceso es institucionalizado como un proceso administrado. Es un proceso en ejecución, que se ha planificado y ejecutado de acuerdo a una política. Utiliza gente capacitada que tiene los recursos adecuados para producir salidas controladas, afecta a involucrados relevantes, es monitoreado, controlado y revisado, y es evaluado en términos de su adherencia a sus descripción de proceso.



A continuación se establecen los programas, subprogramas y actividades a ser ejecutadas.

Programas					
Organización del equipo de proyecto	Administración de requerimientos	Planificación de proyectos	Monitoreo y Control	Medición y análisis	Administración de la configuración
Subprogramas					
Definición de roles	Solicitud de requerimientos	Administración de riesgos	Informe de control externo	Identificación de métricas	Aprobación de cambios
Definición de actividades	Administración de requerimientos	Plan de comunicaciones	Informe de control interno	Plan de medición	Línea base
		Plan de desarrollo	Monitoreo y control de proyectos		
		Plan de prueba			
Actividades					
Asignación de recursos	Desarrollar requerimientos	Estimar alcance	Seguimiento a parámetros de planeación	Establecer objetivos de medición	Identificar elementos de configuración
	Obtener compromisos	Estimaciones	Seguimiento a compromisos	Especificar mediciones	Crear línea base
	Administrar cambios	Identificar riesgo	Revisión de estado	Procedimiento de recolección y almacenamiento de datos	Auditorías de configuración
	Reastreabilidad	Planear recursos	Acciones correctivas	Análisis de datos	
	Identificar inconsistencias	Necesidades y habilidades			
		Plan de proyecto			

Figura 5. Programas, subprogramas y actividades, creación propia.

A continuación se describen cada uno de los programas:

Organización del equipo de proyecto.

La finalidad es el dividir el trabajo del proyecto en diferentes roles y asegurarse que las metas del proyecto se cumplan mediante la vinculación de cada rol del equipo con una responsabilidad principal dentro del proyecto.



También explica que cada rol y cada miembro del equipo tienen la responsabilidad y la igualdad de participación en el proyecto. La visión y el esfuerzo es compartido entre todo el equipo.

Administración de requerimientos.

Administrar los requerimientos de los productos del proyecto y componentes del producto e identificar inconsistencias entre esos requerimientos y los planes del proyecto y producto de trabajo.

Planificación de proyectos.

Establecer y mantener planes que definan las actividades del proyecto.

Monitoreo y control.

Proporciona un entendimiento del progreso del proyecto de tal forma que se pueda tomar acciones correctivas cuando la ejecución del proyecto se desvíe significativamente del plan.

Medición y análisis

Desarrollar y mantener una capacidad de mediciones utilizada para soportar las necesidades de información en los diferentes niveles de la organización.

Administración de la configuración.

Establecer y mantener la integridad de los productos de trabajo mediante la identificación, control, estado y auditorías de configuración.

Los tiempos de ejecución de los programas antes mencionados son en un periodo no mayor a un año lo que implica que son a corto plazo.



2.2 Áreas de proceso.

La guía incluye la definición, establecimiento e implementación del proceso de desarrollo con base al modelo CMMI, sustentado por la metodología MSF.

Con un plan de acción que mantenga la esencia de capturar lo explícitamente necesario y no llenar el proceso con texto que nunca se leerá. Mantenerlo simple facilita que se pueda seguir, el proceso debe ser comprensible de manera fácil, de alto nivel para tareas conocidas, comunes, y detallado para tareas críticas y de alto riesgo. Iniciar y mejorar en ciclos subsiguientes, no paralizarse. Si el proceso facilita las cosas, será más fácil adoptarlo, los líderes de proyecto o directores deben soportar el uso del proceso, deben de informar a la gente el estatus en la adopción de las nuevas prácticas y mantener la mejora de forma continua, a partir de las lecciones aprendidas y de los casos de éxito. No existe una sola estructura o proceso que se aplique a todas las necesidades y entornos. Sin embargo, es posible crear una guía base sobre los principios fundamentales, en lugar de las medidas definitivas y, por lo tanto, hacer los procesos aplicables a una amplia gama de situaciones. MSF tiene ese enfoque. Con la intención de orientar a las organizaciones en organizar a las personas y proyectos para planificar, construir y desplegar con éxito soluciones.

Esta implementación de CMMI, se centra en las prácticas y actividades relacionadas a las áreas de proceso. Se concentra en el modelo de objetivos genéricos, el ambiente de la organización y el concepto de visión compartida. Se recomienda al lector contar con conocimientos básicos del lenguaje unificado de modelado (UML), incluyendo casos de uso.

A continuación se describen una a una las áreas de proceso o programas, así como los subprogramas y actividades de cada una de las áreas.



2.2.1 Organización del equipo de proyecto.

La primer actividad a realizar corresponde al establecimiento de los roles y actividades del equipo de proyecto. MSF especifica seis diferentes roles, se muestran en la siguiente figura.



Figura 6. "Modelo de equipo MSF"¹.

El caso de estudio del presente trabajo la organización FINASIST cuenta con el siguiente organigrama.

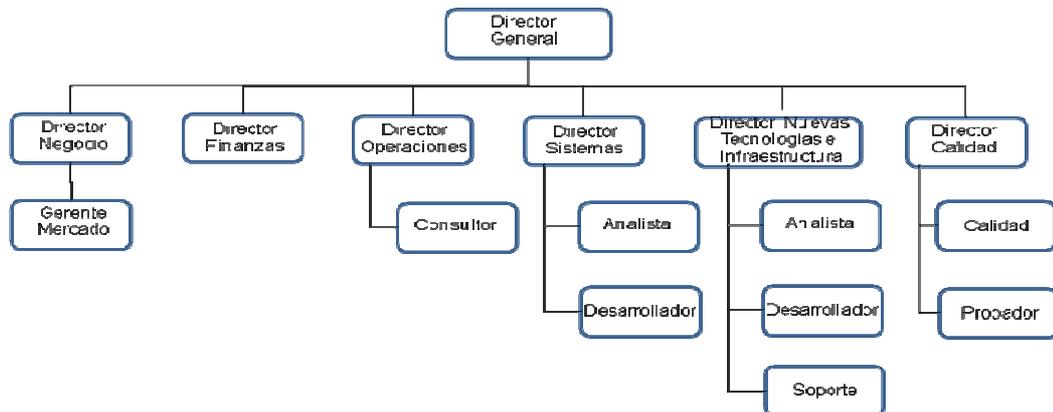


Figura 7. Organigrama FINASIST, creación propia.

¹ Analyzing Requirements and Defining Microsoft .NET Solution Architectures Course 2710 MSDN Training, pág. 14



De acuerdo a los roles definidos en MSF se puede encontrar una relación con la áreas de FINASIST, uno o más roles definidos en MSF pueden ser desarrollados por diferentes áreas de la organización.

MSF	FINASIST
Administrador de producto	Director Negocio, Director Operaciones, Consultor de negocio.
Administrador de Programa	Director Sistemas, Director Nuevas Tecnologías e infraestructura.
Desarrollador	Analista, Desarrollador.
Pruebas	Ingeniero de pruebas.
Administrador de liberaciones	Administrador de la configuración.
Usuario Experimentado	Director Negocio, Director Operaciones, Consultor.

Roles

Responsabilidades

Administrador de producto	Comunicaciones y expectativas del cliente. Durante la fase de diseño reúne los requisitos del cliente y se asegura que se satisfagan las necesidades del negocio, también trabaja sobre los planes de comunicación del proyecto, reuniones de información a los clientes, comercialización a los usuarios, demostraciones y lanzamientos del producto.
Administrador de programa	El proceso de desarrollo y la entrega de la solución al cliente dentro de los lineamientos del proyecto.
Desarrollador	Desarrollo de la solución de tecnología de acuerdo a las especificaciones provistas por el rol de administrador de programa.



Probador	Responsable de identificar y direccionar todos los problemas de la calidad del producto y aprobar la solución para su liberación. Este rol evalúa y valida diseños funcionales y la consistencia con la visión y alcance del proyecto.
Administrador de liberaciones	Despliegue de la solución, valida las implicaciones de infraestructura de la solución para asegurarse que puede ser desarrollada y soportada.
Usuario experimentado	Analiza las necesidades de rendimiento y problemas de soporte de los usuarios y considera las implicaciones del producto y la satisfacción de estas.

Los miembros adicionales al equipo de proyecto incluyen:

Patrocinador del proyecto. Son aquellos que inician y aprueban el proyecto y su resultado.

Cliente. Personas que esperan obtener mejores negocios gracias a la solución.

Usuario final. Individuo o sistema que interactúa directamente con la solución.

Operadores. Organización responsable de la operación de la solución después de la entrega.

La importancia de definir los roles: Permite identificar claramente la posición de cada uno de los integrantes dentro la organización, define las actividades a desempeñar y como se relacionan con el trabajo de otras personas, promueve el compromiso con la empresa.



2.2.2 Administración de requerimientos.

El objetivo es administrar los requerimientos y componentes e identificar inconsistencias entre esos requerimientos y los planes del proyecto y producto de trabajo.

Meta específica: Administrar requerimientos.

Actividades específicas:

- Desarrollar y entender los requerimientos.
- Obtener el compromiso para los requerimientos.
- Administrar cambios a los requerimientos.
- Rastreabilidad de los requerimientos.
- Identificar inconsistencia entre las actividades del proyecto y los requerimientos.

A continuación se presenta una propuesta para la administración de requerimientos en la cual se visualizan dos etapas: solicitud y requerimiento. Se establecen los lineamientos para la generación de documentos para las actividades antes mencionadas y el análisis, así como una serie de componentes necesarios durante el proceso. La etapa de solicitud inicia cuando el cliente solicita un desarrollo, el objetivo consiste en entender claramente la necesidad expresada y elaborar una propuesta que establezca el plazo y costo del proyecto, esta etapa debe ser lo más ágil posible. Una vez que el cliente aprueba la solicitud, esta se convierte en un requerimiento y se inicia el ciclo de vida del proyecto, existe la posibilidad de identificar que el alcance es mayor durante el análisis, lo cual afecta de forma directa los costos y tiempos, por lo que se debe de negociar nuevamente la propuesta con el cliente.



Se inicia con la solicitud de un nuevo proyecto.

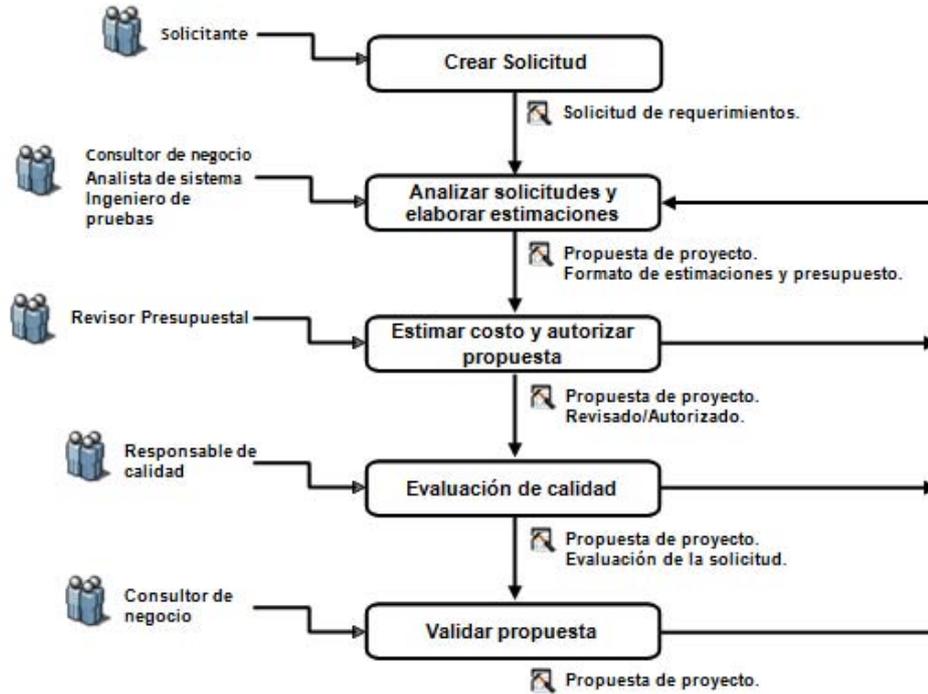


Figura 8. Proceso de creación de solicitudes de nuevos proyectos, creación propia.

Creación de solicitudes de nuevos proyectos.

Rol	Descripción
Solicitante	<ul style="list-style-type: none"> Levanta una solicitud de requerimientos de nuevo proyecto. <p>Nota: Puede ser el cliente, director de negocio, operaciones o consultor, la persona que identifique alguna necesidad.</p>



Elaboración de especificación de requerimiento de software.

Rol	Descripción
Consultor	<ul style="list-style-type: none"> ▪ Crea la especificación de requerimiento de software, excepto el modelo de casos de uso, es decir el detalle de estos.

Análisis de solicitudes y elaboración de estimaciones.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> ▪ Realizará el análisis de los requerimientos con ayuda del analista de sistemas. ▪ Basado en el resultado del análisis y en las estimaciones, formula la propuesta económica. ▪ Para aquellas solicitudes que resulten observadas técnicamente, corregirá todas las observaciones y enviará nuevamente la solicitud a revisión. ▪ Deberá generar un documento de lecciones aprendidas.
Analista de sistemas	<ul style="list-style-type: none"> • Apoyará al consultor de negocio en la realización de la propuesta en los aspectos técnicos esenciales y elaborará un documento con las estimaciones además podrá realizar observaciones a la propuesta.
Ingeniero de pruebas	<ul style="list-style-type: none"> ▪ Apoyará al Analista de Sistemas, en la elaboración de las estimaciones de tiempos para la ejecución de las pruebas al producto, según el alcance de la misma.



Estimar costo y autorizar propuesta.

Rol	Descripción
Revisor presupuestal	<ul style="list-style-type: none"> ▪ Realizará un análisis económico de la propuesta con base a las estimaciones realizadas y el alcance del proyecto. Se puede incluir los costos de construcción del proyecto. ▪ Podrá aprobar o proveer observaciones, e integrará dicho presupuesto a la propuesta que se enviará al cliente. ▪ Una vez finalizada la revisión de costos, enviará la solicitud al responsable de calidad para completar las revisiones. <p>NOTA: Esta actividad puede ser desempeñada por el consultor de negocio depende de la confidencialidad de las cuestiones económicas de la organización.</p>

Evaluación de calidad.

Rol	Descripción
Responsable de calidad	<ul style="list-style-type: none"> ▪ Realizará una evaluación de la propuesta. ▪ Decidirá la aprobación u observación de la misma y se la devolverá al consultor de negocio.

Validar propuesta.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> ▪ Validará que la propuesta cuente con la aprobación de calidad y se debe de presentar al cliente, en caso de ser necesario, convocar al analista de sistemas para que brinde soporte en los aspectos técnicos del mismo. ▪ Debe procesar y documentar la respuesta del cliente, puede ser la aprobación, el rechazo o la observación de la propuesta. ▪ Para aquellas propuestas a las que el cliente les realice observaciones, generará un informe detallado de los puntos observados por el cliente, los integrará a la propuesta, generará una segunda versión e informará los cambios a los involucrados en el requerimiento.



	<ul style="list-style-type: none">▪ Para las propuestas que resulten aprobadas por el cliente, solicitará dicho documento firmado. <p>NOTA: el resultado de toda la comunicación con el cliente deberá registrarse en minutas, que indiquen las discusiones sobre la funcionalidad abarcada por el producto, en función de las limitaciones establecidas por el cliente y demás negociaciones realizadas.</p>
--	---

En cada etapa, se debe de generar un documento, donde se definan los elementos obtenidos, el presente trabajo deja al lector la definición de dichos documentos, por lo que solo se recomienda la creación de estos.

Un elemento neurálgico durante el proceso de desarrollo es la creación de un documento, que sirva de eje durante todo el ciclo de vida del proyecto, el cual se completará en cada una de las etapas.

Este documento es el de Especificación de Requerimiento de Software ERS, se iniciara la creación durante esta etapa. Debe de contener los siguientes elementos.

- Objetivo.
 - Debe indicar de manera general que se pretende lograr con el desarrollo del sistema.
- Alcance.
 - Indicar en términos generales las funciones que el sistema debe realizar, son los requerimientos expresados por el cliente, deben estar numerados y priorizados para poder tener rastreabilidad con los casos de uso que se deriven.



- Lo que no se contempla.
 - Indica aspectos funcionales o no funcionales que se desea destacar, no estarán incluidos en el producto
- Restricciones y supuestos.
 - Cualquier aspecto que deba ser considerado para el desarrollo, que pueda afectar el cumplimiento del requerimiento, que viene desde el ambiente del negocio, o acordado con anterioridad, o cuestiones legales.
- Listado de la funcionalidad del sistema.
 - Descripción general de la funcionalidad del sistema, identifican los casos de los requerimientos. Se establecen prioridades entre ellos, su complejidad en función de su dificultad para el desarrollo y se determinan en función de dos parámetros básicos, interfaces complicadas o cálculos complicados o combinación de ambos.

Se muestra a continuación un formato de presentación sugerido:

N° Req. Cliente	N° Caso de Uso	Nombre del Caso de Uso	Prioridad	Complejidad

- Diagrama de casos de uso.
 - Se debe incluir el diagrama que muestre de manera gráfica los alcances funcionales del producto.
- Listado de actores.
 - Los actores mencionados en el diagrama de casos de uso son indicados aquí. Los actores son personas, roles o sistemas que interactúan con la funcionalidad solicitada. Se debe de indicar el tipo del actor si es concreto o abstracto, una categoría o estereotipo, si se trata de un actor visual, un software, o un hardware.



- **Perspectiva del producto.**
 - La relación con otros productos. Si el producto es independiente y totalmente auto contenido, si es componente de un sistema más grande, la relación con los componentes del sistema más grande, identificándose las interfaces de comunicación.

A través de los casos de uso identificados se realizarán las estimaciones iniciales, de desarrollo y de pruebas del producto.

Se mide el esfuerzo, es decir las horas que llevará la construcción, sin establecer fechas de inicio o fin. Adicionalmente se realiza la estimación de los siguientes elementos, se definen como un porcentaje sobre el tiempo de desarrollo, estos datos deben de almacenarse, con el objetivo de ajustarse al paso del tiempo.

Actividad	Porcentaje sobre desarrollo
Análisis y diseño.	40%
Planificación.	5%
Prueba.	30%
Despliegue.	10%
Administración de la configuración.	5%
Aseguramiento de la calidad.	Tiempo fijo, aplicación de cuestionarios de calidad se estima 4 Hrs.
Monitoreo y control de proyectos.	5%
Medición y Análisis.	10%

Después de contar con los elementos antes expuestos se realiza la propuesta de esfuerzo al cliente, cabe mencionar que esta es la primera estimación de esfuerzo, esta debe de realizarse en un periodo de tiempo relativamente corto.



Esta estimación es creada con el objetivo de ofrecerle al cliente una idea del costo del proyecto, para que el pueda determinar la viabilidad del mismo, hace notar que esta estimación puede variar, al realizar un estudio más detallado del proyecto.

La propuesta entregada al cliente debe contener al menos los siguientes elementos

- Un resumen ejecutivo.
 - Debe de contener el objetivo del requerimiento y una breve descripción y hacer referencia a la especificación de requerimiento de software.
- Plazo de ejecución
 - Se debe de establecer el esfuerzo estimado para el desarrollo del requerimiento, el total de horas o días, con un resumen de actividades y tiempos.
- Plan de entregas.
 - Depende de la complejidad o duración de requerimiento este se puede hacer en una o varias entregas. Se debe definir el número de estas y los entregables en cada una de ellas.
- Oferta económica.
 - Inversión del requerimiento, establece la moneda, la validez de la oferta y la forma de pago.
- Responsabilidades.
 - Se deben de definir las responsabilidades tanto del cliente como del proveedor.

Una vez que la propuesta fue evaluada por el cliente y aceptada se inicia el proceso de requerimientos.



A continuación se describe el proceso de requerimientos:

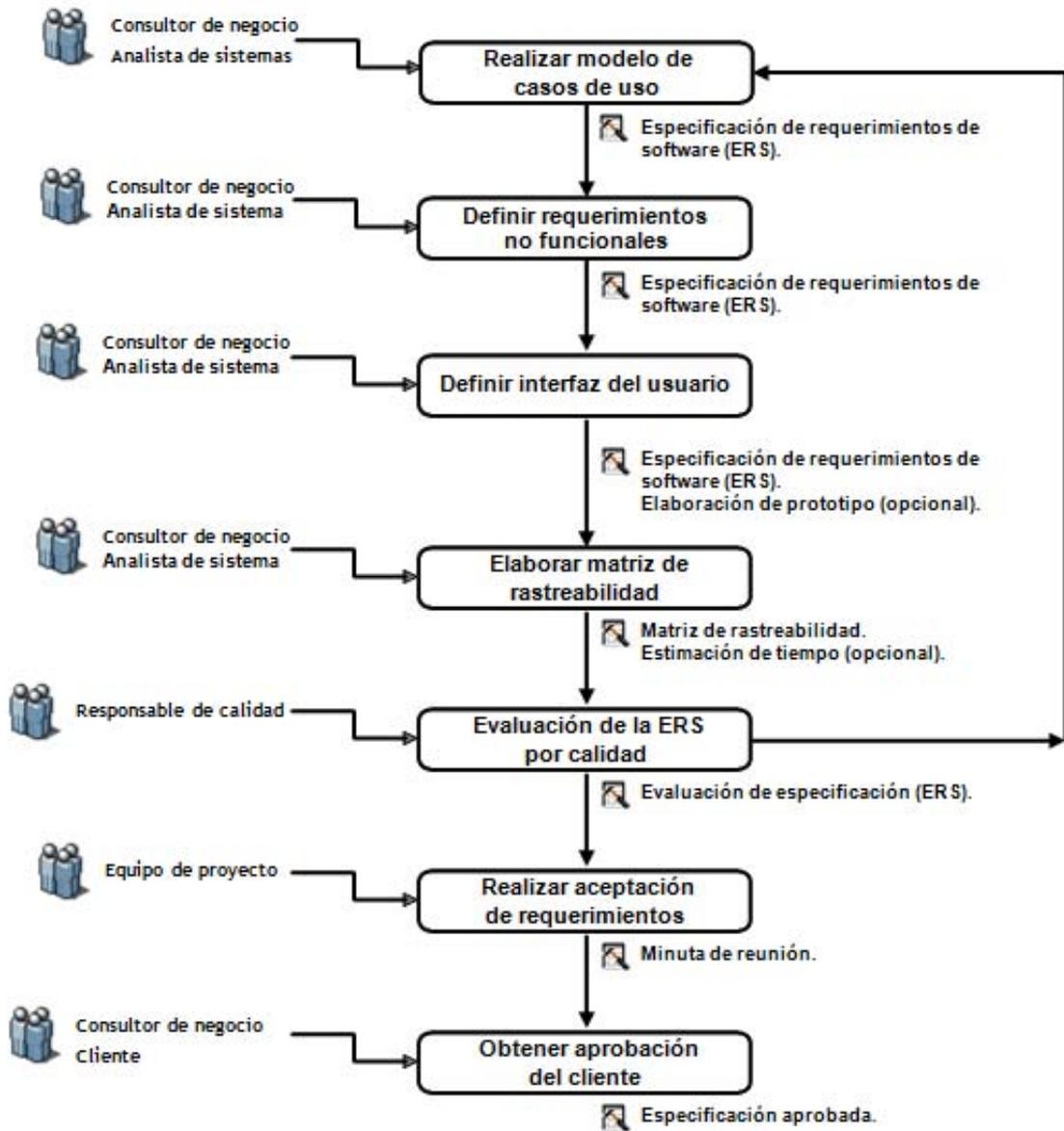


Figura 9. Proceso de administración de requerimientos, creación propia.



