



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

**VICERRECTORADO DE INVESTIGACIÓN,
INNOVACIÓN Y TRANSFERENCIA DE TECNOLOGÍA**

**MAESTRÍA EN INGENIERÍA DE SOFTWARE
TERCERA PROMOCIÓN**

**TESIS DE TITULACIÓN, PREVIO A LA OBTENCIÓN DEL
TÍTULO DE MAGÍSTER EN INGENIERÍA DE SOFTWARE**

**TEMA: DISEÑO DE UN MODELO DE PLANIFICACIÓN PARA
PROYECTOS DE DESARROLLO DE SOFTWARE EN LA
DIRECCIÓN NACIONAL DE SISTEMAS DE INFORMACIÓN, DE
LA INTENDENCIAS DE SISTEMAS Y TECNOLOGÍAS DE LA
INFORMACIÓN DE LA SUPERINTENDENCIA DE ECONOMÍA
POPULAR Y SOLIDARIA**

**AUTOR: SILVIA CONSUELO ORDÓÑEZ ESCOBAR
DIRECTOR: LUCAS ROGERIO GARCÉS GUAYTA**

LATACUNGA

2016



ESPE
UNIVERSIDAD DE LAS FUERZAS ARMADAS
INNOVACIÓN PARA LA EXCELENCIA

VICERRECTORADO DE INVESTIGACIÓN, INNOVACIÓN Y TRANSFERENCIA DE TECNOLOGÍA

MAESTRÍA EN INGENIERÍA DE SOFTWARE

CERTIFICACIÓN

Certifico que el trabajo de titulación, "*DISEÑO DE UN MODELO DE PLANIFICACIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE EN LA DIRECCIÓN NACIONAL DE SISTEMAS DE INFORMACIÓN DE LA INTENDENCIA DE SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN DE LA SUPERINTENDENCIA DE ECONOMÍA POPULAR Y SOLIDARIA*" realizado por la señorita **SILVIA CONSUELO ORDÓÑEZ ESCOBAR**, ha sido revisado en su totalidad y analizado por el software anti-plagio, el mismo cumple con los requisitos teóricos, científicos, técnicos, metodológicos y legales establecidos por la Universidad de Fuerzas Armadas ESPE, por lo tanto me permito acreditarlo y autorizar a la señorita **SILVIA CONSUELO ORDÓÑEZ ESCOBAR** para que lo sustente públicamente.

Latacunga, 30 de marzo del 2016

Ing. Lucas Rogerio Garcés Guayta
DIRECTOR DE PROYECTO



VICERRECTORADO DE INVESTIGACIÓN, INNOVACIÓN Y TRANSFERENCIA DE TECNOLOGÍA

MAESTRÍA EN INGENIERÍA DE SOFTWARE

AUTORÍA DE RESPONSABILIDAD

Yo, **SILVIA CONSUELO ORDÓÑEZ ESCOBAR**, con cédula de identidad N° 1103951768, declaro que este trabajo de titulación "**DISEÑO DE UN MODELO DE PLANIFICACIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE EN LA DIRECCIÓN NACIONAL DE SISTEMAS DE INFORMACIÓN DE LA INTENDENCIA DE SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN DE LA SUPERINTENDENCIA DE ECONOMÍA POPULAR Y SOLIDARIA**" ha sido desarrollado considerando los métodos de investigación existentes, así como también se ha respetado los derechos intelectuales de terceros considerándose en las citas bibliográficas.

Consecuentemente declaro que este trabajo es de mi autoría, en virtud de ello me declaro responsable del contenido, veracidad y alcance de la investigación mencionada.