Realizar modelo de casos de uso.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> ▪ Será el responsable de elaborar la ERS. ▪ Realizará el modelo detallado de casos de uso. Durante la solicitud de requerimiento solo se nombraron los casos de uso, los actores involucrados. En esta etapa se detalla cada caso de uso, desde el punto de vista de negocio, define flujos, funcionalidad y validaciones a mayor detalle. ▪ Solicitará una junta con el analista de sistemas para presentar el modelo detallado de casos de uso para el entendimiento correcto del requerimiento.
Analista de sistemas	<ul style="list-style-type: none"> ▪ Validará el modelo de casos de uso, identificará la funcionalidad completa que se requiere, el flujo de la información y restricciones. Debe comprender completamente el requerimiento para poder dirigir el desarrollo del mismo. Verificará la viabilidad del requerimiento desde el punto de vista tecnológico. ▪ Identificará inconsistencias dentro de los requerimientos.

Definir requerimientos no funcionales.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> ▪ Establecerá los requerimientos adicionales pueden ser: diseño del requerimiento, restricciones de tiempo de ejecución del requerimiento, todo aquello que deba realizarse pero que no sea funcionalidad del requerimiento.
Analista de sistemas	<ul style="list-style-type: none"> ▪ Debe validar la viabilidad y factibilidad de los requerimientos no funcionales, y completar en caso de que se requiera.

Definir interfaz de usuario.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> ▪ Establecerá una idea de la interfaz deseada para el usuario, en caso de que aplique.



Analista de sistemas	<ul style="list-style-type: none"> Elaborará el prototipo de la interfaz solicitada por el consultor de negocio, en caso de aplicar.
----------------------	---

Elaborar matriz de rastreabilidad.

Rol	Descripción
Consultor de negocio	<ul style="list-style-type: none"> Identificará los impactos a la(s) aplicación(es) desde el punto de vista negocio.
Analista de sistemas	<ul style="list-style-type: none"> Identificará los impactos a la(s) aplicación(es) desde el punto de vista técnico.

Evaluación de la ERS por calidad.

Rol	Descripción
Responsable de calidad	<ul style="list-style-type: none"> Deberá realizar la revisión de la Especificación de Requerimientos de Software.

Realizar aceptación de requerimientos.

Rol	Descripción
Equipo de proyecto	<ul style="list-style-type: none"> Deberá manifestar su consenso respecto a la viabilidad de implementación de los requerimientos especificados en la Especificación de Requerimientos de Software por medio de una Minuta. Deberá de evaluar que los tiempos definidos en la solicitud son los correctos, o definir los nuevos tiempos, informará a los consultores y al cliente las desviaciones en caso de existir, establece la justificación de las modificaciones. <p>NOTA: Es en esta etapa en donde se puede realizar la última actualización de los tiempos del proyecto.</p>



Obtener aprobación del cliente.

Rol	Descripción
Consultor de Negocio	<ul style="list-style-type: none"> ▪ Validará con el cliente la Especificación de Requerimientos de Software para asegurar que esté correcta. ▪ Validación los prototipos y los casos de uso. ▪ Si la especificación no es aceptada, se actualiza hasta su aprobación por el cliente. Si en estas negociaciones se solicita un cambio que afecta al alcance de la propuesta, debe evaluarse. ▪ Las negociaciones y aprobación deben quedar registradas en minutas.

Durante este proceso se identificaron los requerimientos, se estableció una posible solución y se obtuvo la aceptación del cliente con la misma, esta etapa es de vital importancia para el proceso de desarrollo, ya que establece las bases y el sustento de todo el proyecto. Todo el tiempo dedicado a este proceso facilitará el desarrollo de los procesos posteriores.

2.2.3 Planificación de proyectos.

El objetivo es establecer y mantener planes que definen las actividades del proyecto.

Actividades específicas.

- Definir el ciclo de vida del proyecto.
- Identificar los riesgos del proyecto.
- Planear el manejo de la información.
- Planear los recursos del proyecto.
- Planear las necesidades de conocimientos, y habilidades.



- Establecer el plan de proyecto.
- Obtener compromisos del plan.

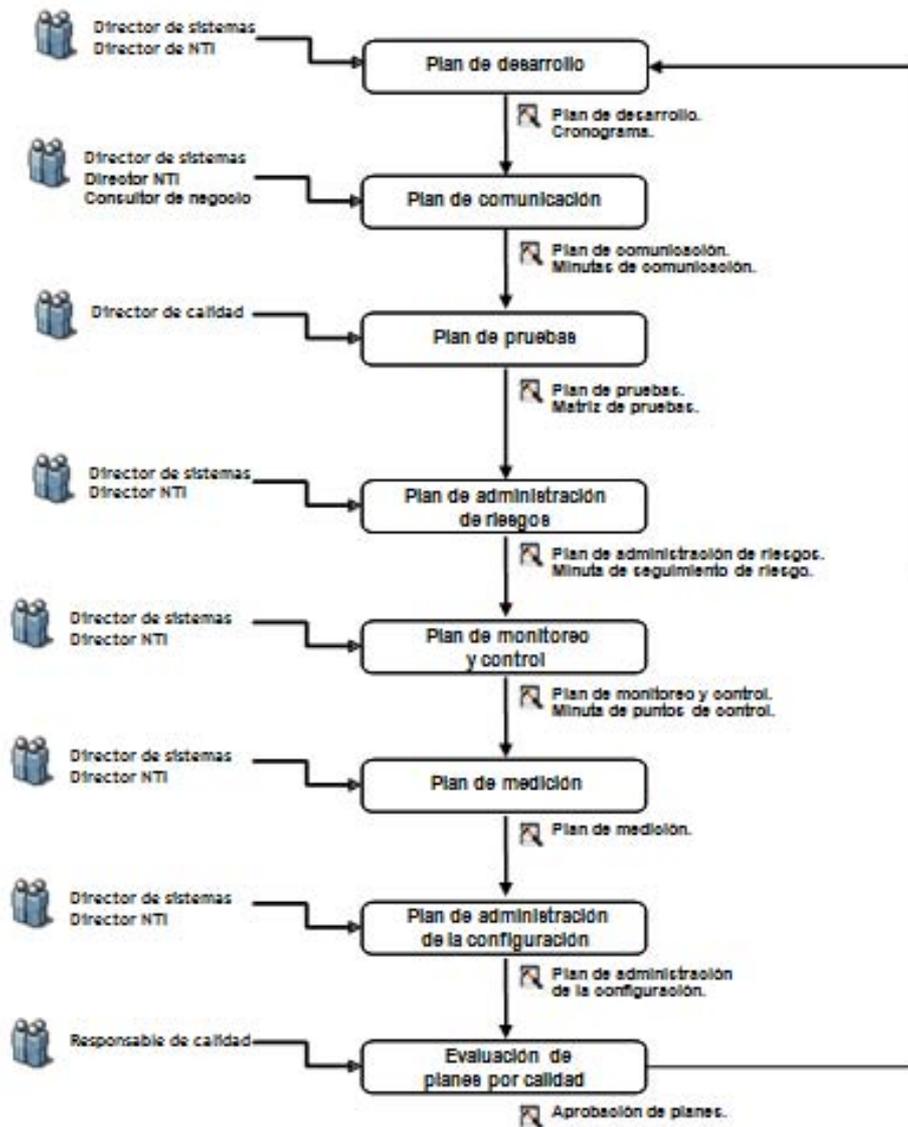


Figura 10. Proceso de planificación de proyectos, creación propia.



Plan de desarrollo.

Rol	Descripción
<p>Director de sistemas</p> <p>Director NTI</p>	<ul style="list-style-type: none"> ▪ Obtendrá las bases para la elaboración del plan de desarrollo de software, las cuales serán, la ERS y las solicitudes de requerimientos aprobadas. La ERS puede no estar completa al momento de la elaboración del plan de desarrollo. Definirá el ciclo de vida del proyecto. ▪ Identificará los recursos humanos y materiales necesarios para llevar a cabo la ejecución de las actividades del proyecto. ▪ Si es necesario elaborar planes de iteración, de acuerdo al ciclo de vida elegido. ▪ Las tareas que deben realizarse, serán la base para la derivación del cronograma. Realizará la derivación de la programación del proyecto (cronograma). ▪ Deberá tener en cuenta las siguientes consideraciones para realizar esa derivación: <ul style="list-style-type: none"> ○ Cantidad de personas con las que cuenta en el proyecto y perfiles de cada uno. ○ Disponibilidad de las personas en función del momento en que podrán integrarse al proyecto y la asignación diaria que tendrá para el mismo. ○ Las dependencias de las actividades que se deben realizar. ○ El índice de solapamiento factible para las actividades. ○ Las características de las actividades a realizar, para asignar aquellas que son similares u homogéneas o estén muy relacionadas entre sí, al mismo responsable. ○ Las características del producto a construir o los entregables definidos para ese producto ○ Los riesgos, si alguno puede influir en las actividades de calendarización del proyecto. ▪ Asignará a las personas que estarán involucradas en el proyecto, la asignación de las personas debe realizarse mediante una reunión en la cual se analizan la disponibilidad de recursos, la decisión final deberá quedar documentada. Realizará la asignación de trabajo a los miembros del equipo de proyecto.



	<ul style="list-style-type: none"> ▪ En los casos que no sea posible determinar quién será el responsable de ejecutar alguna tarea ya definida designará un recurso responsable y al momento de su ejecución, determinará la afectación de los recursos. ▪ Especificará las necesidades de capacitación. ▪ Deberá detectar las necesidades de capacitación del equipo del proyecto. ▪ Para cada necesidad, se indicará la fecha límite para tomar la capacitación (la fecha indica cuándo debe finalizar la capacitación) y los miembros del equipo candidatos a tomar el curso. ▪ Obtendrá el compromiso por parte del equipo de proyecto y demás involucrados. ▪ Administrará el proyecto de acuerdo con el proceso de desarrollo de software. ▪ Si existen aspectos de implementación tareas del plan de desarrollo de software o planes asociados que no puedan ser definidos al momento de su elaboración, estos quedarán pendientes y conforme avance el proyecto y puedan ser definidos, se modificará el plan y se registrará el historial de cambios. ▪ En todo momento se puede apoyar en algún activo histórico realizado por proyectos similares.
--	---

Plan de comunicación.

Rol	Descripción
Director de sistemas Director NTI	<ul style="list-style-type: none"> ▪ Establecerá las juntas de inicio de proyecto. ▪ Calendarizará las reuniones periódicas durante el proyecto, con el equipo y con la dirección general. ▪ Definirá las fechas para los reportes internos de estado del proyecto, el formato y la forma en que serán distribuidos dentro de la organización.
Consultor de negocio	<ul style="list-style-type: none"> ▪ Definirá las fechas para los reportes externos de estado del proyecto, el formato y la forma en que serán distribuidos al cliente.



Plan de pruebas.

Rol	Descripción
Director de Calidad	<ul style="list-style-type: none"> ▪ Deberá crear el plan de prueba del proyecto, se puede apoyar algún activo histórico realizado por proyectos similares. ▪ Para la programación del plan de prueba del proyecto deberán tenerse en cuenta las estimaciones, la cantidad de ciclos de prueba estimados, reportes de desviaciones, de tiempos muertos y restricciones identificadas en la estimación anterior. ▪ Deberá identificar los casos de uso que están contemplados para el proyecto, los tipos y casos de prueba que se ejecutarán en el ciclo de pruebas. ▪ Para realizar la programación del ciclo de prueba se tendrá en cuenta el personal disponible para la ejecución de las pruebas. ▪ Elaborará la matriz de pruebas.

Plan de administración de riesgos.

Rol	Descripción
Director de sistemas Director NTI	<ul style="list-style-type: none"> ▪ Administración del riesgo pre activa, continua evaluación de los riesgos y toma de decisiones. ▪ Evaluación continua, monitoreo y administración activa del riesgo. ▪ Identificación de riesgo, potenciales problemas. ▪ Análisis de riesgo, toma de decisiones acerca de las prioridades. ▪ Formulación de estrategias, planes y acciones para controlar los riesgos. ▪ Seguimiento de riesgos, monitoreo del estatus de riesgos específicos y documentación de los planes de acción. ▪ Control de riesgos, ejecución de planes de acción y reporte de estatus. ▪ Aprendizaje de riesgo, formalizar las lecciones aprendidas, los conocimientos y los registros de forma que sean reutilizables por el equipo y por la empresa.



Planes.

Rol	Descripción
Director de sistemas Director NTI	<ul style="list-style-type: none"> ▪ El plan de monitoreo y control será elaborado. ▪ Deberá definir en forma conjunta con el Director de Sistemas y/o Director de Nuevas Tecnologías e Infraestructura las auditorías y revisiones que se realizarán en el proyecto ▪ El plan de medición y análisis será elaborado. ▪ Deberá programar la asignación de control de manera tal que se puedan realizar las mediciones especificadas sobre el avance del proyecto. ▪ El plan de administración de la configuración será elaborado. ▪ Se pondrán bajo Administración de Configuración. ▪ Si el plan de desarrollo de software o alguno de sus planes asociados es rechazado por alguno de los revisores, deberán corregir el mismo y volver a iniciar el circuito de aprobaciones. ▪ Si necesita más precisiones acerca de las razones por las cuales cualquiera de los planes ha sido rechazado deberán ser solicitadas al revisor correspondiente. <p>NOTA: Estos planes se detallan más adelante ya que corresponden a otras áreas de proceso.</p>

Evaluaciones de planes por calidad.

Rol	Descripción
Responsable de calidad	<ul style="list-style-type: none"> ▪ Deberá realizar la revisión de los planes.



2.2.4 Monitoreo y control.

El objetivo es proporcionar un entendimiento del progreso del proyecto de tal forma que se puedan tomar acciones correctivas al encontrar que la ejecución del proyecto se desvíe significativamente del plan. Actividades específicas.

- Seguimiento a los parámetros de planeación del proyecto, compromisos y riesgos.
- Revisiones del estado del proyecto y acciones correctivas.

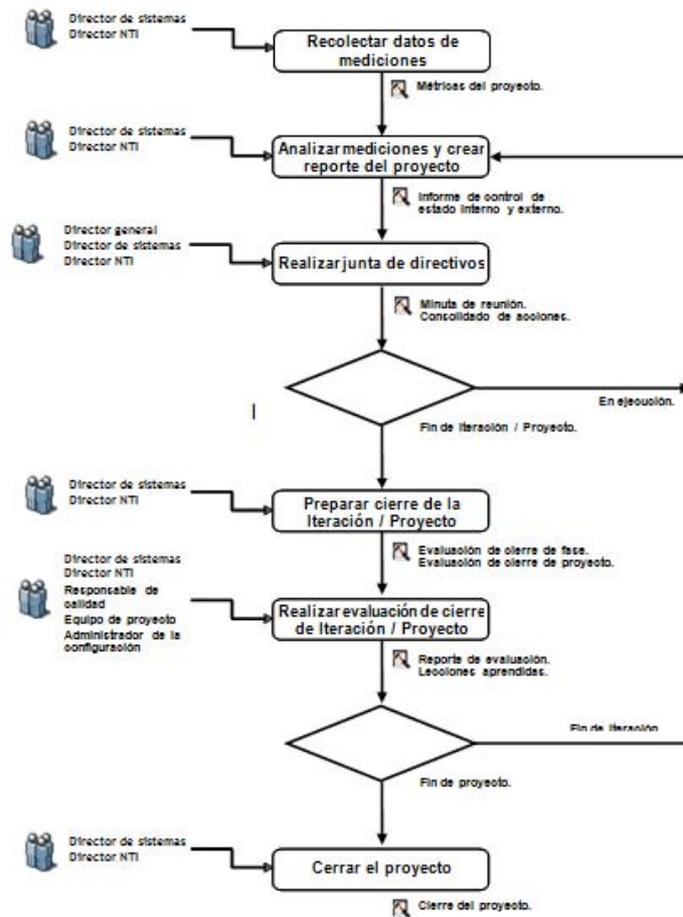


Figura 11. Proceso de monitoreo y control, creación propia.



Puntos de control y acciones a seguir

Rol	Descripción
Equipo de Proyecto	<ul style="list-style-type: none"> • Los miembros del equipo de proyecto reportan el avance de las tareas y registran en caso que sea necesario, lecciones aprendidas.
Director de Sistemas o Director de Nuevas Tecnologías e Infraestructura	<ul style="list-style-type: none"> • Periódicamente revisa los avances reportados y asigna actividades a los miembros del equipo de proyecto e involucrados. • Realizará las siguientes actividades: <ol style="list-style-type: none"> 1. Analizará dada una fecha, una actividad determinada, o la culminación de una fase, el estado de las actividades y productos del proyecto con respecto al plan al cual pertenecen. 2. Revisará y analizará los planes del proyecto y sus objetivos contra la ejecución real del proyecto. Las variaciones deben ser identificadas, analizadas y las acciones correctivas llevadas a cabo. Tomar especial atención en: calendario, recursos, costo y tamaño. 3. Revisará el estado de los riesgos y de los recursos de equipamiento. 4. Si hay un retraso por alguna causa ajena a los miembros del equipo de proyecto o si las causas involucran al equipo de proyecto de alguna manera (mala estimación, mala capacitación del recurso), se deberá informar en el control de estado y se indicará el motivo. 5. Se controlan las estimaciones de tiempo, esfuerzo, tamaño y recursos (humanos y materiales) documentadas en los planes con las mediciones obtenidas. 6. Ante el acontecimiento de un evento calificado como riesgo, se deberán realizar las actividades de mitigación o contingencia. Si surge un nuevo riesgo se deberá analizar y priorizar. 7. Si es necesario implementar una acción correctiva en el plan afectado deberá ser aprobada. 8. Si existiera algún problema durante la ejecución del plan, se identifican las acciones correctivas y se les da seguimiento hasta su cierre.



Actividades de monitoreo y control.

Rol	Descripción
Director de Sistemas o Director de Nuevas Tecnologías e Infraestructura	<ul style="list-style-type: none"> ▪ Llevará adelante las actividades de monitoreo y control del proyecto, donde deberá controlar al menos: <ul style="list-style-type: none"> ○ Los indicadores del proyecto definidos, tamaño, esfuerzo, costo y cumplimiento de calendario. ○ Los riesgos, problemas y dependencias presentados. ○ El análisis y aplicación de las soluciones a problemas encontrados. ○ Los reportes de auditorías y revisiones y los hallazgos de las mismas. ○ Los defectos encontrados por revisiones por pares y en las fases de pruebas. ▪ Preparará el informe de avance del proyecto. ▪ Las reuniones de seguimiento del proyecto se ejecutan periódicamente para determinar las acciones necesarias para que los resultados y el rendimiento del proyecto, se ajusten a las necesidades actuales y futuras de los clientes, del negocio y de los usuarios finales en forma apropiada. ▪ El resultado de las reuniones quedará reflejado en una minuta.

Actividades de fin de iteración.

Rol	Descripción
Director de Sistemas o Director de Nuevas Tecnologías e Infraestructura	<ul style="list-style-type: none"> ▪ Se reúnen el equipo de proyecto y el director de sistemas o director de nuevas tecnologías e infraestructura para realizar la evaluación de fin de fase o iteración. ▪ Si se ha detectado otro problema, deberá implementar las acciones correctivas necesarias y darles seguimiento hasta su cierre. ▪ El resultado de las reuniones quedará reflejado en una minuta. Realizarán la evaluación del fin de iteración. ▪ Revisarán todos los informes de revisiones, monitoreos y evaluaciones realizados al proyecto. ▪ Registrarán lecciones aprendidas.



Actividades de fin de proyecto.

Rol	Descripción
Director de Sistemas o Director de Nuevas Tecnologías e Infraestructura	<ul style="list-style-type: none"> ▪ Riesgos, problemas, cambios, dependencias y errores deben estar propiamente identificados y documentados para su seguimiento y cierre. ▪ Se deben identificar aquellos estándares, procedimientos, buenas prácticas, nuevos o que necesiten mejoras y comunicarlos a la organización para ser considerados en futuros proyectos. ▪ Realizarán la evaluación del fin de proyecto. ▪ Revisarán todos los informes de revisiones, monitoreos y evaluaciones realizados al proyecto. ▪ Se cierra formalmente el proyecto.

2.2.5 Medición y análisis.

El objetivo es desarrollar y mantener una capacidad de mediciones utilizada para soportar las necesidades de información en los diferentes niveles de la organización.

Actividades específicas.

- Establecer y mantener objetivos de medición.
- Especificar mediciones.
- Procedimientos de recolección y almacenamiento de datos.
- Análisis de datos.

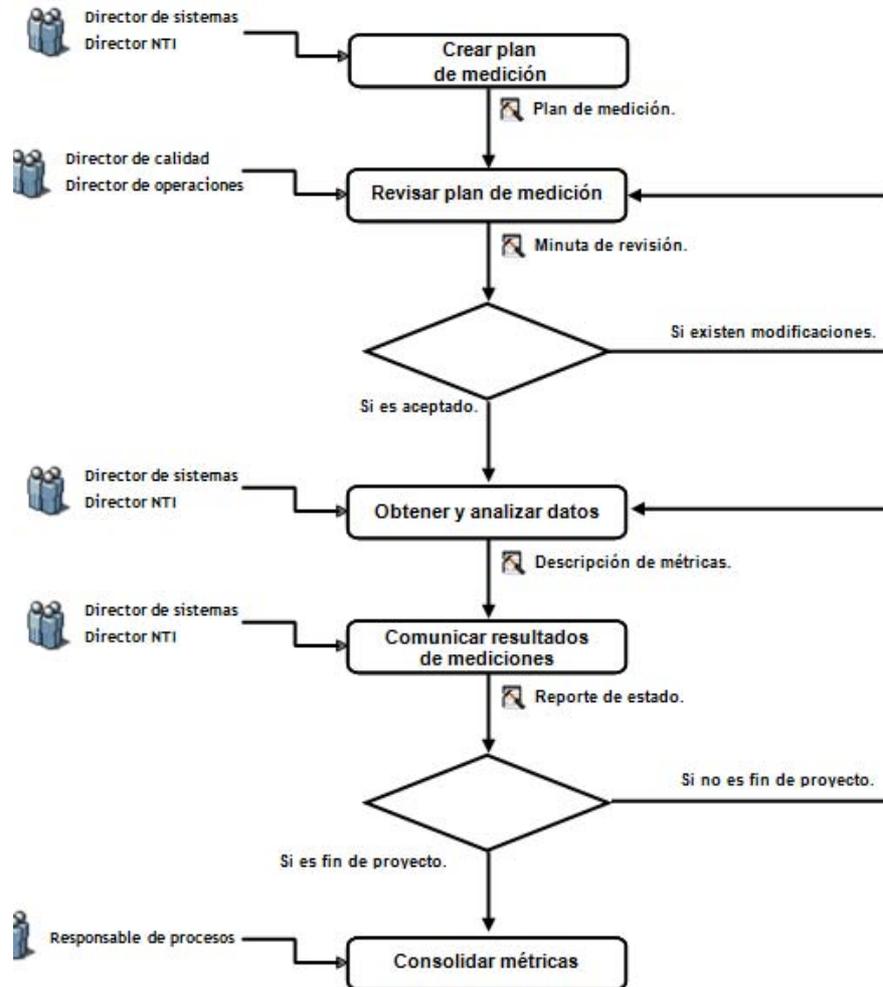


Figura 12. Proceso de medición y análisis, creación propia.

Definición del plan de medición.

Rol	Descripción
Director de Sistemas Director de	<ul style="list-style-type: none"> ▪ Creará el plan de medición para el proyecto. ▪ Tomará las métricas especificadas en el documento de descripción de métricas y en función de las características del proyecto y del producto. Especificadas en el plan de desarrollo de software y el documento de especificación respectivamente. ▪ Verán las métricas que aplican y explicarán, si fuera



Rol	Descripción
NTI	<p>necesario, cómo las tomarán en el contexto del proyecto.</p> <ul style="list-style-type: none"> ▪ Si existen aspectos de implementación de alguna tarea del plan de medición que no puedan ser definidos al momento de su creación, estos quedarán pendientes y conforme avance el proyecto y puedan ser definidos, se modificará el plan de medición, si el mismo ya forma parte de una línea base del proyecto, deberá respetar el procedimiento de administración de cambios. ▪ Colocará el plan de medición una vez creado bajo administración de configuración, lo pondrá como pendiente de revisión e informará al director de operaciones y al director de calidad para que procedan a la revisión y aprobación del mismo. ▪ Será sometido a las auditorías y revisiones planificadas.

Revisión del plan de medición.

Rol	Descripción
Director de Operaciones Director de Calidad	<ul style="list-style-type: none"> ▪ Aprobarán el plan de medición en el momento de la creación y cada vez que el mismo sea modificado. ▪ Si por alguna razón debieran rechazarlo, informará los motivos a través de la minuta de reunión para que éste realice las correcciones necesarias.

Recolección de datos.

Rol	Descripción
Director de Sistemas Director de NTI	<ul style="list-style-type: none"> ▪ Deberá registrar o designar un responsable dentro del equipo del proyecto para registrar en el formato de métricas de proyecto las métricas relacionadas con sus actividades según lo especificado en el plan de medición.



Análisis de datos.

Rol	Descripción
Director de Sistemas	<ul style="list-style-type: none"> ▪ El análisis en función de las métricas obtenidas. Los resultados de este análisis deberán documentarse en una minuta. ▪ Si fuera necesario, deberá implementar acciones correctivas y realizar su seguimiento. ▪ Puede involucrar a otros roles, por ejemplo el responsable de aseguramiento de calidad y/o el administrador de configuración para revisar y analizar la información de las métricas del proyecto.
Director de NTI	

Comunicación de resultados de medición

Rol	Descripción
Director de Sistemas	<ul style="list-style-type: none"> ▪ Deberá comunicar de manera periódica los resultados recolectados a los miembros del equipo y a los involucrados relevantes a quien deba de reportar el avance y estado de las actividades del proyecto. Si el reporte de la información de mediciones se lleva a cabo mediante una reunión, entonces deberá documentarse su resultado en una minuta.
Director de NTI	

Consolidación de métricas.

Rol	Descripción
Responsable de Procesos	<ul style="list-style-type: none"> ▪ Deberá ingresar las métricas obtenidas por todos los proyectos en el formato de métricas de manera periódica para su posterior análisis con la dirección general. ▪ Deberá verificar que los datos sean correctos en términos de fechas y consistencia. ▪ Debe presentar las métricas organizacionales de manera periódica con la dirección general. La periodicidad estará definida de acuerdo a lo especificado en el punto comunicación de resultados de mediciones en el ámbito organizacional.



2.2.6 Administración de la configuración.

El objetivo es establecer y mantener la integridad de los productos de trabajo mediante la identificación, control, estado y auditorías de configuración.

Actividades específicas.

- Identificar los ítems de configuración.
- Crear línea base.
- Seguimiento de requerimientos de cambios.
- Auditorías de configuración.

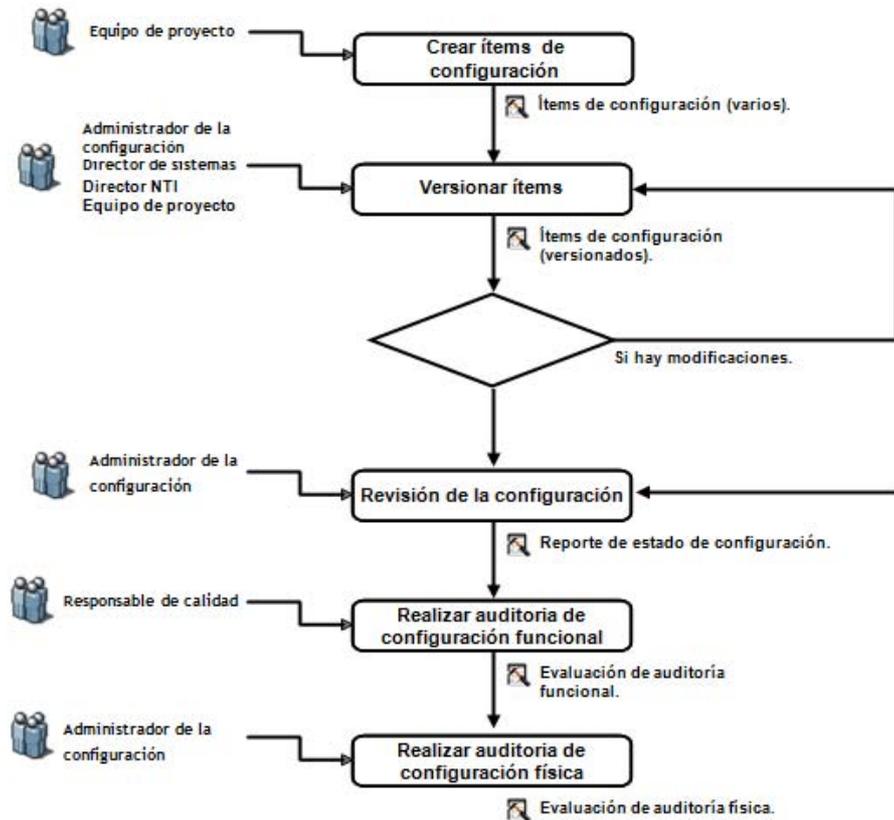


Figura 13. Proceso de administración de la configuración, creación propia.



Creación de ítems de configuración.

Rol	Descripción
Equipo de Proyecto	<ul style="list-style-type: none"> ▪ Todos los archivos físicos creados en el contexto del proyecto de desarrollo de software, contemplados en el plan de administración de configuración serán administrados en distintos niveles según el archivo de que se trate: <ul style="list-style-type: none"> ▪ Se administra el versionarlos. ▪ Se completa su historia de cambios. ▪ Se incluye en la línea base. ▪ Solo se podrán crear ítems de configuración a partir del repositorio habilitado para este proyecto. ▪ Desarrollarán las versiones iniciales de los ítems de configuración y, al concluir el desarrollo, los ingresarán a repositorio especificado, se notificará al resto de los miembros afectados el estado. ▪ El nombrado de los ítems se especifica en el plan de administración de configuración de software.

Nota: Se recomienda el uso de una herramienta de control de versiones como lo es Source Safe para entornos de desarrollo de Microsoft.

Revisión de la configuración.

Rol	Descripción
Administrador de Configuración	<ul style="list-style-type: none"> ▪ Verificará que los objetos seleccionados a modificar se encuentren solicitados o apartados para no ser modificados por dos personas al mismo tiempo.

Auditorías de configuración.

Rol	Descripción
Responsable de QA	<ul style="list-style-type: none"> ▪ Realizará estas auditorías y se utilizará un Checklist. ▪ Estas auditorías deberán ejecutarse dos veces durante la ejecución del proyecto, una antes de la creación de una línea base y otra después de la creación de una subsiguiente línea base.



Rol	Descripción
Administrador de Configuración Global	<ul style="list-style-type: none"> ▪ Que los Ítems de configuración se generen a partir de los repositorios aprobados para el proyecto. ▪ Que el proyecto implemente la estructura de directorios, si aplica. ▪ Estas auditorías deberán ejecutarse dos veces durante la ejecución del proyecto, una antes de la creación de una línea base y otra después de la creación de una subsiguiente línea base. ▪ Documentará el resultado de sus auditorías se utilizará un Checklist de Auditoria de Configuración Física.

2.3 Indicadores.

Un elemento fundamental en la administración de proyectos, corresponde a la definición de las métricas o indicadores, a seguir durante las fases del ciclo de vida, que nos permiten evaluar la mejora, resultado de la implementación de la guía metodológica. Estos indicadores deben de ser definidos con la dirección ya que deben de estar alineados con los objetivos de la empresa.

A continuación se describe una forma de elaborar dichas métricas, mediante la implementación del siguiente formato.

Objetivos	Preguntas	Métricas	Dirigida
Mejorar la productividad del personal.	¿Se entregaron los requerimientos en tiempo?	Fechas reales menos fechas planeadas.	Alta dirección.
	¿Porcentaje de horas por re-trabajo?	Horas asignadas a incidencias, defectos y errores contra horas de requerimiento.	



Objetivos	Preguntas	Métricas	Dirigida
Mejorar la calidad del producto.	¿Cuál es la cantidad de defectos encontrados en fase de pruebas?	Cantidad de defectos por ERS.	Alta dirección.
	¿Cuál es la cantidad de defectos encontrados en fase de pruebas Integración?	Cantidad de defectos reincidentes.	
	¿Cuál es la cantidad de defectos encontrados en Producción cliente?	Cantidad de defectos encontrados en producción	

Se debe de elaborar la siguiente guía para cada una de las métricas definidas, en la cual se describe la unidad de medida, los elementos a utilizar para su medición y la forma de cálculo.

Actividad	Descripción
Concepto Medible.	¿Qué se quiere medir? Descripción de la métrica.
Indicador.	La unidad de lo que se mide.
Métrica.	Cómo está conformada esa métrica.
Cálculo.	Describir como se calculará.
Guía de Análisis.	Cómo se interpretan los datos de la medición). Parámetros de aceptación o rechazo
Forma de Obtención.	Lugar de donde se obtiene los datos.

La definición de objetivos y de métricas depende de la naturaleza de cada una de las empresas, así como la misión y la visión de esta. Para cada una de las métricas definidas anteriormente se deben de definir sus guías.

Por cada métrica se debe establecer la frecuencia, un formato de reporte y el responsable. La frecuencia depende de la naturaleza de la empresa.



A lo largo de este capítulo se analizó cada una de las áreas de proceso del nivel dos del modelo de madurez, se presentó una propuesta de implementación, la cual incluye las metas y prácticas genéricas, un flujo de operación, roles y actividades, así como los componentes a ser generados. Lo anterior proporciona una guía para la administración de proyectos de software, que incluye las mejores prácticas definidas en el modelo de madurez, sustentadas por la metodología de desarrollo de Microsoft. Lo que lleva a generar software de calidad que se entregue en tiempo, libre de errores, que cumpla los requerimientos del cliente, en costo, tiempo, funcionalidad y desempeño logrando así su satisfacción.

En el siguiente capítulo se evalúa el impacto directo del modelo dentro de la empresa tomada como caso práctico, midiendo la mejora después de la implementación.

Conclusiones.

La guía para la administración de proyectos de software propuesta en la sección 2.2 del presente capítulo, la cual cubre las metas, programas y subprogramas definidos en la sección 2.1, permiten generar software de calidad, mediante la implementación de las áreas de proceso definidas en el modelo. El contar con un marco de trabajo que establezca las mejores prácticas a ejecutar durante la administración de proyectos de software, facilita el proceso de gestión, ofrece una propuesta para la operación diaria y promueve la mejora continua, mediante la identificación de las áreas de oportunidades. Cada una de las áreas de procesos del modelo se encuentran estrechamente relacionadas y sus actividades y prácticas deben de ser ejecutadas en cada una de las fases del ciclo de vida de un proyecto y tienen un impacto en toda la organización. La definición, institucionalización y mejora de procesos es el camino para organizar una empresa y afecta directamente la calidad del producto desarrollado.



Capítulo 3.

Implementación y evaluación del modelo.

3.1. Guía metodológica para la administración de proyectos de software.

A continuación se presenta una guía para administrar proyectos de software:

Análisis de la organización.

Identificar claramente las fortalezas y debilidades de la organización, mediante la evaluación de la ejecución y apego a cada uno de los programas, subprogramas y actividades presentadas en la sección 2.1 del capítulo 2 con el objetivo de identificar el nivel de madurez de la organización.

Definición del ciclo de vida del producto y del proyecto.

Definir el plan de negocio, la idea central del producto, el ciclo de vida del proyecto, la operación, las disposiciones y las actualizaciones. El presente trabajo se centra en los proyectos de desarrollo para lo cual se debe definir una política organizacional de proyectos y el conjunto de fases del proyecto desde su inicio hasta su fin, así como las relaciones correspondientes a las operaciones de la organización. Debe contener:

- Fases, transición de una fase a otra.
- Trabajo técnico que deba realizarse, entregables.
- Como controlar y aprobar cada fase.
- Trazabilidad, la forma en la que los componentes evolucionan durante el proyecto.



No existe una manera única, que sea la mejor, para definir el ciclo de vida ideal de un proyecto. Se puede estandarizar todos los proyectos con un ciclo de vida único o elegir el más apropiado para el proyecto en cuestión.



Figura 14. Ciclo de vida de desarrollo de software, creación propia.

El presente trabajo propone que en cada una de las fases del ciclo de vida del proyecto se implementen todas las áreas de proceso del modelo de madurez. Así mismo se propone definir e implementar las actividades de una fase antes de iniciar con la siguiente fase, esto permitirá ajustar las fases anteriores, al diseñar e implementar la siguiente fase, lo que permite evitar la parálisis por análisis, y de la misma forma la calidad debe ser evaluada durante toda las fases del ciclo de vida y en cada uno de las áreas de proceso del modelo de madurez.



Organización del equipo de proyecto.

Definir claramente a través de roles a los integrantes del equipo de proyecto, y el organigrama de la organización. Se debe describir:

- Roles, funciones, responsabilidades y autoridades.
- Objetivos, metas y entregables.
- Perfil, conocimientos, habilidades y valores.
- Capacitación requerida, para desempeñar funciones y procesos.
- Métricas de desempeño que describa la obtención, análisis, evaluación y almacenamiento de datos utilizados para las métricas.

A continuación se describe el ciclo de vida propuesto en el presente trabajo.

Visión.

En esta fase se crea una visión compartida del proyecto entre los principales responsables, se definen los objetivos, metas, alcances y restricciones individuales y del equipo, todos los interesados coinciden en la definición del ámbito del sistema y las estimaciones. Entendimiento de los requisitos, como evidencia de la fidelidad de los casos de uso principales. Se obtiene el compromiso para con los requerimientos por parte de los participantes y se identifica e involucra al personal relevante.

Durante esta fase se debe elaborar:

Especificación de requerimientos. Debe contener la declaración del problema, establecer los objetivos del negocio, definir los requerimientos del usuario y el proceso existente. Identificar los perfiles de los usuarios quienes van a ser beneficiados de la solución. Definir el alcance y el ámbito de proyecto.



El modelo de negocio, concepto y diseño de estrategias de solución. Valoración preliminar del riesgo, identificación y fuentes de incertidumbre. Actores, casos de uso y escenarios básicos que definen la funcionalidad y prototipos exploratorios para probar conceptos, se decide que será producido en cada ciclo y la rastreabilidad del requerimiento.

Propuesta de proyecto. Estimación del costo en recursos y tiempo del proyecto. Iteraciones y entregables por iteración, formas de pago y vigencia, Se hacen estimaciones iniciales de esfuerzo y tamaño.

Glosario inicial. Homologar la terminología del proyecto.

Durante la fase se debe de:

Administrar los requerimientos. Administrar las plantillas de especificación de requerimientos y propuestas, sus versiones, cambios, revisiones y aprobaciones. Administrar las minutas de reuniones con los clientes, compromisos y acuerdos con los mismos, administrar el levantamiento de requerimientos, solicitudes, especificaciones y propuesta de los proyecto, con métodos de estimaciones, tiempos, iteraciones y entregables. Aprobación de propuestas. Administran los cambios de los requerimientos durante el proyecto, impacto de los proyectos en sistemas actuales y matriz de rastreabilidad.

Planificar. Se debe de planificar el inicio y fin de elaboración de requerimientos, el tiempo de análisis de requerimientos y estimaciones, la presentación y aprobación de propuestas, se define la comunicación con la fase subsecuente, mitigación de riesgos, capacitación, generación y presentación de reportes de revisión de administración de requerimientos.



Monitoreo y control. Elaboración de requerimientos, estimaciones, propuestas, aprobaciones, seguimiento a riesgos, compromisos y acuerdos internos y externos, revisión de las actividades, adherencia a los procesos, identificación de inconsistencias en requerimientos, desempeño contra el plan.

Medición y análisis. Evalúa objetivamente la adherencia de los procesos de administración de requerimientos contra la descripción de procesos, estándares, procedimientos e incumplimiento. Se miden y evalúan los tiempos de: Elaboración de requerimientos y presentación de propuestas. Calidad de la especificación de requerimientos.

Administración de configuración. Designar el lugar de los productos de trabajo plantillas y documentos de proyectos de requerimientos, propuestas y aprobaciones del cliente. Procedimientos de privacidad, seguridad, respaldo de repositorio y publicación de documentación.

Planeación.

Se elaboran los planes de trabajo de cómo crear la solución y se determina que se va a desarrollar, se crea el diseño de la solución, se prepara el calendario para los diferentes entregables, involucra el análisis de requerimientos, los diseños conceptuales, lógicos y físico. Se crea la especificación funcional y la arquitectura de la solución, diagrama de descomposición funcional, estructura de desglose de trabajo, modelo visual del software, diagramas de secuencia, de colaboración, de clases, de transición de estados, vista de componentes, dependencia de tareas y de puesta en marcha, se especifican los ambientes de desarrollo y prueba, criterios para acciones correctivas basadas en desviaciones del plan del proyecto; Lista, revisión histórica y prioridades del riesgo y frecuencia del monitoreo.

Durante esta fase se debe elaborar:



Planes. Plan maestro del proyecto, que incluya la acta de constitución del proyecto, integración del proyecto, enunciado del alcance, gestión de tiempos, como dirigir y gestionar la ejecución del proyecto, como supervisar y controlar el trabajo, el control integrado de cambios; Preparación del presupuesto de costos; Planes de calidad, comunicaciones, riesgo del proyecto, compras, adquisiciones, contratación, capacitación y administración de la configuración donde se identifica los tipos de datos que deben ser administrados así como los repositorios. Planificación de recursos, generación y presentación de informes a niveles altos de dirección e involucrados importantes del proyecto, auditorías y revisiones de datos del proyecto, definición de hitos o momentos importantes del proyecto

Cronograma. Definición del equipo de trabajo, actividades, secuencia y duración de las mismas, desarrollo del cronograma y comunicación del mismo.

Durante la fase se debe de:

Administrar los requerimientos. La documentación de los planes, versiones y cambios, cronograma de actividades, diseño conceptual, lógico y físico, diagramas, modelos visuales y vistas, inconsistencias en los requerimientos, listado de riesgo.

Monitoreo y control, Medición y análisis. Elaboración de planes, cronogramas, diseños, modelos y vistas, compromisos internos y externos, apego al proceso.

Administración de configuración. Designar el lugar de los productos de trabajo plantillas, documentos y planes de proyectos. Respaldo de repositorio y publicación de documentación.



Desarrollo.

Durante esta fase el equipo crea la solución. Se convierten los diseños en código, se implementa la solución, documentación del código y la infraestructura. El equipo debe iniciar el ciclo de desarrollo, crear prototipo de la aplicación, desarrollar los componentes, construir la solución, cerrar la fase y realizar pruebas de concepto.

Durante esta fase se debe elaborar:

Código fuente y ejecutables. Diseñar y desarrollar la capa de presentación, datos y negocio, especificaciones de seguridad, desarrollo iterativo, desarrollo de componentes, encapsular el hardware y las dependencias, mejorar el mantenimiento y la extensibilidad, reutilización de componentes, utilización de marcos de trabajo, verificación del ambiente.

Integración. Se compilan y prueban los módulos y unidades, pruebas de interacción y desempeño, integración de componentes ejecutables en ambientes y elementos de base de datos, interoperabilidad con sub.-sistemas y sistemas.

Durante la fase se debe de:

Administrar los requerimientos. Administración de código fuente, versiones y cambios, formatos de diseño y plantillas de liberación, líneas base de código.

Planificar. Auditorias de código, reportes de avances de proyectos, de variaciones, estatus y compromisos.

Monitoreo y control. Desarrollo del código fuente, avances conforme al plan y al cronograma, tiempos de estimaciones, implementación de estándares y mejores prácticas, cumplimiento de requerimientos, inconsistencia con los requerimientos, ejecución de diseño, pruebas de desempeño, problemas del desarrollo, acciones



cuando los proyectos presentan desviaciones, informes internos y externos del proyecto al personal relevante, registro del riesgo del proyecto, revisión, estatus, probabilidad, prioridad y severidad del riesgo, seguimiento e informe de los hitos o logros del proyecto de acuerdo al plan.

Medición y análisis. La calidad del código fuente, tiempos de actividades comunes, cantidad de errores dentro de pruebas unitarias, tipos de errores, apego al proceso, registro de desviaciones significantes.

Administración de la configuración. Repositorios del código fuente, política de respaldo y líneas base, reportes de los hitos del proyecto.

Estabilización.

Durante esta fase el equipo ejecuta la integración y pruebas betas de la solución, el equipo prueba los escenarios de liberación. El equipo se centra en identificar, priorizar y resolver problemas, la solución es preparada para liberarse, revisiones, inventarios, listas maestras de los datos o componentes del proyecto, estatus de los repositorios.

Durante esta fase se debe de:

Probar la solución. Implementar plan de pruebas para validar la solución, una vez que la solución es considerada estable, una prueba piloto es conducida en un ambiente de pruebas. Una prueba rigurosa incluye:

- Pruebas de componentes.
- Pruebas de funciones del negocio.
- Pruebas de interfaces de usuario.
- Pruebas de base de datos.



- Pruebas de infraestructura.
- Pruebas de seguridad.
- Pruebas de integración.
- Pruebas de rendimiento y volumen.

Prueba piloto. Liberación de la solución en un ambiente y pruebas de la solución con usuarios y escenarios reales usados.

Durante la fase se debe de:

Administrar los requerimientos. Plantilla de pruebas, la matriz de pruebas, los casos de pruebas, la evidencia de las pruebas.

Monitoreo y control. El apego al plan y cronograma de pruebas, el cumplimiento de los requerimientos del cliente, corrección de errores.

Medición y análisis. La cantidad de defectos encontrados, el tiempo invertido en la corrección de errores y pruebas de la corrección, los tipos de errores, responsables.

Administración de la configuración. Repositorios de casos de pruebas y evidencias de pruebas.

Liberación.

Durante esta fase el equipo libera la solución tecnológica y componentes, estabiliza la liberación, transfiere el proyecto a operaciones y soporte y obtiene la aprobación final del cliente, después de liberar el equipo realiza una revisión del proyecto y encuesta de satisfacción del cliente

Durante esta fase se debe de elaborar o ejecutar:



Procedimientos de liberación. Documentación formal de la liberación.

Liberación y estabilización. Finalización de los componentes actuales y liberación en sitio.

Revisión del proyecto. Revisión después del proyecto con el cliente y equipo de proyecto, revisión histórica de las estimaciones, comparación del desempeño del proyecto.

Durante la fase se debe de:

Administrar los requerimientos. Versión final de todos los documentos, componentes de desarrollo, código fuente, lista de revisión de acciones correctivas tomadas y evidencia de las mismas.

Planeación. Plan de capacitación, incorporación de lecciones aprendidas.

Monitoreo y control. Problemas y errores de la solución en ambiente de pruebas de cliente o producción.

Medición y análisis. Efectividad, desempeño de la solución y satisfacción del cliente, inconsistencias con los requerimientos, analizar el resultado de las acciones correctivas tomadas.

Administración de la configuración. Repositorios de código y documentos, aprobaciones finales del cliente.



Calidad de los procesos.

Se define, describe y documentan los procesos, estándares y procedimientos en conjunto con los miembros del equipo, se debe asegurar que los procesos son implementados como se planearon. Proveer a los altos niveles de administración la apropiada visibilidad de los procesos, evolución, evaluación de los procesos y es el responsable de la mejora continua.

Se debe:

Administrar los requerimientos. Se administran la descripción de los procesos, formatos y plantillas que den sustento a los procesos, modificaciones o mejoras a las plantillas. Evaluaciones, auditorías, listados de verificación y reportes de informes.

Planeación. Plan para implementar el proceso, auditorías de proceso y producto, criterios de evaluación y evaluaciones, así como responsables de la evaluación. Reportes con la alta dirección, actividades, estatus y resultados del monitoreo del apego al proceso, medición del cumplimiento de objetivos, como se obtienen los datos y la frecuencia.

Monitoreo y control. El apego al proceso, cumplimiento de los objetivos, la implementación de políticas y procedimientos, se identifican las áreas de oportunidades.

Medición y análisis. El impacto del proceso en la calidad de los productos desarrollados, en tiempo, costos, efectividad, errores, defectos, conocimiento del proceso por el equipo, información estadística de la mejora del proceso.



Administración de la configuración. Reportes, revisiones, repositorios, almacenamiento de datos de medición tanto de apego como de mejora, publicación de procesos, niveles de seguridad de formatos y plantillas de proceso.

3.2 Apego al proceso.

En el capítulo anterior se definieron las áreas de proceso o programas a ser ejecutados como mejores prácticas administrativas, durante el proceso de desarrollo de software, establecidas en el modelo de madurez. En cada una de las áreas de proceso se estableció un procedimiento en el cual se definen: Las actividades, miembros que participan y la documentación sugerida a ser generada. El proceso de evaluación del nivel dos del modelo de madurez consiste en el apego al proceso, el cual debe ser definido e institucionalizado. Con el objetivo de medir el apego al proceso a continuación se establecen una serie de auditorías o revisiones en las distintas etapas del ciclo de vida de desarrollo. Las cuales consisten en preguntas correspondientes a los elementos neurálgicos que deben ser cubiertos en cada una de las etapas.

Evaluación de la solicitud de nuevo proyecto.

El proceso de administración de requerimientos incluye la solicitud y el requerimiento.

La solicitud contiene la primera parte de la especificación de requerimientos de software, la estimación del requerimiento y la propuesta del proyecto. A continuación se establecen los elementos a utilizar para auditar el proceso de solicitudes.

Se inicia con la evaluación de la primera etapa de la especificación de requerimientos.



Verificación	Si	No	Observaciones
¿Se ha definido el objetivo del sistema a desarrollar?			
¿Se ha descrito el alcance de la especificación?			
¿Se han indicado aspectos funcionales?			
¿Se han indicado restricciones y supuestos que deben ser considerados para el desarrollo?			
¿Se han listado cada uno de los casos de uso que conforman la funcionalidad del producto?			
¿Se ha generado un diagrama con los casos de uso y sus correspondientes actores para el sistema a desarrollar?			

Así mismo se evalúa la propuesta de proyecto.

Verificación	Si	No	Observación
¿El requerimiento cuenta con el resumen ejecutivo, que corresponda al ERS?			
¿Se especifico el plazo de ejecución de acuerdo a las estimaciones?			
¿Se definió el plan de entrega?			
¿Se especifico la oferta económica?			
¿Se definieron las responsabilidades?			

En los primeros cinco meses del año 2008, se han elaborado un total de 67 proyectos nuevos que incluyen solicitudes y requerimientos. Durante el proceso de revisión de calidad se obtuvieron los datos mostrados en la figura 14, correspondientes a las solicitudes aceptadas y rechazadas. Se identifico que en el primer mes todas las solicitudes fueron rechazadas, por lo que se tomo la decisión de generar un informe con la causa del rechazo, así como buscar directamente al consultor de negocio responsable del requerimiento con el objetivo de realizar las



modificaciones conjuntamente con el personal de calidad, para que no existiera una segunda devolución debido a que esto incrementaría el tiempo del proyecto.

Se observó una tendencia a la baja en el rechazo de las solicitudes excepto en el mes de mayo que presentó un incremento, esto fue debido a la incorporación de un nuevo consultor de negocio, al cual no se le proporcionó un curso adecuado de inducción sobre el llenado de los documentos, esto evidenció la falta de claridad en las plantillas de los documentos lo que obligó a documentarlas con mayor detalle y la necesidad de capacitación.

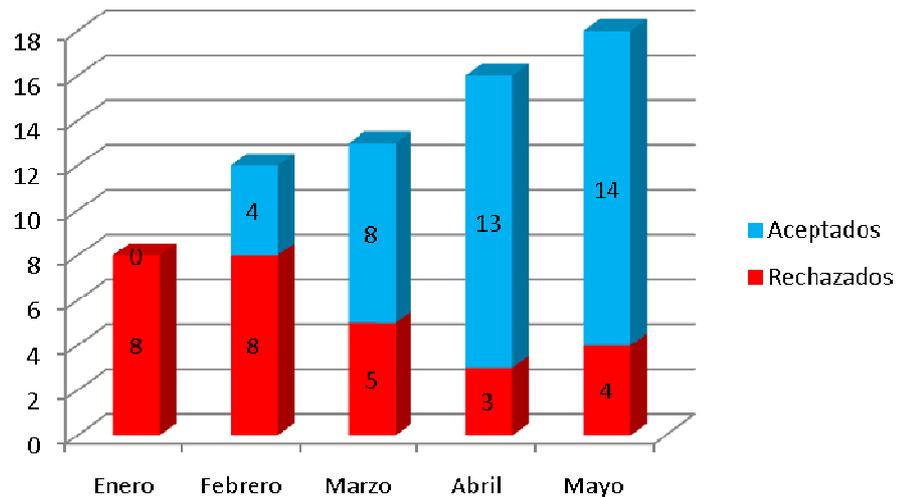


Figura 15. Solicitudes rechazadas y aceptadas, creación propia.

Las principales causas de rechazo de las solicitudes son:

- Ambigüedad en las plantillas. La documentación de las plantillas no es clara, lo que ocasiona que se entienda diferente entre los consultores de negocio.



- Documentación incompleta. Los consultores de negocio no llenan completos los documentos, al dejar elementos de la plantilla que deben ser llenados por ellos.
- No se incorporan observaciones. Durante las reuniones de análisis de las solicitudes, los analistas e ingenieros de pruebas, realizan observaciones que no son incorporadas dentro de la especificación de requerimientos de software por los consultores de negocio.
- Diferencias en los plazos. Los plazos de ejecución definidos en la propuesta no corresponden con el archivo de estimaciones, diferencias entre el tiempo que se cobra y el total de actividades realizadas, entre las que se encuentra la administración de configuración y el aseguramiento de la calidad, etc.
- Análisis de elementos. Elementos como restricciones y supuestos se establecen como no aplica, sin realizar un análisis por parte de los consultores de negocio.

En la siguiente figura se muestran los datos de las 28 solicitudes rechazadas por los motivos antes mencionados, las solicitudes son rechazadas en promedio por 1.71 razones es decir casi por dos razones.

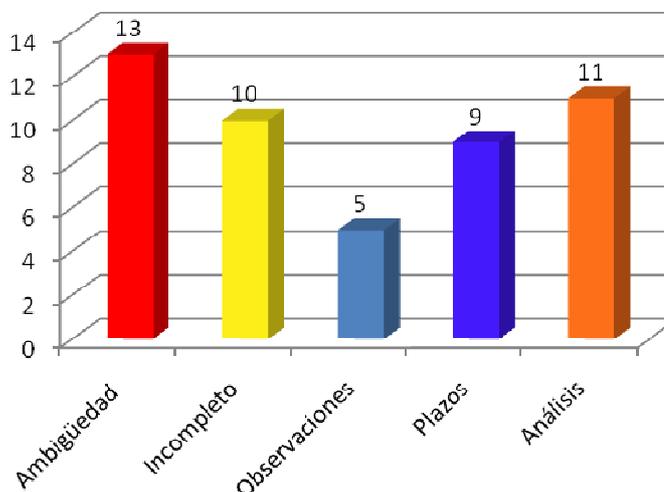


Figura 16. Razones de rechazo de solicitudes, creación propia.

Las acciones tomadas para solventar los puntos antes mencionados son:

- Documentar detalladamente las plantillas.
- Planear capacitación sobre el llenado de los documentos.
- Entrega de estimaciones hasta que el documento cuente con las observaciones realizadas.

La información antes mencionada se obtiene del archivo de desviaciones elaborado por el área de calidad.

Una vez elaborada la propuesta de la solicitud de proyecto se entrega al cliente. El cual deberá de evaluar si cumple con sus necesidades, aceptará los términos, funcionalidades, costos y tiempos expresados en ella. Debe de quedar registro de esta aceptación para posteriores aclaraciones, ya sea a través de un medio digital (correo), o físico (propuesta firmada por el cliente).



Ya que la propuesta de proyecto fue autorizada por el cliente se convierte de solicitud a requerimiento. Se procede a realizar los casos de uso detallados, definir los requerimientos no funcionales, prototipos si es que se requiere y la matriz de rastreabilidad, se vuelve a estimar los tiempos y en caso de existir diferencias se entregan los nuevos tiempos, nuevamente es presentado al cliente para su aprobación y posterior a esto, se inicia el proceso de desarrollo.

A continuación se presenta el cuestionario de evaluación de requerimientos.

Verificación	Si	No	Observación
¿Se realizó el detalle de los casos de uso?			
¿Los casos de uso detallados incorporan las validaciones de negocio requeridas?			
¿Se definieron los aspectos no funcionales?			
¿Se diseñó la interfaz del usuario?			
¿Se elaboró la matriz de rastreabilidad?			

Se identificó que en el mes de enero, todas las solicitudes aprobadas por el cliente, eran inmediatamente asignadas a desarrollo para iniciar su codificación, sin ejecutar el proceso de requerimientos, en meses posteriores se trabajó más en la implementación de los requerimientos.

Se hizo evidente la importancia de los requerimientos y la aprobación por parte del cliente, debido a que en algunas liberaciones, el cliente expresó inconformidad con los desarrollos entregados, pero al contar con el requerimiento aprobado, la causa fue imputada al cliente por aceptar un documento sin revisarlo.

También se detectó que no se realiza el proceso de re-estimación por parte de desarrollo, en ocasiones al realizar el análisis más profundo se encuentra con impactos no contemplados anteriormente, los cuales afectan el tiempo de desarrollo, pero estos tiempos no son informados por lo que desarrollo absorbe esos tiempos.



A continuación se muestra la información de las solicitudes que se convierten en requerimientos, aquellas que son aprobadas para su desarrollo, las que si se elaboro la documentación solicitud (casos de uso detallados, prototipos, matriz de rastreabilidad etc.), las que su documentación fue rechazada por calidad , aquellas que después del análisis se determino que el tiempo era diferente, por impactos no contemplados a la aplicación durante la etapa de solicitud, por lo que se realiza una nueva estimación de los tiempos identificando claramente todos los elementos, y por ultimo aquellas que se presentaron al cliente para su autorización ya que se contaba con el requerimiento desarrollado completamente.

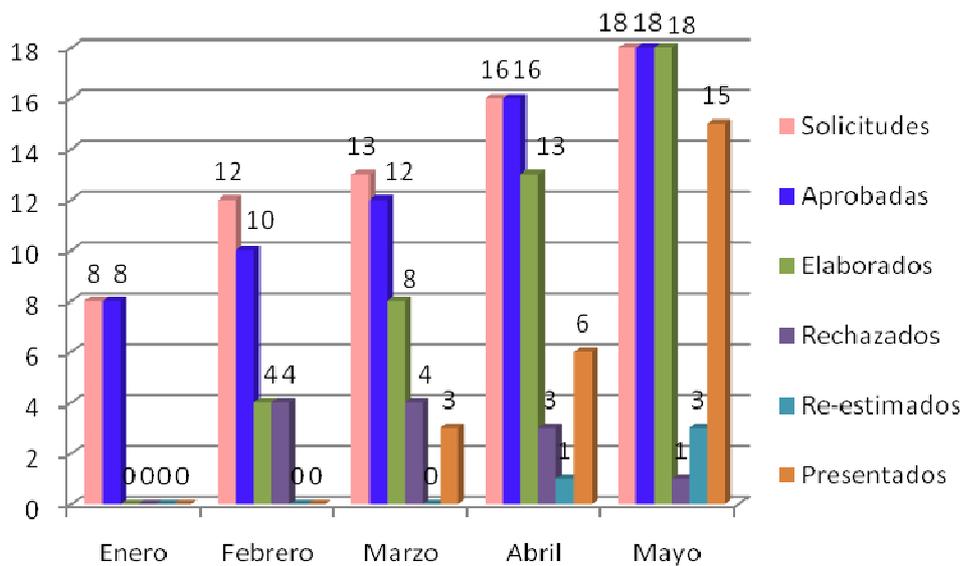


Figura 17. Requerimientos de solicitudes aprobadas, creación propia.

Las principales causas de rechazo de requerimientos por calidad son:

- No se definen los requerimientos no funcionales.
- No se elabora la matriz de rastreabilidad.



Las acciones tomadas para solventar los problemas antes mencionados son:

- Capacitación en el proceso de requerimientos.
- Políticas de entradas para el proceso de desarrollo.

Una vez que la solicitud de requerimiento de software ha sido aceptada por el cliente y el mismo está conforme con la especificación de requerimiento de software. La solicitud pasa a ser realmente un proyecto de desarrollo de software y se inicia la planificación del mismo.

Evaluación de planes.

El área de proceso de planificación establece que se deben generar los siguientes planes:

- Plan de desarrollo de software.
- Plan de soporte del proyecto.
 - Plan de administración de riesgos.
 - Plan de comunicación.
 - Plan de medición.
 - Plan de administración de la configuración.
- Plan de pruebas.

A continuación se describen los elementos que deben contener los distintos planes elaborados.

Verificación	Si	No	Observaciones
¿Se ha descrito el plan de desarrollo de software?			
¿Se han definido el alcance del plan de desarrollo de software?			



¿Se ha definido el objetivo de negocio que intenta cubrir el proyecto?			
¿Se han descrito la estructura organizacional del equipo de proyecto?			
¿Se ha identificado el ciclo de vida y las fases e iteraciones a ser utilizadas en el proyecto?			
¿Se ha referenciado o descrito las estimaciones de tamaño, esfuerzo y cualquier cálculo e información de soporte?			
¿Se ha referenciado a un documento separado o incluido las estimaciones de costo e información de soporte?			
¿Se ha descrito o referenciado un plan de recursos?			
¿Se ha descrito o referenciado un plan de capacitación?			
¿Se ha descrito o referenciado un plan de administración de riesgo?			
¿Se ha descrito o referenciado un plan de pruebas de software?			
¿Se ha descrito o referenciado un plan de mediciones?			
¿Se ha descrito o referenciado un plan de comunicaciones?			
¿Se ha descrito un plan de dependencias?			
¿Se identificaron estándares y procedimientos aplicables al proyecto?			
¿Se han especificado la frecuencia y asistencia de las reuniones del equipo de proyecto?			



Durante la planificación se definen los planes a seguir la asignación de actividades a recursos y la calendarización de las mismas. Inmediatamente después se inicia la fase de codificación, pruebas y despliegue, que consiste en el desarrollo del producto.

Inicialmente se pretendió desarrollar los planes por cada uno de los requerimientos, pero por los tiempos tan cortos y la cantidad de requerimientos, que deben ser desarrollados, se detectó que no era viable generar planes por cada requerimiento por lo que se procedió a desarrollar planes fijos que no tuvieran que ser ejecutados por requerimientos.

Los planes definidos cubren al cien por ciento los elementos contenidos en el cuestionario de evaluaciones.

Evaluación de plan de mediciones.

Posterior a la codificación se deben de iniciar las mediciones, se definen las métricas que sean importantes a la organización, estas métricas deben de documentarse en un plan y tienen que ser definidas por la dirección. A continuación se evalúa los elementos básicos que debe de contar el plan de mediciones.

Verificación	Si	No	Observaciones
¿Se ha descrito el propósito del plan de medición?			
¿Se ha descrito el alcance del plan de medición?			
¿Se han indicado para c/u de los participantes el nombre y el rol?			
¿Se han identificado los objetivos de negocio relacionados con el proyecto?			
¿Se han definido las métricas?			



¿Para cada métrica se definió el rol al que va dirigida?			
¿Se ha completado la información requerida para describir en forma detallada cada métrica?			
¿Se han descrito los procedimientos, herramientas y métodos relacionados con la obtención de las métricas identificadas?			

El objetivo de la evaluación antes mencionada esta direccionada al plan de medición, el cual es el sustento para realizar las métricas, cabe mencionar que el plan definido en FINASIST, cumple al cien por ciento las preguntas presentadas.

De acuerdo a lo expuesto en el capítulo dos en el apartado de Indicadores, se establece a continuación las métricas sugeridas y los datos obtenidos.

Requerimientos en tiempo	Descripción
Concepto Medible.	Cantidad de proyectos en tiempo.
Indicador.	Cantidad.
Métrica.	FR = Fechas reales. FE = Fechas estimadas.
Cálculo.	Contar aquellas que $FE > FR$
Guía de Análisis.	Si existe alguna diferencia se debe identificar la causa de la desviación.
Forma de Obtención.	Fechas de entrega informadas al cliente y archivos de liberación.

La aplicación de la métrica durante los cinco primeros meses del año mostro que la mayor parte de los requerimientos se entregan fuera de tiempo, la fase que genera este problema es la de desarrollo ya que inicia tarde y entrega tarde, esto se debe al cambio de prioridades y a la asignación de tiempo de desarrollo a otras actividades como son:



Juntas no planeadas, apoyo en la resolución de dudas a la gente de pruebas y al personal de atención al cliente y principalmente por el tiempo dedicado a la corrección de errores. Después de detectado lo anterior se optó por establecer que las juntas se realicen en la tarde y con un día de anticipación, adicionalmente se estableció que los consultores de negocio apoyaran en primera instancia al personal de pruebas y atención al cliente, con el objetivo de no distraer al área de desarrollo, se definieron días de 6 horas, con dos horas diarias destinadas a juntas, corrección de errores o cualquier eventualidad.

En los meses posteriores se observó una leve mejora en las entregas de los proyectos, pero seguían presentándose retrasos, el análisis de esto determinó que, durante la fase de desarrollo y pruebas se encontraban casos que no estaban contemplados dentro de los requerimientos, así como afectaciones adicionales. Se detectaron deficiencias en la matriz de rastreabilidad, ya que se llena solo como requisito, en un formato difícil de llenar.

Se procedió a modificar el formato para hacerlo más amigable y entendible, así como a prestar mayor atención a la misma. Pero aún después de realizar lo antes mencionado se presentan problemas y la causa son los impactos no contemplados en la aplicación. La causa es que la matriz se llena de acuerdo a la experiencia del analista por lo que existe un porcentaje alto de falla, ya que si el analista no conoce el módulo en cuestión, no es capaz de ver todos los impactos. Por lo que se propuso la creación de una herramienta que ayude a la identificación de impactos de desarrollo de forma automática, se parte del supuesto que cada modificación afecta directamente en algún objeto de la base de datos, mediante la identificación de todos los objetos que puedan ser afectados por dicha modificación, se puede determinar de una manera más exacta los impactos de los cambios, adicionalmente la creación de una base de datos correspondiente a la matriz de rastreabilidad que se vaya llenando con cada requerimiento.



Para que la identificación de impactos en un futuro sea lo más eficiente posible. A continuación se muestra la información de los proyectos entregados en tiempo.

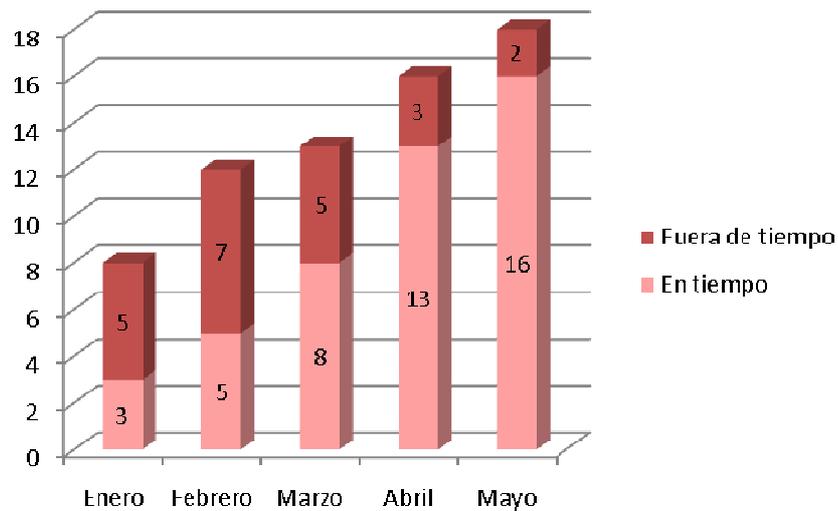


Figura 18. Requerimientos entregados en fechas planeadas, creación propia.

Si bien continúan los problemas de errores debido al análisis de impactos, estas deficiencias están mitigadas, gracias al monitoreo de proyectos que permite tomar decisiones para que los proyectos se entreguen a tiempo, la mayoría de ellas consiste en trabajar fuera de horarios laborales, lo que busca es que el personal realice su trabajo lo mejor posible desde el principio para no ser afectado.

Otra métrica importante a considerar corresponde al porcentaje de desviación de las estimaciones, el cual tiene un impacto directo en la entrega a tiempo de los proyectos.



Desviación de estimaciones	Descripción
Concepto Medible.	Promedio de porcentaje de desviación de estimaciones por mes, en cada una de las actividades del ciclo de vida del proyecto.
Indicador.	Porcentaje.
Métrica.	HE = Horas estimadas. HR = Horas reales. PD = Porcentaje de desviación. PRD = Promedio de desviaciones. TP = Total de proyectos
Cálculo.	$PD = ((HR - HE) / HE) * 100$ $PRD = (\sum PD) / TP$
Guía de Análisis.	Si el porcentaje es mayor a un 15% se debe de analizar la causa y documentar para proyectos posteriores.
Forma de Obtención.	Cronograma de actividades y tiempos por actividades.

Se evaluó el porcentaje de desviación mensual por cada una de las etapas del ciclo de vida del proyecto, definidas en el capítulo 2 página 56 y se obtuvo la siguiente información. Las actividades que presentan una mayor desviación corresponden al análisis y las pruebas.

Durante el análisis se identificó que la causa de desviaciones es el tiempo que se dedica a la especificación de requerimientos, afectada por la atención a la operación del cliente.

Esto mostro la necesidad de mayor capacitación en el uso de la aplicación. Pero el porcentaje de desviaciones mayor se encuentra en la fase de pruebas donde se llegan a encontrar desviaciones hasta de un 50% lo que obliga a mover recursos para apoyar las pruebas o que los requerimientos se tengan que entregar aun cuando no se hayan terminado las pruebas y estas se realicen después de liberado el producto.



Esto se debe principalmente a los errores detectados por pruebas ya que esto implica que se debe de regresar a desarrollo para su corrección y posteriormente se vuelve a probar, por lo que se propone adquirir una herramienta que apoye el proceso de pruebas, el cual permita grabar las pruebas para una posterior ejecución.

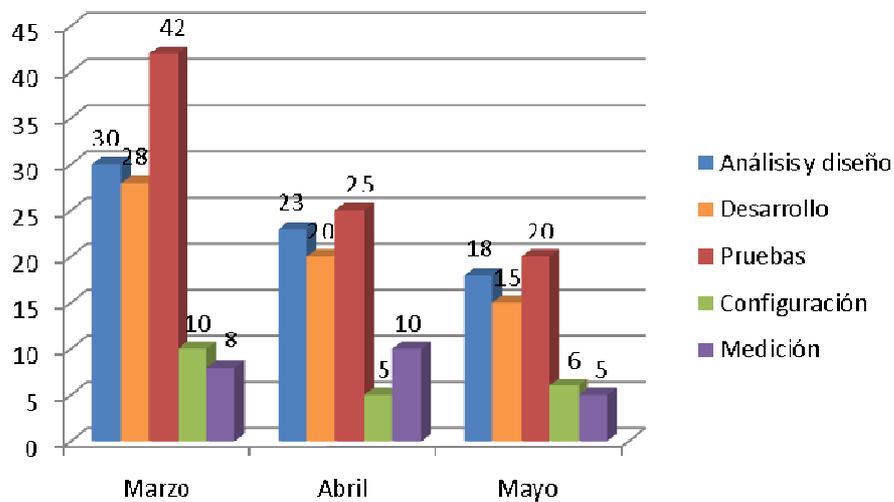


Figura 19. Porcentaje de desviaciones de estimaciones, ceración propia.

La información presentada corresponde a las actividades del ciclo de vida que presentan desviaciones. La desviación de las estimaciones de análisis no afecta directamente las fechas de entrega de los proyectos ya que estas se desarrollan antes de proporcionar fechas de entrega a los clientes, pero su implicación es sobre la rentabilidad de la empresa al tener un costo mayor que el cobrado al cliente. Las desviaciones de pruebas si afectan la fecha de entrega pero se solventa desviando recursos, solicitando apoyo por desarrollo o entrega sin termino de pruebas.



Las desviaciones de la configuración y la medición no afectan la entrega de proyectos ya que son actividades que se ejecutan en paralelo a la codificación del proyecto pero el impacto es en las ganancias de la empresa.

El elemento más importante a considerar, es la cantidad de errores presentados durante el proceso de desarrollo, de ahí la necesidad de la siguiente métrica, la cual consiste en cuantificar la cantidad de errores presentados en lo que va del año.

Errores	Descripción
Concepto Medible.	Cantidad de errores.
Indicador.	Cantidad.
Métrica.	NE = Número de errores.
Cálculo.	NE.
Guía de Análisis.	Contabilizar la cantidad de errores presentados.
Forma de Obtención.	Actividades calendarizadas.

La cantidad de errores presentados ha mostrado una disminución en la generación de estos, a excepción del mes de marzo, en el cual se observó un incremento, la principal causa, fue la incorporación de un nuevo cliente, el cual realizó pruebas integrales al sistema, lo que evidenció errores de procesos que no han sido probados. Adicionalmente el impacto en la disminución de errores, con la implementación del modelo no ha sido tan exitoso como se esperaba, esto es porque la aplicación cuenta con errores, que no son causados por la generación de nuevos proyectos, si no están dentro de la aplicación y que no han sido detectados.

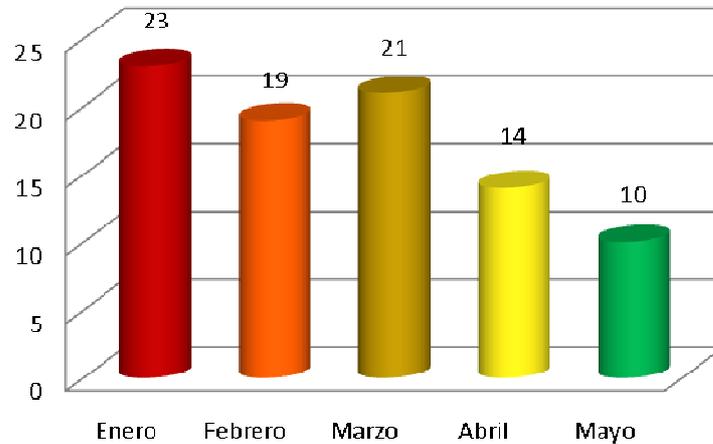


Figura 20. Cantidad de errores por mes, ceración propia.

La gran cantidad de parámetros y configuraciones del sistema, hace muy complicado el poder probar todo el sistema y todo el conjunto de posibles combinaciones, más aun es imposible realizar esto por cada nuevo requerimiento, con el objetivo de poder atacar el problema presentado, se analiza la posibilidad de incorporar una herramienta de pruebas automática, dentro de las opciones se encuentra: IBM Rational Robot v 7.0.1, Mercury Test director, Visual Studio Team System, entre otras.

Un elemento adicional de vital importancia consiste en la administración y control de todos los archivos generados durante el proceso el cual corresponde a la siguiente área de proceso, administración de la configuración.

Evaluación de administración de la configuración.

La administración de la configuración consiste en la definición de un plan de administración de versiones no solo del código fuente del proyecto si no de toda la documentación que se genere dentro del ciclo de vida del desarrollo.



Durante esta actividad se promueven las líneas base, que son estados en el tiempo en el cual se congelan los componentes requeridos para la liberación de un proyecto.

Verificación	Condiciones		Observaciones
	Si	No	
¿Se ha definido el propósito del plan de administración de la configuración?			
¿Se ha descrito el alcance del plan de administración de configuración?			
¿Se ha especificado toda la información referida a los ambientes de trabajo del proyecto?			
¿Se han listado los componentes de configuración del proyecto?			
¿Se han identificado las librerías de software que utiliza el proyecto?			
¿Se han descrito en qué puntos del ciclo de vida del proyecto serán establecidas las líneas base?			
¿Se han descrito los procedimientos para el pedido de cambio de los productos de trabajo del proyecto?			
¿Se han descrito auditorías?			
¿Se han descrito los recursos requeridos para implementar las actividades específicas de administración de configuración?			
¿Se han descrito las actividades relacionadas con el mantenimiento y			



almacenamiento de las líneas base del proyecto?			
---	--	--	--

Esta actividad puede ser realizada de una manera eficiente mediante el apoyo en herramientas de control de versiones, que permitan automatizar estas actividades, agiliza el proceso y evita posibles errores durante el almacenamiento, actualización de componentes, lo que facilita la incorporación de varios recursos a un proyecto. Por lo que la evaluación se encuentra más centrada en la definición del plan y proceso más que en la actividad misma.

A continuación se realizan algunas recomendaciones de software de control de versiones:

- Visual Source Safe.
- Borland Starteam.
- Subversión.
- Free VCS / Jedi VCS.
- CVS.
- Sourcegear Vault.
- QSC team Coherence.
- Microsoft Team System.
- Perforce.

Debido a que la mayoría de los desarrollos son con tecnología de Microsoft, se decidió implementar el Visual Source Safe, ya que este se integra directamente en los ambientes de desarrollo utilizados.

La implementación se realizó primero con un solo módulo del sistema, con el objetivo de analizar la facilidad de uso, las ventajas y desventajas.



La prueba fue un éxito completo, ya que es una herramienta, muy fácil de usar, no requiere una capacitación exhaustiva a los usuarios finales, al integrarse dentro de los ambientes de desarrollo facilita y promueve su uso, la capacidad de administrar otro tipo de archivos permite administrar el total de documentos utilizados durante el proceso de desarrollo. De lo antes mencionado se obtienen los siguientes datos.

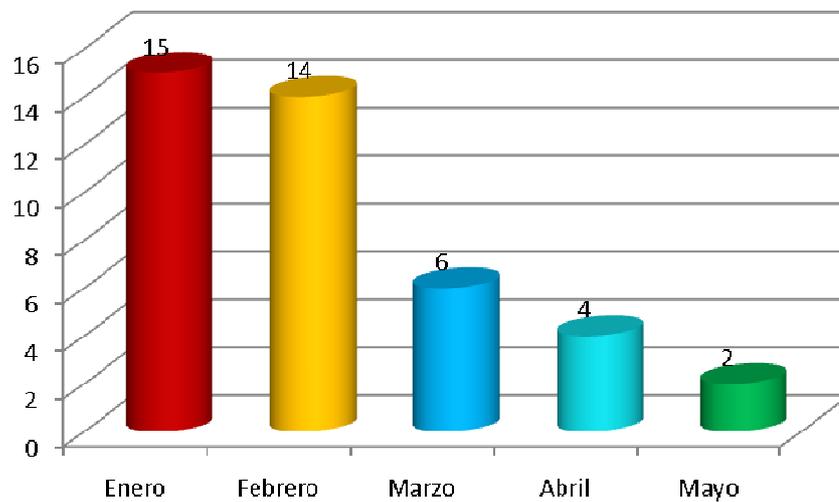


Figura 21. Errores por control de versiones, creación propia.

Los errores que se presentan todavía en el control de versiones consiste, en el proceso manual de pasar modificaciones entre los diferentes clientes, o en el cambio de ambientes del mismo cliente, los errores por causa de que múltiples programadores trabajen en un mismo archivo, que se pierdan versiones, han sido eliminados casi en su totalidad con el uso de la herramienta de control de versiones.



3.3 Impacto del modelo.

Después de un estudio de cada uno de los errores, se establecen las siguientes clasificaciones, en la que se agrupan los errores de acuerdo a su naturaleza:

- Desarrollo.
- Entregas.
- Análisis.
- Diseño.

La presente grafica muestra el porcentaje de errores de acuerdo a cada una de las clasificaciones.

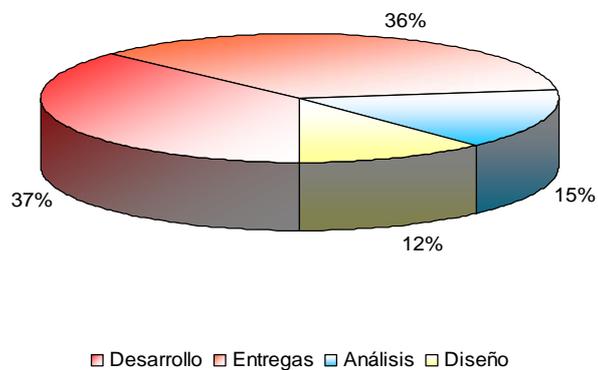


Figura 22. Clasificación de errores, creación propia.



Errores de desarrollo.

La mayor parte son detectados y reportados por el cliente ya que se presentan en producción o en alguno de sus ambientes de pruebas, algunos de estos son identificados durante el ciclo de vida del proyecto. Dentro de estos errores se encuentran:

- **Ejecución.** Generan un error en la aplicación o en los componentes de la aplicación que no permite la ejecución del sistema, inhabilita el sistema o no despliega el componente. Estos son de los más críticos por que detienen la operación del cliente.
- **Operación.** Se considera error de operación, algún funcionamiento equivocado del sistema, ya sea un cálculo o la generación incorrecta de información en pantalla, reporte, documento de salida o en alguna interfaz con otro sistema. Depende de la naturaleza del error puede ser crítico porque puede detener la operación del cliente o ser aislado que afecta la funcionalidad del sistema o la integridad de los datos.
- **Técnico.** Pueden presentarse durante el proceso de liberación o durante la ejecución del programa. Estos se deben a objetos o elementos que hayan sido desarrollados pero que no se entreguen al cliente durante la liberación, falta de permisos o de explicación clara y concisa del proceso de liberación. Cualquier asunto relacionado con la parte técnica.
- **Análisis.** Se presentan durante la operación del sistema y son detectados por el cliente, al determinar que con la liberación anterior se eliminó funcionalidad utilizada o se afectó algún otro proceso dentro del sistema.
- **Internos.** Se presentan durante la etapa de desarrollo o pruebas (unitarias del programador o del área de pruebas o calidad). Se pueden establecer dos tipos de errores adicionales los cuales son defectos e incidencia.



Un defecto son aquellos errores detectados durante la etapa de pruebas, calidad o inclusive durante el desarrollo, pero son errores que no corresponden al desarrollo en curso pero que son un defecto detectado en la aplicación. Las incidencias son errores generados durante el desarrollo del proyecto en curso detectados por el área de pruebas o calidad, estos son corregidos antes de la liberación al cliente.

Los responsables de estos errores son la gente de desarrollo, a pesar de que antes de liberar algún proyecto, este pase por un área de probadores o calidad, se determina dicha responsabilidad ya que son ellos los que originan estos problemas. A continuación se presentan las estrategias y actividades presentadas en esta investigación orientadas a solventar los errores antes mencionados.

	Equipo de proyecto	Administración de requerimientos	Planificación de proyectos	Monitoreo y control	Medición y análisis	Administración De configuración
Ejecución			✓	✓		✓
Operación		✓	✓	✓		✓
Técnico		✓	✓			✓
Análisis		✓	✓	✓		
Interno			✓	✓		

Figura 23. Impacto del modelo en tipos de errores, creación propia.

La clasificación de los errores en los primeros cinco meses del 2008 mostro que en lo que va del año no se ha presentado ningún error critico el cual detenga la operación de algún cliente, los errores de tipo de operación y análisis se han disminuido pero no se han erradicado y se siguen presentando principalmente por errores de los desarrolladores, causados por descuidos, incomprensión del requerimiento, falta de experiencia, presión del tiempo entre otros pero que han sido mitigados gracias a los procesos institucionalizados, el principal impacto en errores lo tiene la administración de la configuración al erradicar los errores técnicos.



Los datos obtenidos hizo evidente que no existe un solo desarrollo que sea liberado para pruebas que no contenga errores durante su proceso de codificación,. Adicionalmente el sistema sigue contando con gran cantidad de errores que por falta de uso por parte de los clientes no han sido detectados.

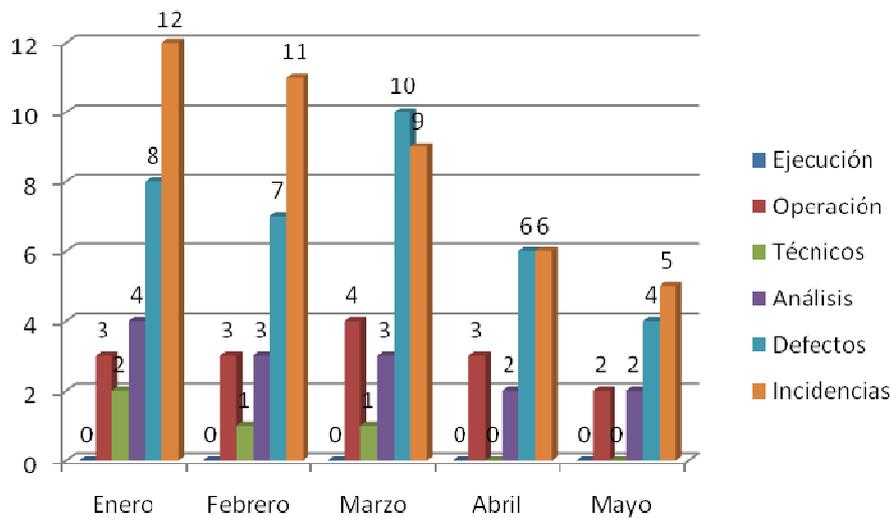


Figura 24. Errores clasificados, creación propia.

Como resultado a esto se establece la creación de un documento de arquitectura en el cual, desarrolladores con mayor experiencia, tanto en la aplicación como en el negocio, definan la forma de realizar el requerimiento, esta actividad es propia del nivel tres de CMMI pero se recomienda su uso para evitar los problemas antes mencionados. En caso de que algún programador detecte algo debe de dirigirse con el analista pertinente para realizar la sugerencia o modificación y este modifique el documento de arquitectura en caso de que se requiera. Con respecto a los errores que ya tiene la aplicación se establece que cada analista es responsable del proceso o modificación en turno y que debe verificar el correcto funcionamiento del proceso que esté trabajando para corregir posibles errores latentes a la aplicación.



Errores de entregas.

Los proyectos son entregados fuera del plazo pactado con el cliente, lo que representa que las estimaciones con las cuales se hizo la propuesta económica al cliente no son correctas y el costo del proyecto es superior al cobrado, esta diferencia es absorbida por la empresa lo que representa una pérdida, adicional pueden existir penalizaciones de acuerdo a los contratos de servicio celebrados con los clientes.

El retraso de un proyecto afecta el tiempo de inicio de otros debido a la asignación de recursos. Las principales causas de estos problemas son estimación incorrecta de tiempo, mala identificación del alcance del proyecto, planeación incorrecta, falta de seguimiento durante el proceso de desarrollo, cambios a los requerimientos y a las prioridades de los proyectos, demoras en entregas de definiciones por parte de externos.

	Equipo de proyecto	Administración de requerimientos	Planificación de proyectos	Monitoreo y control	Medición y análisis	Administración de configuración
Estimación		✓				
Alcance		✓				
Planeación			✓	✓		
Seguimiento				✓		
Cambios		✓	✓			
Prioridades		✓				
Dependencia		✓	✓	✓		

Figura 25. Impacto del modelo en entregas de proyectos, creación propia.

Los errores de entrega se han disminuido excepto en el mes de marzo en el cual se incremento debido al lanzamiento de un cliente a producción, ya que se modificaron prioridades de otros clientes por apoyar el lanzamiento.



El impacto en la disminución de las entregas tardías no se debe a la exactitud de las estimaciones o a la disminución de errores o a que el sistema no tenga errores, se debe gracias a la interrelación entre la administración de requerimientos al contar con un documento que exprese claramente las funcionalidades a desarrollar, a la planeación de proyectos que permite calendarizar adecuadamente los recursos y los tiempos de las actividades, así como el monitoreo y control de los proyectos sobre las actividades planificadas, es esto lo que ha permitido tener un incremento en la entrega a tiempo de los proyectos. Las principales causas de retrasos en los proyectos son.

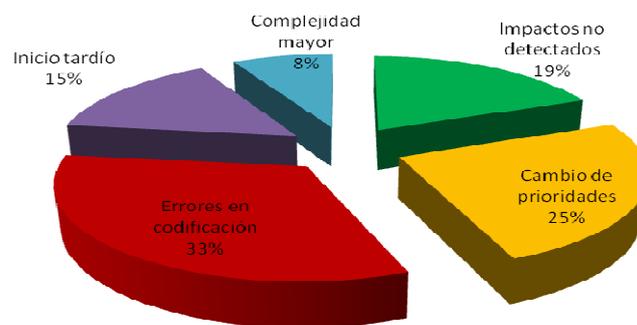


Figura 26. Causas de retrasos, creación propia.

La principal causa de retrasos corresponde a los errores en codificación, con el objetivo de atacar esta área se propone el documento de arquitectura, el cual establece claramente la forma de implementar la solución del requerimiento y se proponen auditorías de código durante el proceso de desarrollo. El cambio de prioridades es algo que es difícil de tratar pero el punto importante corresponde la falta de notificación, en el retraso de proyectos. Los impactos no detectados son consecuencia nuevamente del uso inadecuado de la matriz de rastreabilidad, mismos que se solventan con la propuesta de una base de datos para realizar esta tarea.



Errores de análisis

Los proyectos son desarrollados de acuerdo a los requerimientos entregados por parte del área de negocio, pero después de su liberación el cliente determina que no cubren sus necesidades, en funcionalidad, desempeño o tiempo de respuesta.

Los responsables de estos problemas son los encargados de llevar a cabo el levantamiento de requerimientos con el cliente, ya que en la mayoría de los casos no se tiene una aprobación del requerimiento final y no existe una visión clara del desarrollo hasta que el mismo se le es entregado.

Se debe comprender que el levantamiento de requerimientos es de las actividades más importantes durante el ciclo de vida de un proyecto. Si se cuenta con un requerimiento mal definido es seguro que todo el proyecto este mal o no cumpla con las necesidades del cliente, si el requerimiento es ambiguo o incompleto, ocasionara retrasos en el proyecto al no tener claro lo que se desea, lo que afecta tanto el tiempo del desarrollador y del consultor al tener que dedicar tiempo adicional a un requerimiento. Entre mayor sea el tiempo invertido en esta etapa, tendrá frutos importantes en las posteriores etapas del proceso de desarrollo, al permitir ser más eficiente cada una de ellas al tener claro lo que se espera del desarrollo.

	Organización del equipo de proyecto	Administración de requerimientos	Planificación de proyectos	Monitoreo y control	Medición y análisis	Administración De configuración
Requerimientos		✓			✓	✓

Figura 27. Impacto del modelo en los errores de análisis, creación propia.



Uno de los primeros elementos que se implementó, corresponde a la generación de la especificación de requerimientos de software tanto en la etapa de solicitud como en la de requerimientos.

Gracias a la generación de este documento, al involucrar al cliente durante la creación de este documento así como en la evaluación y aceptación por parte del cliente, ha permitido tener un impacto directo en este tipo de errores, los cuales eran sumamente dañinos para la empresa ya que en algunas ocasiones representaba el tirar todo el trabajo realizado e iniciar de nuevo.

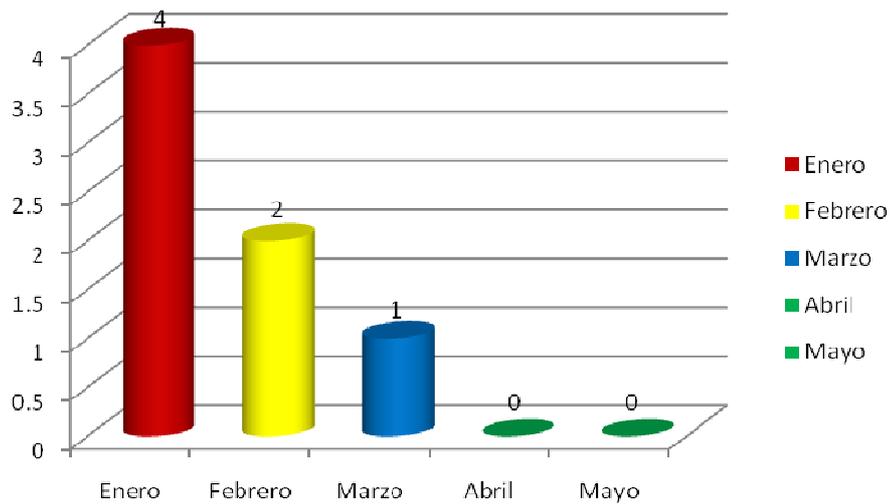


Figura 28. Errores de análisis, creación propia

Pero desde su implementación, los resultados fueron una disminución y desaparición de este tipo de errores.



Errores de diseño

Los proyectos son desarrollados pero no coincide con la definición de los requerimientos. Estos problemas son identificados en la mayoría de los casos hasta que el proyecto es liberado al cliente o una vez que el desarrollo y las pruebas son terminadas, lo que tiene un impacto directo en tiempo y costo al tener que trabajar nuevamente en un proyecto el cual se suponía terminado, genera al mismo tiempo problemas de entrega al cliente ya sea por entrega tardía al encontrar desviaciones después de terminado el proyecto o por que el cliente no acepta el desarrollo o solicita modificaciones que cubran sus necesidades, lo que nuevamente representa tiempo adicional invertido en un proyecto.

Los responsables directos por estos problemas son los desarrolladores al no tomarse el tiempo para elaborar el diseño de la solución antes de iniciar la codificación, diseño mismo que no tiene una aprobación o por asumir ciertos elementos durante el proyecto. Normalmente se inicia la codificación sin tener una visión clara de toda la funcionalidad del desarrollo, y la falta de seguimiento de los responsables de desarrollo, genera estos errores.

Otros factores que promueven estos problemas son la falta de seguimiento de los proyectos por parte de la gente de operaciones o de negocio, los cuales después de entregar los requerimientos se deslindan de los proyectos y no tienen conocimiento del producto final entregado, solo hasta que el cliente presenta inconformidades durante la operación del proyecto liberado. Otra causa es que la gente de negocio asume que el desarrollador o el responsable del desarrollo cuentan con conocimientos del negocio y que es su responsabilidad determinar errores o inconsistencias en los requerimientos.



	Equipo de proyecto	Administración de requerimientos	Planificación de proyectos	Monitoreo y control	Medición y análisis	Administración de configuración
Diseño		✓		✓		

Figura 29. Impacto del modelo en errores de diseño, creación propia

Al igual que con los errores de análisis la implementación de la administración de requerimientos tiene un impacto directo en este tipo de errores, ya que al contar con un documento en el cual se encuentra definido el requerimiento del cliente y aprobado por este mismo, evita sorpresas en la entrega del proyecto.

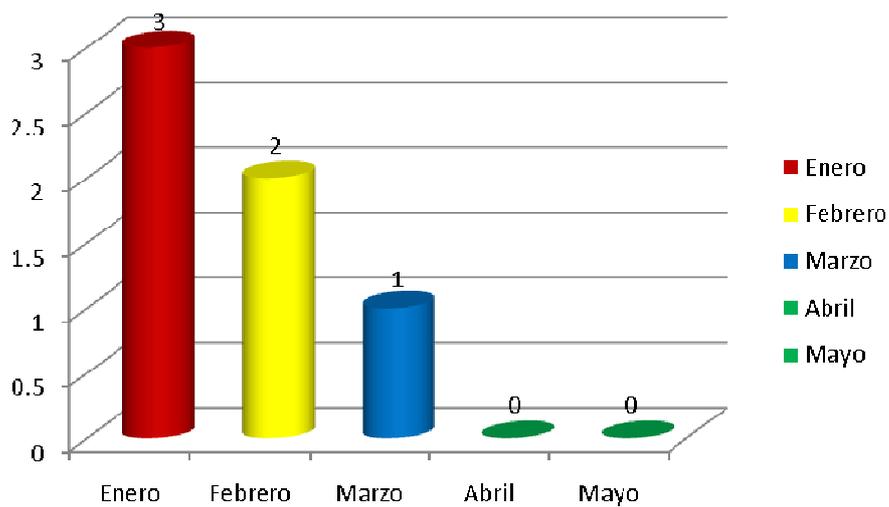


Figura 30. Errores de diseño, creación propia

A lo largo de esta sección hemos visto el impacto directo de las áreas de proceso en los errores más comunes, identificando los beneficios obtenidos e identificando un nuevo ciclo de mejoras, manteniendo el objetivo de producir software de mejora calidad, en búsqueda de la satisfacción del cliente.



3.4 Implementación del modelo.

Implementación de la organización del equipo de proyecto.

La organización del equipo de proyecto, no tiene un impacto directo sobre la disminución de errores presentados durante el ciclo de vida de un proyecto, pero es una actividad primordial que debe de ser definida e institucionalizada, ya que permite mejorar el ambiente laboral, e incrementar el compromiso de cada miembro de la organización al tener claramente definido su papel dentro de la organización, sus responsabilidades y la dependencia con otras áreas.

Se presento la propuesta a la dirección general, a las distintas direcciones, las cuales coincidieron en que las actividades propuestas son entre otras las que desempeñan dentro de sus funciones. Así se delega la responsabilidad a cada director de bajar la información a sus subordinados.

Posterior a esto se aplico el siguiente cuestionario a cada uno de los miembros de la organización.

¿Conoces cuál es tu rol en la organización?

¿Cuáles son las responsabilidades de tu puesto?

¿Qué áreas dependen de tu trabajo?

¿Qué necesitas para desempeñar mejor tu trabajo?

De las preguntas antes expuestas se obtuvo la siguiente información:

- Todos los miembros de la organización tienen claramente definido su rol, sus responsabilidades y quienes dependen de su trabajo.



- La mayoría del considera que se requieren cursos de capacitación del negocio enfocados a la aplicación.
- Los consultores de negocio solicitan apoyo en el desarrollo de requerimientos, los cuales corresponde a actividades muy técnicas.
- Se evidencia falta de organización entre las áreas y los mecanismos de comunicación.

Implementación de la administración de requerimientos.

La implementación de la administración de requerimientos inicia con la generación de la conciencia de la importancia de estos, como elemento medular en el ciclo de vida de un proyecto, identifica que la especificación de requerimientos de software es la columna vertebral del proyecto y que sirve de guía en todas las etapas del desarrollo. Si es ambiguo o incorrecto, existirán errores o retrasos, el tiempo invertido agilizará los demás procesos del proyecto. Se creó la plantilla de solicitud y la de especificación de requerimiento de software.

Los problemas son presentados por los responsables de los requerimientos, consultores de negocio. Las demás áreas involucradas en el ciclo de vida del proyecto, desarrollo, nuevas tecnologías y calidad, enfatizan la importancia de la especificación de requerimientos de software como entrada para el inicio de sus actividades.

Se procedió a modificar las plantillas con documentos más concretos se eliminan los elementos repetitivos y fijos del documento. Con el objetivo de apoyar la identificación de impactos dentro de la aplicación se propuso el desarrollo de un programa que busque de manera automática tanto en el código fuente de los sistemas y en la base de datos, los objetos involucrados en un desarrollo.



Adicional a esto se planteo la generación de una base de datos para la matriz de rastreabilidad que sea alimentada con cada desarrollo, que en un futuro apoyara a determinar los impactos de los nuevos requerimientos de una forma más eficiente.

Con estas modificaciones se han eliminado los errores de análisis al involucrar al cliente y solicitar su aprobación tanto de la solicitud como la especificación de requerimientos y mediante la generación de prototipos el cliente tiene una visión clara del desarrollo antes de que se libere.

La administración de requerimientos ha apoyado en la disminución de errores de desarrollo del tipo de análisis e internos.

Implementación de la planificación.

Durante la implementación de la planificación se han detectado ciertos problemas debido a los cortos tiempos de los proyectos del caso de estudio, por lo que se complica la elaboración de todos los planes para cada uno de los proyectos que se desarrollan. Se ha definido planes fijos. El plan de desarrollo apoya la asignación de recursos de forma eficiente, maximizando el tiempo de ocupación, mediante la generación del cronograma, permite tener claramente la asignación de recursos y disponibilidad.

El plan de comunicaciones interno y sus reportes han sido establecidos una vez a la semana, durante las juntas de dirección, entregando en ella los avances de los proyectos, permite a la dirección tener una visibilidad de los proyectos en desarrollo, las fechas de entrega y los compromisos. Lo que permite vislumbrar fechas posibles en las que se puede adquirir nuevos compromisos. De la misma forma se han establecido juntas semanales con los clientes, para informar mediante los reportes de comunicación externa los avances de los proyectos e identificación de nuevas necesidades.



Durante el plan de pruebas se ha identificado la necesidad de incorporar en los requerimientos, casos de prueba definidos por los expertos del negocio, con el objetivo de enfocar las pruebas y se pruebe lo que se debe de probar. Actualmente pruebas se apoya mucho en desarrollo para solventar deficiencias con respecto al negocio, lo que sesga de cierta forma las pruebas al solo tener la visión del desarrollador, el cual no es experto en el negocio, por lo que se definió que el apoyo a la gente de calidad se brindara por los consultores y no por desarrollo.

Implementación del monitoreo y control.

La implementación del monitoreo y control se ha realizado a través de los casos de uso, ya que las estimaciones de tiempo se definen de acuerdo a estos, el monitoreo y avance del proyecto de la misma forma.

Esta área de proceso ha tenido un gran impacto al determinar desviaciones importantes en los proyectos, a tempranas alturas del proyecto, lo que ha permitido tomar acciones correctivas que permitan entregar a tiempo los proyectos. Lo que ha obligado a tener que trabajar fuera del horario normal, lo que representa un costo adicional para la empresa y afectaciones en el equipo de trabajo.

Pero ha evidenciado que los proyectos siguen con retrasos, debido a deficiencias en la rastreabilidad de los impactos de los proyectos en el desarrollo.



Implementación de medición y análisis.

La ejecución de las métricas definidas se encuentra estrechamente relacionada al monitoreo y control ya que al realizar esto último se genera la información necesaria para las métricas.

Dentro del caso de estudio se implementó una herramienta que apoya la gestión en cada una de las etapas del ciclo de vida del proyecto, de esta forma es posible tener disponible la información necesaria para las métricas, ya que dicha herramienta almacena la información de la calendarización, así como la asignación de actividades, las fechas de inicio de actividades y de cierre de las mismas, además permite comparar las fechas planeadas contra las fechas reales.

Existen varias herramientas que apoyan la gestión de proyectos de desarrollo se recomienda ampliamente tener una, que permita almacenar información histórica de los proyectos.

Implementación de la administración de la configuración.

Para la implementación de esta área de proceso se decidió utilizar una herramienta que apoyara en esta labor, la herramienta seleccionada fue la que se propuso en el presente trabajo, Microsoft Visual Source Safe, ya que se integra directamente sobre los entornos de desarrollo que se utilizan y adicionalmente permite administrar proyectos de otras tecnologías al proteger archivos.

Se definió un responsable de investigar el funcionamiento de la herramienta para ser implementada, la investigación y prueba en un proyecto independiente llevo 15 días.



La integración en los entornos de desarrollo y la facilidad de uso permitió utilizar la herramienta una semana después de que las pruebas fueron exitosas, lo más complicado consistió en ingresar los proyectos a la herramienta. Permite incorporar rápidamente a todo el personal en la herramienta.

Tiene un impacto muy importante al disminuir casi el total de errores presentados por problemas de versiones, facilitando la generación de líneas base y auditorías de configuración.

Se está trabajando en actividades de mantenimiento y respaldo sobre la herramienta para un mejor funcionamiento de la misma.

3.5 Satisfacción del cliente.

El objetivo final de la generación de software con una mejor calidad es la búsqueda de la satisfacción total del cliente, lo que significa que el producto sea desarrollado y entregado en tiempo, en los costos definidos que cubra sus necesidades en funcionalidad, uso, desempeño y que apoye la rentabilidad de la empresa, otorgándole ventajas competitivas en el mercado que se desenvuelva.

Por lo que el punto final en el presente trabajo corresponde a la evaluación de la satisfacción del cliente con respecto a la calidad de los productos entregados por la empresa FINASIST, antes, durante y después de la implementación del modelo de madurez. Por lo que se elaboró el siguiente cuestionario, dirigido a los líderes de negocio responsables de la operación del producto de FINASIST, donde se evalúan los siguientes criterios: Perspectiva del proveedor de incrementar los ingresos, resolver problemas empresariales, cumplir con los compromisos en cuanto a plazos y presupuestos, contar con flexibilidad y capacidad de respuesta ante las necesidades, expectativas de calidad.



¿De manera general, indique su satisfacción con relación al proveedor FINASIST?

100% 75% 50% 25% 0%

¿Los proyectos desarrollados cubren sus necesidades?

100% 75% 50% 25% 0%

¿Qué porcentaje de proyectos se entregan en tiempo?

100% 75% 50% 25% 0%

¿Qué porcentaje de proyectos se entregan con errores?

100% 75% 50% 25% 0%

¿Los proyectos entregados afectan otros procesos o funcionalidad existente?

100% 75% 50% 25% 0%

¿Los nuevos desarrollos son fáciles de usar?

100% 75% 50% 25% 0%

¿Los desarrollos apoyan la eficiencia de su operación?

100% 75% 50% 25% 0%

¿Los desarrollos apoyan la rentabilidad de la empresa?

100% 75% 50% 25% 0%



Los datos a continuación presentados son simulados, debido a que la empresa considera que no es momento de realizar este tipo de encuestas entre los clientes. Podemos ver que la satisfacción del cliente se incrementa, se espera que para el mes de octubre los requerimientos desarrollados cubran al cien por ciento las necesidades de los clientes, las entregas a tiempo lleguen a un porcentaje del 95% de entregas en tiempo, el margen del 5% se debe a situaciones ajenas al desarrollo, cambio de prioridades urgentes, el porcentaje de errores se reduzca hasta un 10% causado principalmente por los errores no detectados y el apoyo en la eficiencia del cliente y en su rentabilidad por los proyectos sea muy alta.

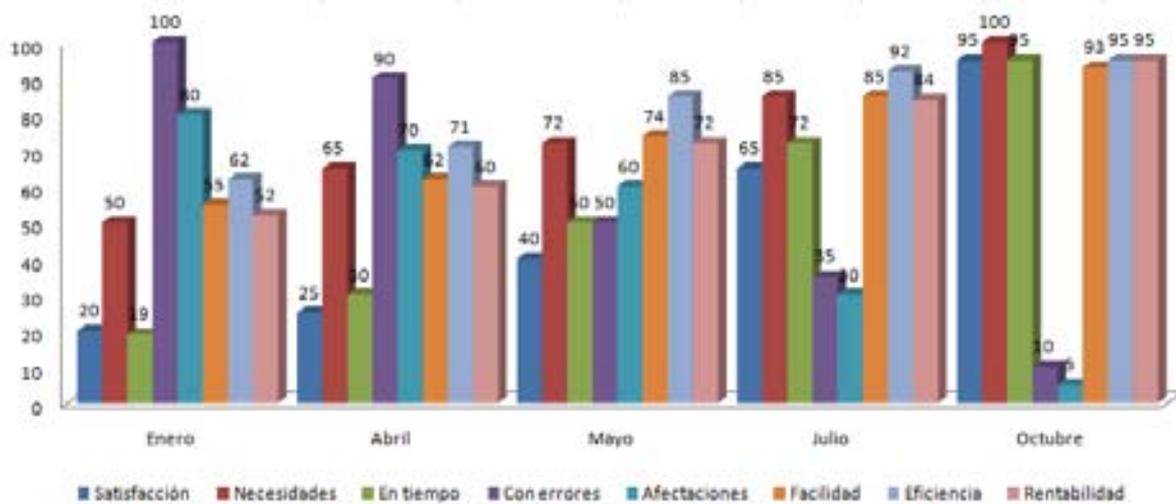


Figura 31. Satisfacción del cliente creación propia

La situación presentada puede llegar a alcanzarse, mediante la corrección de las deficiencias encontradas, principalmente trabajando en el punto de matriz de rastreabilidad y la automatización del proceso de pruebas.



Conclusiones.

El proceso de evaluación y acreditación del modelo de madurez de capacidad integrada, se realiza a través de la medición del apego al proceso por lo que en la sección 3.1 se establecen una propuesta de evaluación para cada una de las áreas de proceso, a lo largo de cinco meses de implementación observamos su evolución hasta estar completamente implementadas sus prácticas y actividades dentro del caso práctico.

La implementación, concientización y avance dentro de cada una de las áreas de proceso demostró que la calidad de los productos de desarrollo, está relacionada directamente con la calidad de los procesos utilizados para realizarla, ya que conforme se ejecuta cada una de las áreas y actividades, la calidad del software desarrollado se incrementa. De la misma forma se pudo identificar que la falla en la implementación de una práctica o la falta de entendimiento de alguna actividad, la cual se ejecute solo por cumplir con el proceso, afecta considerablemente el impacto en la mejora, desmotivando al equipo, generando un sentimiento de burocratización y rechazo, por lo que se debe de entender claramente, el objetivo de cada área de proceso, de sus actividades y prácticas, dejando claramente definido el responsable, la forma de ejecutarse, los involucrados, la forma de medición y la presentación de resultados.

Lo antes expuesto queda demostrado al entregar desarrollos que cubren las necesidades y requerimientos del cliente, ya que estos interactúan continuamente en el proceso de desarrollo, aprobando los requerimientos y conociendo el producto antes de su liberación, así como en la disminución de los errores y las entregas a tiempo de los proyectos, lo cual aparte de mejorar la calidad del desarrollo de software, fomenta la satisfacción del cliente e incrementa la rentabilidad de la empresa.



Conclusiones finales.

El presente trabajo ha proporcionado una guía para la administración de proyectos de software, mediante el análisis de las áreas de proceso definidas en el nivel dos del modelo de madurez de capacidad integrada, donde sus prácticas y metas genéricas proporcionaron el marco general de trabajo, y su ejecución se desarrolla a través de la metodología de desarrollo de Microsoft.

Después de la implementación de la guía dentro del caso práctico FINASIST se obtuvieron los siguientes resultados:

Antes de iniciar la implementación del modelo no se elaboraba un documento formal de especificación de requerimientos de software; después de la implementación de la guía el 100 % de los proyectos cuenta con una especificación de requerimientos de software de negocio, que describe claramente las funcionalidades solicitadas y el detalle de las mismas, de estas el 83.3% se presentan al cliente, como se muestra en la figura 16 de la sección 3.1, el involucrar al cliente desde el inicio y solicitar su aprobación desde la solicitud y posteriormente cuando se convierte en requerimiento, ha conseguido eliminar casi en su totalidad los errores de análisis y de diseño como se observa en la sección 3.2.3.

Otros de los errores dentro de los cuales la propuesta tuvo un impacto directo corresponden a los errores por control de versiones que disminuyeron de 15 en enero a 2 en mayo, como se puede ver en la sección 3.1.4, esto gracias a la implementación de la herramienta para el control de versiones que fue sugerida dentro de la guía.



El modelo permitió pasar de un 37.5 % a un 88.8 % de entregas a tiempo de los proyectos, si bien continúan los retrasos y errores en los proyectos, esto se debe a la falta de la identificación oportuna de todos los impactos en la aplicación, pero gracias al monitoreo se ha podido tomar acciones que permita incrementar las entregas a tiempo. Como medida tomada para solventar el problema de impactos se propone la generación de una herramienta que permita identificar de forma más exacta y eficiente las afectaciones.

De acuerdo al análisis realizado en los primeros cinco meses del año 2007 se habían presentado alrededor de 158 errores, para los primeros cinco meses del año 2008, solo se han presentado 87 errores en los proyectos de desarrollo de software de los requerimientos elaborados, lo que significa:

Disminución del 55% en errores en los mismos errores.

Si se tomas en cuenta que en promedio corregir un error tarda 24 Hrs.

Año	Errores	Tiempo Total
2007	158	3792
2008	87	2088

Se observa que en los primeros cinco meses del año 2007 se habían dedicado un total de 3792 horas a corregir errores, aproximadamente 474 días, equivalente a que una persona labore por más de un año, única y exclusivamente en la corrección de errores.

La disminución en los errores implica no retrasar otros proyectos, al no tener que asignar tantos recursos y por tanto tiempo a la corrección de errores.



Si lo antes mencionado se expresa en términos de costos se obtienen los siguientes datos.

Tiempo En Hrs.	Recurso responsable	Actividad	Costo Por Hora	Costo Total
3	Atención a clientes	Replicar error y documentar.	37.50	112.50
2	Director de desarrollo	Asignación de recurso adecuado.	185.00	370.00
9	Desarrollador	Análisis, desarrollo y pruebas unitarias en versión de producción y versiones necesarias.	62.50	562.50
8	Probador	Pruebas integrales y de liberación.	41.60	332.80
2	Configurador	Paquete de liberación, e integración de versiones.	50.00	100.00

Total: 1,477.80

Periodo	Costo por error	Total de errores	Total en pesos
2007	1,477.80	158	233,492.40
2008	1,477.80	87	128,568.60

Lo que representa una disminución en pérdidas de 104,923.80 para los primeros cinco meses del año.

Observamos que existe una relación directamente proporcional entre la implementación de las practicas y actividades de cada área con la disminución de los errores, por lo que podemos concluir que la guía para la administración de proyectos de software permite generar software con una mejor calidad.



La medición y análisis proporciono información sobre las principales causas de retraso como se muestra en la figura 25 de la página 103, esto junto con la clasificación de errores y la evaluación de los tipos de errores presentada en la sección 3.2 permite tener información que apoye la toma de decisiones, al saber de manera clara las áreas de oportunidades, lo que permite generar un nuevo ciclo de mejoras, que puede continuarse sucesivamente.

Gracias a la mejora continua de los procesos se puede visualizar para el 2010 como escenario factible y viable, el denominado los campos elíseos en el cual, el sector financiero se encuentra estable, en crecimiento, la competencia en el sector financiero es agresiva, lo que implica mayores solicitudes de requerimientos de desarrollo para apoyar a los clientes en sus operaciones, donde FINASIST es capaz de generar software con una mejor calidad, consiguiendo de esta forma la satisfacción de sus clientes. Se puede profundizar en este tema en el anexo de planeación de escenarios futuros.

El presente trabajo demostró la importancia de la mejora de procesos, marcando las directrices a seguir en el proceso de administración de proyectos de software. Es importante identificar que esto solo es el principio hacia la mejora continua, queda abierto al lector el continuar avanzando hacia los siguientes niveles del modelo de madurez.

Un elemento vital durante el desarrollo del presente trabajo fue la planeación ya que proporciona una visión particular para hacer frente a un problema que se presenta. La planeación establece las bases de cómo definir las acciones de hoy, en la búsqueda de un futuro deseado, centrado en la mejora continua al igual que el modelo de madurez.



Anexo planeación de escenarios futuros.

El caso de estudio del presente trabajo corresponde a la empresa FINASIST, una empresa dedicada al desarrollo de software, no como fábrica de software si no propietario de un producto. Es un proveedor de software de crédito para el sector financiero y organizaciones auxiliares del crédito (Arrendadoras, SOFOLES, SOFOMES, Uniones de crédito). FINASIST ofrece una solución integral para la administración del crédito, la cual integra todos los procesos administrativos y operativos que permiten a sus clientes administrar eficientemente la cartera de crédito.

Una de las dos variables principales a ser consideradas dentro de la planeación de escenarios futuros es el sistema al cual se encuentra dirigido el software desarrollado por FINASIST, por lo que a continuación se presenta la evolución del sector de Sociedades Financieras de Objeto Limitado (SOFOL) y Sociedades Financieras de Objeto Múltiple (SOFOM).

Un evento importante a tomar en cuenta es el “marco jurídico y normativo en el cual destaca el cambio que entro en vigor el 18 de julio del 2007 y que da origen a la figura de Sociedad Financiera de Objeto Múltiple”¹. Esto permite a una sola institución ofrecer créditos y arrendamientos, anteriormente se tenían que conformar dos organismos dentro de un grupo, uno para el crédito y otro para el arrendamiento, lo que significaba dos diferentes sistemas. El producto desarrollado fue diseñado con la visión de dicha entidad, lo que permite administrar en una sola herramienta créditos y arrendamientos, lo que ofrece una ventaja competitiva ante la competencia.

1 Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág. 6



Por las características propias del sector existe una incidencia elevada de aspectos regulatorios que inciden sobre su costo de operación, se considera que la mayor barrera de entrada lo constituye el monto de la inversión requerida (Capital, Activos fijos, Personal calificado, red de distribución), para iniciar operaciones. En el caso de SOFOLES existe una restricción adicional que incide sobre su crecimiento y que está dada por la imposibilidad de captar recursos del público.

“Al cierre del cuarto trimestre de 2007, el sector de SOFOLES/SOFOMES se integró por 63 entidades afiliadas a la AMFE (Asociación Mexicana de Entidades Financieras), 31 SOFOLES, 25 SOFOMES (antes SOFOLES), y 7 nuevas SOFOLES, existen 434 SOFOMES registradas en operación, de las cuales 32 están afiliadas a la AMFE”².

Las SOFOLES/SOFOMES hipotecarias y automotrices son las de mayor importancia del sector. En diciembre de 2007, la participación de estas empresas en los activos totales ascendió a 78.7%.

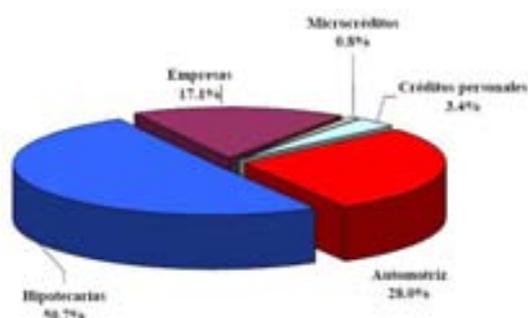


Figura 32. “Porcentaje de participación en los activos totales del sector”³.

² Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág. 6

³ Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág.8



La mayor parte de los clientes de FINASIST se encuentran en el sector automotriz, de la información antes presentada se puede observar que un sector que debe ser tomado en cuenta es el Hipotecario.

“Las instituciones financieras presentan una alta dependencia en aspectos relacionados con la tecnología de la información. El sector se encuentra actualmente dentro de una etapa estable/madura”⁴. El dinamismo de la actividad depende de factores primordialmente vinculados con la evolución económica. Salvo el caso de uniones de crédito donde existe una tasa de fracaso elevada, en el resto de las instituciones se estima que la tasa de fracasos es prácticamente nula. Existe una relación muy estrecha entre la expansión del crédito, la captación y la calidad de la cartera de las instituciones con el crecimiento económico, la inflación, el empleo, las tasas de interés y el tipo de cambio.



Figura 33. “Relación del PIB del sector financiero con respecto al PIB Total, 1994 - 2007”.

4 Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág.9



Durante el cuarto trimestre de 2007 se continuó la recuperación de la actividad económica. En este periodo el PIB registró un crecimiento anual de 3.8% y uno trimestral de 0.74%.

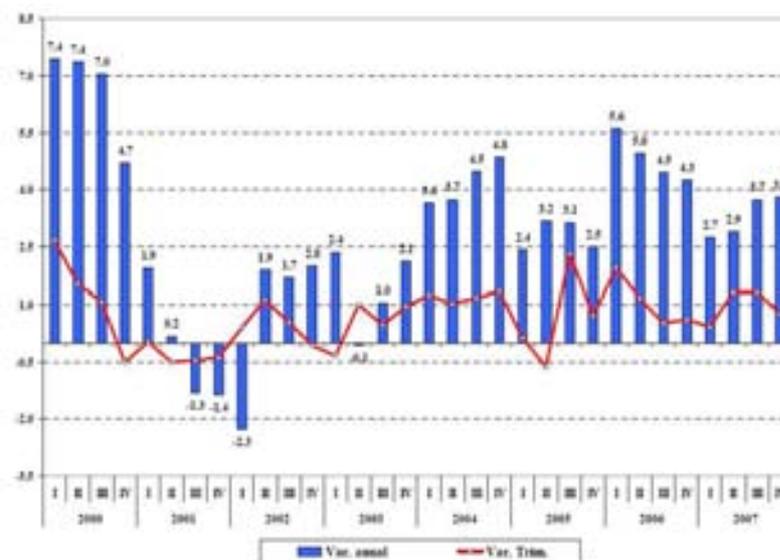


Figura 34. "Producto Interno Bruto trimestral, 2000-2007 (Variación % anual y trimestral)"⁵.

El crecimiento del PIB durante 2007 fue 3.3%; sin embargo, su evolución confirma el inicio de un proceso de desaceleración gradual, en especial, de los sectores estrechamente ligados al crecimiento económico de los EUA.

De manera consistente con la evolución de algunos sectores productivos la generación de empleo en el país durante el cuarto trimestre de 2007 registró una disminución importante respecto a la observada en el mismo periodo de 2006. Aunque la generación de empleos permanentes en 2007 fue mayor que la registrada en 2006 (47.1% contra 45.3 %), la correspondiente a los eventuales fue elevada

⁵ Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág.11



(52.9%). De manera consistente con el crecimiento económico, la generación de empleo formal en la economía se registró, fundamentalmente, en el sector de servicios y la industria de construcción.

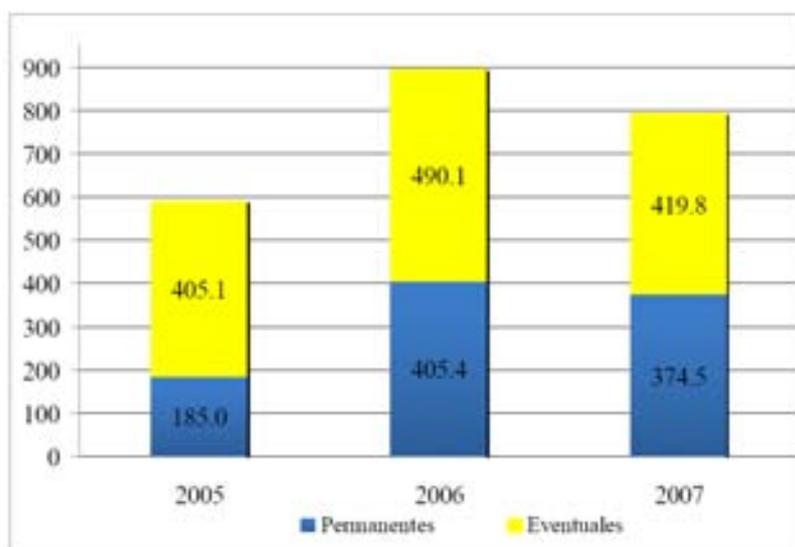


Figura 35. "Asegurados permanentes y eventuales en el IMSS (Miles de personas a diciembre de cada año)"⁶.

Las tasas de interés de los CETES se incrementaron casi medio punto porcentual por encima del nivel registrado al cierre de diciembre de 2007, ubicándose en 7.42%, 7.5%, 7.54% y 7.43%, en los plazos de 28, 91, 182 y 360 días (respectivamente).

El tipo de cambio (fix) se ubico en \$10.70 y la cotización del futuro del peso, para marzo del 2008, en \$11.05, lo que implica que el peso se ha apreciado sólo 4.3% y 2.5% (respectivamente) respecto al nivel registrado en septiembre de 2007.

⁶ Grupo de economistas y asociados, Evolución del sector SOFOL/SOFOM, México, Asociación Mexicana de Entidades Financieras, 2008, pág.15



Al cierre de febrero de 2008, el índice de riesgo país ascendió a 161 puntos, 38 puntos por encima del nivel registrado en la misma fecha de 2007.

El grado de competencia en el sector financiero es agresivo, la competencia estriba en su habilidad para atender a los segmentos aún no atendidos por el sistema bancario tradicional. La rentabilidad del sector es estable, las tasas de incumplimiento del sector es muy baja. Su éxito depende fundamentalmente de su habilidad de adoptar la tecnología que les permita avanzar en el conocimiento de su clientela objetivo.

La sensibilidad a la inflación es alta, al tipo de cambio y a la tasa de interés es elevada, los riesgos más importantes que enfrenta el sector son que se registre un deterioro de la cartera de crédito por una desaceleración de la actividad económica y el empleo o bien por un incremento importante en las tasas de interés y el tipo de cambio. No obstante se considera que este riesgo es poco factible de registrarse en el corto o mediano plazo.

Las perspectivas de crecimiento del sector para 2008 continúan positivas. Si bien se anticipa una desaceleración del crecimiento económico de México, las perspectivas sobre el dinamismo del mercado interno y la generación de empleo en la economía son favorables.

La segunda variables principales a ser consideradas corresponde a la industria de Tecnología de Información y Comunicaciones TIC, en México, básicamente a la sección de desarrollo de software en la categoría de Software Empaquetado y en la personalización de esta aplicación, incorporándolo al caso práctico del presente trabajo. A continuación se presenta información del Área Metropolitana de Monterrey, Nuevo León (AMM) y la Zona Metropolitana del Distrito Federal (ZMDF).



Un elemento básico en este tipo de productos corresponde a la arquitectura del software empaquetado, en el cual se deben de tomar en cuenta los siguientes elementos:

- Lenguajes de programación.
- Plataforma.
- Base de datos.

Lenguajes de programación.

La mayoría de las empresas utilizan principalmente los siguientes lenguajes: Visual Basic, SQL Server, HTML, SQL, Java, XML y Java Script. El lenguaje de programación Sybase tiene el mayor porcentaje de uso con un 79% en ZMDF, lo que contrasta con un 13% de las empresas de AMM.

Lenguajes de programación

	Visual Basic	b/Com	C	C++	Cobol	Xbase	Power builder	SQL Server
ZMDF	67	17	30	38	7	2	17	56
AMM	70	26	23	32	15	11	17	70

	Servd WWW	Delphi	HTML	XML	Java	Java Script	Active x	Latte	Visual Café
ZMDF	43	23	65	53	60	54	27	0	7
AMM	49	19	68	51	45	40	38	0	4

	SQL	RPG	Clipper	Progress	Oracle	Basic	Fox Pro	Dbase
ZMDF	68	10	5	14	44	9	11	5
AMM	55	19	13	19	32	17	30	17

	Sybase	Infor mix	DB2	ERP	Macro media	Firm ware	Otro	Macro Excel	Smallt aik
ZMDF	79	19	25	23	33	5	16	25	1
AMM	13	17	21	26	26	11	21	34	9

Figura 36. "Principales lenguajes de programación"⁷

⁷ Dirección General de comercio interior y economía digital, Estudio de nivel de madurez y capacidad de procesos de la industria TI, México, Secretaría de Economía, 2004, pág.22



Plataformas.

Los resultados correspondientes a las plataformas más utilizadas, muestran que las principales son Windows, Unix y Linux.

Plataformas

	OS/2	Windows	Windows CE	Palm OS	Unix	Unix- base OS	Linux
ZMDF	10	81	33	26	49	9	46
AMM	11	83	40	23	40	13	45

	Apple Mac OS X 10.1.5	Apple Mac OS X 9.2.2	Sistema Mini computer	Sistema Mainframe	Sistema Operat. (RTOS)	Otra
ZMDF	4	2	11	10	5	4
AMM	6	4	26	11	9	4

Figura 37. "Principales plataformas"⁸

Base de datos.

Los resultados correspondientes a las bases de datos más utilizadas son MS SQL Server, Oracle, Access y MySQL.

Bases de datos

	Sybase	Oracle	Informix	DB2	MySQL	Dbase	VMS	MS SQL Server	SQL Base	Ingres
ZMDF	33	57	32	40	38	15	2	69	14	1
AMM	19	43	21	26	36	19	11	77	19	4

	Progress	ODBC	JDBC	Pervasive	Clipper	Excel	Acces	Mopix	Cobol	Otro
ZMDF	16	40	31	5	5	32	47	0	6	11
AMM	19	49	26	11	13	32	53	4	17	9

Figura 38. "Principales plataformas"⁹

⁸ Dirección General de comercio interior y economía digital, Estudio de nivel de madurez y capacidad de procesos de la industria TI, México, Secretaría de Economía, 2004, pág.23

⁹ IDEM



“Las exportaciones exclusivamente de software muestran que 15 empresas realizan dicha actividad, esta cifra representa sólo el 12% del total de empresas encuestadas. Además es importante señalar que 8 empresas se ubican en la ZMDF y 7 en el AMM.

Exportaciones de software

Tamaño	Exportación de ventas totales (millones de pesos)						Total
	Menos 2.5	2.5 - 7	7 - 15	15 - 30	30 - 50	50 - 150	
Micro	2	2	0	0	0	0	4
Pequeña	2	3	2	0	0	0	7
Mediana	0	0	0	0	1	0	2
Grande	0	1	0	0	1	1	2
Total	4	6	2	0	2	1	15

Figura 39. “Exportaciones de software”¹⁰

A nivel internacional existen diversos modelos de mejora de procesos de software, en general todos coinciden en que el objetivo principal consiste en incrementar la productividad de las empresas y de los ingenieros de desarrollo de software a través de prácticas para la mejora continua en calidad de desarrollo de software.

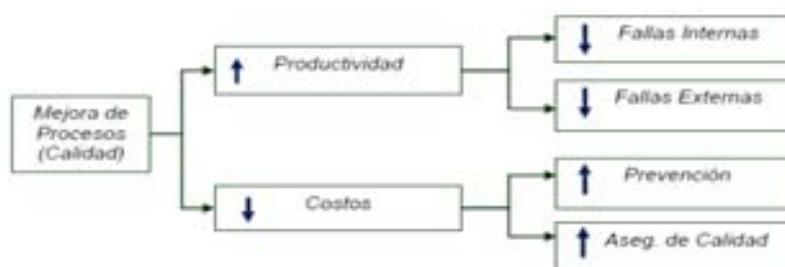


Figura 40. “Impacto en productividad y costos”²¹

¹⁰ Dirección General de comercio interior y economía digital, Estudio de nivel de madurez y capacidad de procesos de la industria TI, México, Secretaría de Economía, 2004, pág.28



La inversión en mejora de procesos de software mide el compromiso de la dirección de la empresa en la implementación de mejoras en sus procesos de desarrollo de software.

	Total	MPYMES ⁷	GRANDES	ZMDF	AMM
1.1. Gasto anual en actividades de calidad y mejora de procesos de software.					
	1,133,837	947,868	2,547,200	1,593,929	275,000
1.2. Porcentaje de ventas invertido en calidad y mejora de procesos de software.					
	11%	12%	8%	13%	8%
1.3. Gasto anual en capacitación y entrenamiento de ingenieros de software.					
	206,845	199,440	299,414	214,359	189,000
1.4. Costo de la capacitación y entrenamiento por ingeniero de software.					
	16,185	17,125	4,902	17,937	11,429
1.5. Razón Costo / Beneficio de mejora de procesos de software.					
	1.3	1.3	1.1	1.4	1.5

Figura 41. Inversión en mejora de procesos.

El gasto anual promedio en actividades de calidad y mejora de procesos de desarrollo de software es alrededor de 1.1 millones de pesos por empresa, destacando que las empresas grandes gastan el doble de promedio.



La productividad es un concepto de eficiencia en el uso de los recursos en términos físicos, consiste en hacer más y mejor con los mismos recursos, miden los efectos de los recursos en los resultados.

	TOTAL	MPYMES	GRANDES	ZMDF	AMM
2.1. Incremento en Líneas de Código (LOC) por unidad de tiempo.	25%	26%	18%	24%	28%
2.2. Reducción de defectos en etapas previas a la liberación del producto.	32%	30%	49%	33%	31%
2.3. Reducción de tiempos para pruebas.	24%	24%	39%	27%	24%
2.4. Reducción de tiempos para entrega de producto (desarrollo del software).	29%	27%	46%	30%	29%

Figura 42. "Indicadores de productividad"¹¹

En términos generales estos indicadores miden la eficiencia de los recursos humanos y del modelo de organización de la empresa, observándose resultados favorables superiores al 25% en todos los indicadores.

¹¹ Dirección General de comercio interior y economía digital, Estudio de nivel de madurez y capacidad de procesos de la industria TI, México, Secretaría de Economía, 2004, pág.28



La calidad se basa principalmente en la plena satisfacción del cliente y consiste en la pertinencia en que recibe el producto o servicio y se mide a través del Coeficiente de Satisfacción el cual se obtiene por medio de encuestas aplicadas al usuario de los servicios.

	TOTAL	MPYMES	GRANDES	ZMDF	AMM
3.1. Reducción de reportes de defectos post-liberación del SW.	30%	28%	45%	30%	32%
3.2. Satisfacción del cliente.	9	9	9.5	9.3	9.4

Figura 43. "Indicadores de calidad"¹²

Los resultados son superiores al 25% en la reducción de reportes de defectos post-liberación del SW y calificación de 9.0 en la satisfacción del cliente.

A continuación se presentan cifras estadísticas de las "evaluaciones del modelo CMM reportadas por el Software Engineering Institute (SEI) del Carnegie Mellon University"¹³

El reporte del SEI se realizó sobre la base de 1,345 empresas evaluadas, de las cuales el 47% no son empresas de EUA.

¹² Dirección General de comercio interior y economía digital, Estudio de nivel de madurez y capacidad de procesos de la industria TI, México, Secretaría de Economía, 2004, pág.35

¹³ Process Maturity Profile Of The Software Community 2002 Year End Update. SEMA 3.0.1. SEI: Carnegie Mellon University. April 2003



Tamaño de las empresas	Porcentaje
Menos de 50	28.4
51 a 100	22.8
101 a 200	22.8
201 a 2000	26.1

Nivel	Porcentaje
CMM 1	16.9
CMM 2	43.2
CMM 3	24.6
CMM 4	8.0
CMM 5	7.3

Nivel	Periodo
CMM 1 a 2	22 meses
CMM 2 a 3	21 meses
CMM 3 a 4	25 meses
CMM 4 a 5	15 meses

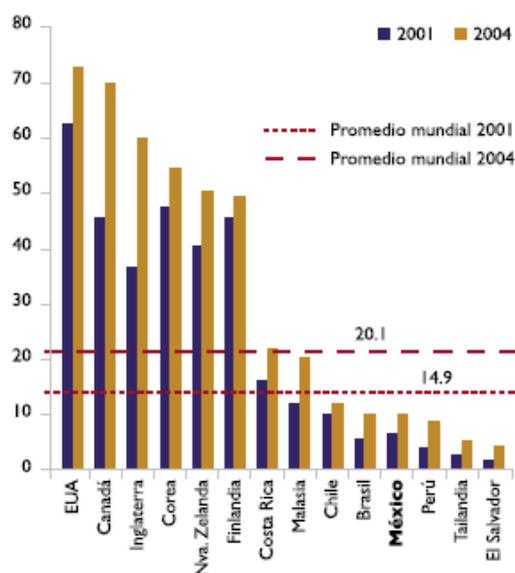
Figura 44. "Evaluación del modelo SEI"¹⁴

México es un país de profundas diferencias en la adopción de Tecnologías de información y Comunicaciones. No solo la penetración y apropiación de las TIC es baja, sino que hay una grave desigualdad en la adopción de dichas tecnologías en las diferentes zonas del país y entre los sectores económicos. El país se encuentra por debajo del promedio mundial en la penetración de computadoras, Internet, líneas telefónicas, teléfonos celulares y suscriptores de banda ancha, como muestran las siguientes graficas.

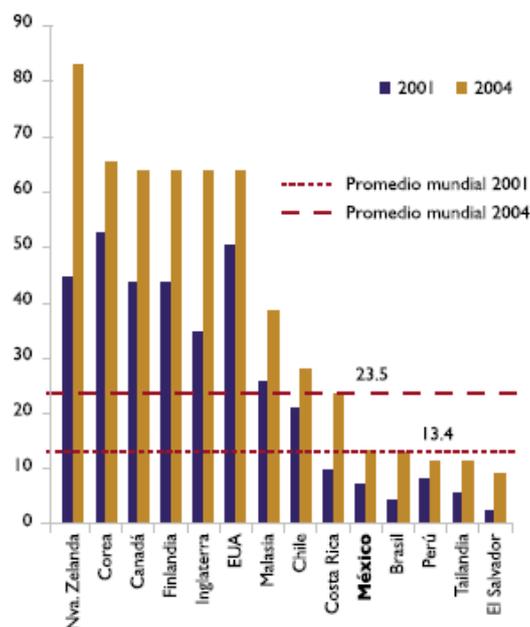
¹⁴ Process Maturity Profile Of The Software Community 2002 Year End Update. SEMA 3.0.1. SEI: Carnegie Mellon University. April 2003



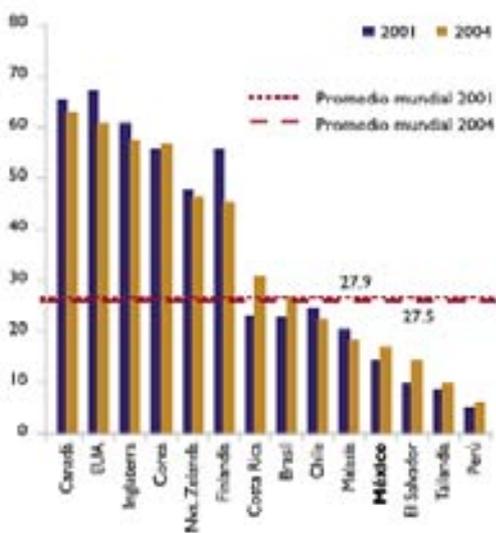
PCs por cada 100 habitantes



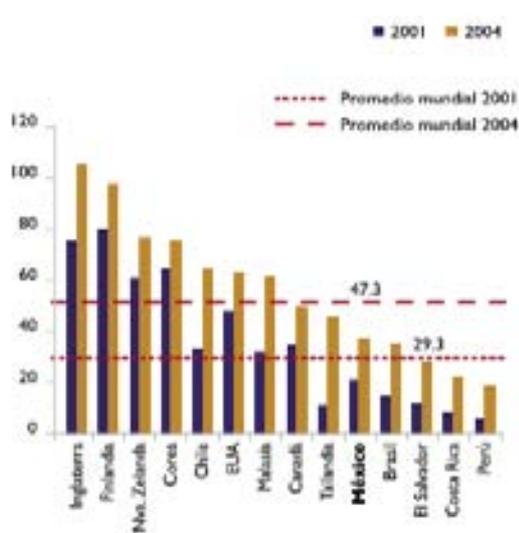
Usuarios de internet por cada 100 habitantes



Lineas telefónicas por cada 100 habitantes



Abonados de teléfonos celulares por cada 100 habitantes





Suscriptores de banda ancha por 100 habitantes, por tecnología, junio 2005

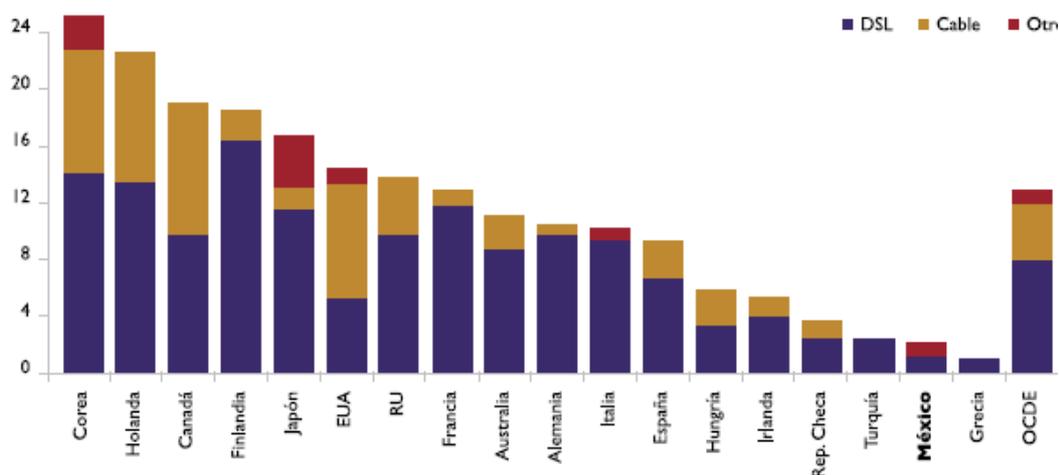


Figura 45. "Estadísticas de la penetración del sector TIC"¹⁵

Históricamente, México ha invertido un bajo porcentaje de su PIB en tecnología comparado con muchos países. No solo en el mercado interno de TIC es pequeño comparado al promedio mundial, sino incluso frente a otras economías emergentes, como la brasileña y la china. De hecho, la penetración de banda ancha en las empresas mexicanas es baja comparada con el promedio de Latinoamérica. El porcentaje de trabajadores con acceso a computadoras y a Internet refleja una situación parecida. Si bien en México el uso de internet es bajo, un elemento competitivo para el producto de Software de FINASIST, es que la aplicación funcione en un ambiente Web.

15 Visión México 2020 Políticas públicas en Materia de Tecnologías de Información y Comunicaciones para impulsar la competitividad de México, 2006, AMITI CANIETI, FMD, Pág. 20



Presupuesto de TIC por empleado, 2005 (USD)



Acceso a PCs e Internet por empleado (por ciento)

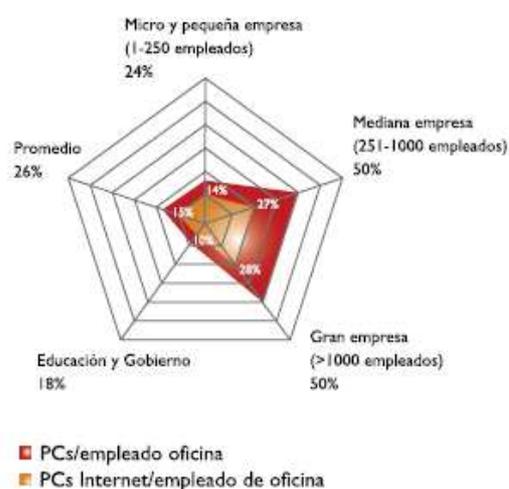


Figura 46 "Penetración por TIC por tamaño de empresa, 2005"¹⁶

Por último una parte importante dentro de la industria de tecnología corresponde a las innovaciones o nuevas tendencias, de ahí la importancia de la interoperabilidad entre sistemas, es decir la capacidad de intercomunicar sistemas en distintas plataformas, y las nuevas tendencias en paradigmas de programación; orientada a servicios, que permite explotar funcionalidades del sistema, por sistemas externos o independientes. Dentro de los factores antes mencionados se encuentra la factura electrónica y el factoraje, que si bien son aspectos de la industria a la cual está orientado el sistema, tienen una relación muy estrecha con el ambiente tecnológico y pueden presentar en un futuro a corto plazo, ventajas competitivas importantes.

16 Visión México 2020 Políticas públicas en Materia de Tecnologías de Información y Comunicaciones para impulsar la competitividad de México, 2006, AMITI CANIETI, FMD, Pág. 22



Variables Principales y secundarias.



Escenario al 2010.

Los campos elíseos (El más optimista).

El sector financiero se encuentra estable, en crecimiento, se consolidan más SOFOLES, SOFOMES, Arrendadoras, lo que implica una mayor demanda del sistema desarrollado, el sector automotriz sigue en auge. La situación económica del país es favorable, se genera un mayor número de empleos, bien remunerados, el poder adquisitivo de la gente es elevado lo que promueve los créditos y el sano mantenimiento de la cartera, las tasas de interés se encuentran en niveles bajos, el tipo de cambio estable sin devaluaciones y la competencia en el sector financiero es agresiva, lo que implica solicitudes de requerimientos de desarrollo adicionales para apoyar a los clientes en sus operaciones.



El sistema de FINASIST, se migro a un ambiente Web, en Java o .NET, mantiene la base de datos en SQL Server, con lo que se disminuye el costo de la licencia, esta migración mejora el desempeño de la aplicación, se diseña de forma que pueda ser ejecutado en múltiples plataformas, no existe una dependencia a un navegador (Internet Explorer). Se realizan innovaciones en la estructura de programación, el código es separado en capas (Presentación, Datos, Negocio) lo que facilita el entendimiento del mismo, la administración, modificación y evolución del mismo, se ejecutan alianzas que permitan incorporar nuevas funcionalidades al sistema la digitalización de documentos. Se implementa el nivel dos del CMMI, consigue la certificación del SEI, lo que tiene un impacto directo en la calidad de los desarrollos y del producto, logrando la satisfacción total del cliente, al recibir los desarrollos en tiempo y sin errores.

El rio Aqueronte.

El sector financiero presenta problemas, debido a regulaciones nuevas, impuestos nuevos, la economía en el país se ve afectada por la desaceleración en EUA, los créditos disminuyen, el empleo decae, las tasas de interés se incrementan, lo que causa que la cartera de los créditos se vea afectada, generando cartera vencida, no se crean nuevas entidades financieras, lo que significa menos ventas de licencias, los clientes actuales no solicitan nuevos desarrollos.

La industria de tecnología sigue crece, FINASIST incorpora nueva tecnología, mejoras tecnológicas cambiando su sistema a un ambiente web, interoperable en distintas plataformas, mejora sus procesos de desarrollo, generando software con mayor calidad pero no se cuenta con nuevos clientes y existe una disminución en el trabajo ya que los clientes no solicitan adecuaciones al sistema, lo que tiene un impacto en la empresa que origina estancamiento y reducción de personal en la empresa.



El tártaro.

El sector financiero sigue en crecimiento consolidando más entidades financieras, incrementando los créditos, el sector automotriz se mantiene un porcentaje alto de participación del mercado y las carteras se encuentran sanas. La inflación, el empleo, las tasas de interés, el tipo de cambio se encuentran estables a las situaciones internacionales, la competencia en el sector financiero es fuerte, centrando sus esfuerzos en operar lo más eficientemente posible, por lo que se requiere de soluciones tecnológicas que apoyen la operación y la eficiencia del mismo.

La industria de tecnologías de información no ha tenido en México la penetración esperada, debido principalmente a la falta de políticas públicas, que motiven la inversión en infraestructura y la falta de interés del sector privado. FINASIST mantiene los problemas actuales de errores en sus requerimientos, no realiza innovaciones tecnológicas en su aplicación lo que le resta competitividad y genera rechazos ante los potenciales clientes, debido a que la tecnología empleada para el desarrollo del producto de FINASIST es obsoleta y ya no se le da soporte. Las referencias de los clientes son negativas lo que genera dificultad en la colocación del producto.

El hades.

El sector financiero presenta estancamiento, resultado de problemas en la economía debido a la afectación de los mercados internacionales, lo que detiene su crecimiento, afecta los presupuestos para el área de tecnología, detienen la compra de licencias y el desarrollo de modificaciones, la industria automotriz se estanca.



Por su parte FINASIST, continua con problemas entregando todos los desarrollos con errores y fuera de los plazos, la incorporación de mejora de procesos es un fracaso, que solo genera burocratización y conflictos internos, el sistema continua en un tecnología atrasada por lo que los clientes no lo quieren por no existir ya soporte en esa tecnología, lo que lleva a un despido de personal.

Escenario al 2015.

Los campos elíseos.

El sector financiero sigue su crecimiento, incorporando nuevos nichos de mercado, la economía del país se encuentra estable, generando más empleos permanentes, incrementando el poder adquisitivo del cliente, el gobierno elabora programas de incentivos fiscales para el sector financiero, en la compra de productos de software desarrollados en el país. La cartera de los créditos se encuentra sana, se incrementan los presupuestos para la industria de tecnología, se hace más evidente la importancia de la tecnología en las empresas del sector financiero, se busca una automatización de todos los procesos del sector, se promueve una cultura del no papel y todos los elementos digitalizados, se hace evidente la necesidad de poder laborar desde cualquier punto y en cualquier momento, por lo que la necesidad de sistema Web es más importante, se requiere de sistemas que puedan intercomunicarse con otros, brindando información más importante que apoye la toma de decisiones. La industria de tecnología continúa en crecimiento incorporando nuevos elementos y brindando mayores oportunidades de interconexión entre sistemas, las decisiones de sistemas web que sean capaces de ejecutarse en distintas plataformas brinda ventajas competitivas en el mercado, el uso de computadoras, y de internet de banda ancha en oficinas y casas se incrementa, el uso de dispositivos móviles.



FINASIST genera mejoras en el sistema que le den una mayor flexibilidad a su producto para poder abarcar otro tipo de créditos, en el plazo 2010-2015 el sistema incorpora funcionalidades de administración de créditos hipotecarios, el factoraje y la factura electrónica. Continúa la mejora de procesos, obtiene en el periodo el nivel tres y cuatro del CMMI, lo que permite desarrollar software de mejor calidad, se vuelve más eficiente y productiva a la empresa, obtiene certificaciones internacionales que le brindan a clientes nuevos una mayor seguridad en la selección del producto, se obtienen apoyos creados por parte del gobierno para la industria de tecnología, obtiene recursos adicionales que permiten el crecimiento de la empresa.

El rio Aqueronte.

El sector financiero presenta problemas debido a cuestiones económicas lo que detiene el crecimiento de este sector, lo que ocasiona que las tasas de interés suban, las cuales generan que los créditos no se puedan pagar y las carteras de la industria financiera se encuentran mal, lo que genera una tendencia a la recuperación de la cartera sobre otras cosas.

La industria de la tecnología avanza y evoluciona generando nuevas tendencias y nuevas tecnologías, todo se encuentra en ambientes Web y dispositivos móviles, sistemas interoperables, que puedan ser ejecutados en distintas plataformas, la mejora de procesos y la adquisición de certificaciones internacionales, le permite a FINASIST incursionar en nuevos sectores adicionales a el sector financiero, desarrollando software a la medida, convirtiéndose en una fábrica de software. Los apoyos obtenidos por las políticas públicas por parte del gobierno a la industria de tecnología permiten a FINASIST obtener capacitación y acercamiento a otros mercados.



El tártaro.

El sector financiero se presenta estable, gracias a las políticas públicas, principalmente a la economía, la estabilidad del tipo de cambio permite incorporar carteras en distintas divisas, el sector se vuelve más demandante en cuestiones tecnológicas, solicitando sistemas confiables, eficientes y robustos que le brinden información no solo de cartera, pagos e intereses, si no información de inteligencia de negocios que brinde ventajas competitivas en el sector.

La industria de tecnología avanza, pero no existen incentivos fiscales, ni planes por parte del gobierno para fomentar el crecimiento en México, ni en capacitación, ni en infraestructura. FINASIST no incorporo elementos tecnológicos innovadores en su producto por lo que se vuelve obsoleto, la plataforma sobre la cual se desarrollo, deja de ser comercial, por lo que se pierden clientes ante la competencia y el sistema deja de ser usado en el mercado.

El hades.

El sector financiero presenta problemas, por lo que existe un estancamiento del sector, debido a los problemas de la economía los sectores automotriz e hipotecario dejan de tener crecimiento y presentan perdidas, las altas cargas fiscales, los nuevos impuestos implementados por el gobierno daña el sector reduce las ganancias y los márgenes.

La industria de la tecnología en México se estanca por problemas de infraestructura y de soporte, a causa de falta de políticas públicas que detonaran el desarrollo, se adquiere software de otros países el cual es incorporado en las industrias, debido al bajo costo de los productos y de los desarrollos.



Escenario al 2020.

Los campos elíseos (El más probable).

El sector financiero sigue su crecimiento, incorporando nuevos mercados, nuevos tipos de créditos, se diversifican. El mercado se vuelve muy competitivo debido a las situaciones favorables de los clientes, estos se preocupan más por las condiciones de los créditos que adquieren, por lo que el mercado se especializa más de acuerdo a los clientes a los que se enfoca, gracias a la utilización de sistemas más sofisticados, la capacidad de análisis es eficiente y se pueden otorgar créditos en tiempos muy ágiles, de la misma forma los estudios de crédito son más exactos al contar con mayor información y acceso.

La industria de tecnologías de información se consolida en México, gracias a las políticas públicas, a los programas y planes desarrollados para apoyar a la industria. Donde el principal consumidor de desarrollo de software es EUA. Gracias a la flexibilidad, tecnología y definición de procesos y a los apoyos del gobierno al sector FINASIST, inicia procesos de exportación al extranjero a EUA y Latinoamérica de su producto de desarrollo.

El río Aqueronte.

El sector financiero en México sufre problemas por devaluaciones, inflación, tasas de interés, por lo que se estanca, no tiene crecimiento, lo que implica mantener los sistemas que tienen y no buscar incrementar funcionalidad o comprar nuevos sistemas. Gracias a las políticas públicas y apoyos del gobierno, FINASIST se aventura en la exportación de su producto y al desarrollo de personalizaciones del producto para otros países, lo que permite tener presencia en EUA y en América Latina, se diversifican sus productos e incrementan sus clientes.



El tártaro.

El sector financiero se mantiene estable con un crecimiento acorde al crecimiento del país, debido a la estabilidad del país y a la disminución de riesgo país, se vuelve un punto importante de inversión, por lo que instituciones financieras de otros países buscan tener presencia en México, lo que abre mas las oportunidades de desarrollo de sistemas para estas instituciones.

La industria de tecnologías de información se encuentra totalmente detenida en el país por lo que las instituciones financieras, se ven en la necesidad de comprar software de otros países, adaptando su operación a dichos sistemas o solicitando la personalización a empresas de desarrollo extranjeras.

El hades.

El sector financiero se encuentra en problemas, debido a los problemas económicos del país, lo que obliga a la incorporación de capitales extranjeros, debido al riesgo que representa el país, las condiciones de los créditos son muy elevadas, lo que dificulta el crecimiento en otros sectores, cerrando las posibilidades de crédito orientados a pequeñas y medianas empresas. Adicionalmente tienen un impacto directo en la industria de tecnologías de información, ya que al pertenecer a grandes consorcios internacionales, estos cuentan con sistemas que son implementados en México.

Al existir menos instituciones financieras, y estas al no necesitar software por contar con sistemas trasnacionales, la industria del desarrollo de software se ve excluida del sistema financiero, relegando su operación a sectores minoritarios.



Escenario al 2025.

Los campos elíseos.

Se han generado nuevas entidades financieras, existen bases de datos del gobierno que pueden ser utilizadas para obtener toda la información de los clientes, la cual esta actualizada, existe un intercambio de información entre las instituciones financieras, lo que permite conocer las situaciones financieras de otros bancos en línea. El sector financiero depende en absoluto de la industria de tecnología, al contar no solo con sistemas que administren su cartera, si no con sistemas de inteligencia de negocios los cuales entregan reportes de las tendencias de los sectores de la sociedad, gustos, preferencias, formas de pago. Los créditos se personalizan por cliente de acuerdo al giro particular, las capacidades y formas de pago de cada cliente. Debido a la competencia tan agresiva del mercado, las instituciones financieras son las que buscan a los clientes, ofreciéndoles créditos a través de medios electrónicos e inclusive visitas personales, las pequeñas y medianas empresas reciben todo tipo de préstamos para su operación o capacitación y no existen solo créditos del sector financiero, si no también programas y apoyos por parte del gobierno.

La industria de la tecnología permite la interconexión a sistemas o medios desde cualquier punto y es accesible a todas las personas, toda persona en México cuenta con una computadora en casa con acceso a internet de banda ancha de forma casi gratuita, las personas pueden realizar cualquier trámite a través de medios electrónicos, las firmas electrónicas permiten autenticar a una persona evitando fraudes. A parte de las funciones administrativas y de operación de los sistemas, estos están más enfocados a la inteligencia de negocios, otorgándole a sus usuarios, información relevante de sus clientes y negocios.



Los sistemas son capaces de almacenar millones de terabytes de información y realizar consultas inmediatas de información, todos los servicios y sistemas, son proporcionados a través de servidores dedicados, las cuales almacenan dichos programas y debido a la gran demanda, los costos son muy bajos, lo que significa que un usuario puede conectarse desde cualquier equipo y tener su información disponible y segura. Los lenguajes de programación son de quinta generación en la cual la forma de programación se realiza de forma grafica sin necesidad de tantas líneas de código, la mayor parte de componentes se encuentran desarrollados y solo se requiere incorporarlos, la comunicación entre plataformas es algo transparente para los desarrolladores debido a estándares comunes en todas las plataformas. Los sistemas desarrollados se cobran por uso y no por licencia, si no se cobra una pequeña comisión por su uso y son cargados en automático a sus cuentas de los clientes. La mejora de procesos a llevado a la industria del desarrollo al grado máximo de calidad al permitir generar software libre de errores, ayudándolo a través de robots de pruebas inteligentes, que son capaces de aprender y replicar ciclos de pruebas por si solos.

El rio Aqueronte.

El sector financiero se encuentra en manos de extranjeros, los sistemas son sistemas traídos de otros países, la economía del país presenta una devaluación, los créditos los pocos que tienen las empresas ya sea pequeñas o medianas empresas, son imposibles de mantener, por las elevadas tasas de interés y no existen créditos para las personas físicas. La industria de tecnologías de información, es afectada por la devaluación, pero se mantiene por el mantenimiento a los sistemas existentes, y los desarrollos a la medida o adecuaciones solicitados por los clientes, por lo que la productividad de la industria es el factor básico para las empresas de desarrollo, donde se tiene que hacer más con menos recursos.



El tártaro.

El sector financiero se encuentra estable y madura, la cartera de los clientes está sana, todos los clientes pagan a tiempo sus créditos, gracias a todos los mecanismos tecnológicos que se les puede proporcionar, no se necesita ir a una sucursal ni para tramitar algún crédito, mucho menos para pagarlo, las transacciones son seguras, no existen fraudes, la totalidad de movimientos se realizan de forma electrónica, lo que los vuelve eficientes y seguros. La industria de tecnología de información en México presenta problemas por falta de infraestructura, de personal capacitado, la mayor parte de sistemas son extranjeros. Las empresas de desarrollo de sistemas financieros mexicano, han sido suplantadas por industrias y sistemas extranjeros.

El hades.

El sector financiero presenta grandes problemas, totalmente en manos de extranjeros, las carteras de las industrias financieras presentan demasiada cartera vencida por lo que las reservas destinadas son demasiado altas, las cargas fiscales y la normatividad del gobierno representan un problema para el sector financiero, se viven las tasas de desempleo más altas, se encuentra en un periodo de devaluaciones, donde las tasas de interés son altas y el tipo de cambio se disparo, la competitividad como país se encuentra en los últimos lugares a nivel mundial. La industria de tecnología de información, se encuentra atrasada más de 50 años a comparación de países de primer mundo, la industria de desarrollo de software no tiene presencia en el país, debido a que la mayoría de los desarrollos son solicitados a fabricas de software en otros países, debido al bajo costo y la superioridad tecnológica de la misma, la penetración de las tecnologías de información quedo detenido por falta de inversión en infraestructura y por falta de personal técnico capaz de mantener y administrar las nuevas tendencias de la industria.



Bibliografía.

Teodora Luz Themsel Paniagua
La protección Jurídica de los programas de cómputo.

Administración exitosa de proyectos.
Jack Gido, James P. Clements
Thomson

Administración profesional de proyectos LA GUIA.
Yamal Chamoun
McGraw-Hill

CMMI Guidelines for process Integration and Product Improvement
Mary Beth Chrissis, Mike Konrad, Sandy Shrum.
Addison Wesley

Analyzing Requirements and Defining Microsoft .NET Solution Architectures
Course 2710 MSDN Training

José Alberto Ortiz "Historia de la Administración", [en línea].
27 de Abril del 2001, 30 de Mayo del 2001, 21 de Diciembre del 2007
<http://www.monografias.com/trabajos7/admi/admin.shtml>

Demonstrating the impact and Benefits of CMMI
Dennis R. Goldenson.
SEI.