Latacunga, 30 de marzo del 2016

Ing. Silvia Consuelo Ordóñez Escobar
C.C: 1103951768



VICERRECTORADO DE INVESTIGACIÓN, INNOVACIÓN Y TRANSFERENCIA DE TECNOLOGÍA

MAESTRÍA EN INGENIERÍA DE SOFTWARE

AUTORIZACIÓN

Yo, **SILVIA CONSUELO ORDÓÑEZ ESCOBAR**, autorizo a la Universidad de las Fuerzas Armadas ESPE publicar en la biblioteca Virtual de la institución el presente trabajo de titulación "**DISEÑO DE UN MODELO DE PLANIFICACIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE EN LA DIRECCIÓN NACIONAL DE SISTEMAS DE INFORMACIÓN DE LA INTENDENCIA DE SISTEMAS Y TECNOLOGÍAS DE LA INFORMACIÓN DE LA SUPERINTENDENCIA DE ECONOMÍA POPULAR Y SOLIDARIA**" cuyo contenido, ideas y criterios son de mi autoría y responsabilidad.

Latacunga, 30 de marzo del 2016

Ing. Silvia Consuelo Ordóñez Escobar
C.C.1103951768

DEDICATORIA

Dedico el presente trabajo de manera especial primeramente a Dios quien me ha dado fortaleza e iluminado para seguir adelante siempre y así poder culminar este ciclo. De igual forma, dedico esta tesis a mis padres pues fueron el principal cimiento para mi formación académica, inculcando en mí los valores de responsabilidad, dedicación y deseos de superación. A mis hermanos y sobrinos, por sus palabras de aliento y compañía que me permitieron confiar firmemente en mis capacidades, a mis amigos y a todas aquellas personas que de alguna u otra manera han contribuido para lograr mis objetivos.

SILVIA

AGRADECIMIENTO

A Dios por todas las bendiciones recibidas en cada paso de este trabajo, que con su infinito amor me dio fuerza y fe alcanzar esta meta. A mis padres por su apoyo incondicional, valores y sacrificio en todo este tiempo, gracias a ustedes he logrado estar aquí y convertirme en lo que soy, han sido el mejor ejemplo de lucha y perseverancia. A mis hermanos y sobrinos por creer en mí en todo momento motivándome a continuar con mis objetivos.

También quiero agradecer de manera especial a mi Director de Tesis por los conocimientos, experiencias compartidas, orientación y paciencia fundamentales para el desarrollo y la finalización de este trabajo.

No ha sido fácil el camino hasta ahora, pero gracias a su apoyo y comprensión todo ha sido más llevadero, por ello hago presente mi gran afecto hacia ustedes.

SILVIA

ÍNDICE DE CONTENIDOS

CERTIFICACIÓN

.....¡Error!
or! Marcador no definido.

AUTORÍA DE RESPONSABILIDAD.....¡Error! Marcador no definido.

AUTORIZACIÓN.....¡
Error! Marcador no definido.

DEDICATORIAv

AGRADECIMIENTOvi

ÍNDICE DE CONTENIDOS..... vii

ÍNDICE DE TABLAS..... xii

ÍNDICE DE GRÁFICOS..... xiii

ÍNDICE DE ANEXOSxiv

RESUMEN

.....xv
i

ABSTRACT

.....xvi
i

CAPÍTULO I

GENERALIDADES 1

1.1 Introducción 1

1.2 Justificación e Importancia 2

1.3 Planteamiento del problema..... 2

1.4	Formulación del problema	3
1.5	Hipótesis	3
1.6	Objetivo general.....	5
1.6.1	Objetivos específicos	5

CAPÍTULO II

MARCO TEÓRICO	6	
2.1	Introducción	6
2.2	Antecedentes Históricos.....	7
2.2.1	Evolución de las metodologías para la planificación de los proyectos de desarrollo de software en el proceso de ingeniería de software en relación al proceso de desarrollo de software	7
2.3	Antecedentes Conceptuales y Referenciales.....	17
2.3.1	Caracterización tecnológica del proceso de ingeniería de software.....	17
2.3.2	Caracterización tecnológica de la planificación de los proyectos de desarrollo de software	23
2.3.3	Estándares para la planificación del desarrollo de software	26
2.3.3.1	Guía de los Fundamentos para la Dirección de Proyectos (PMBOK)	26
2.3.3.2	Directrices para la Dirección y Gestión de Proyectos (Norma ISO 21500).....	29
2.3.3.3	Norma IEEE 1058-1998 para la Planificación de Gestión de Proyectos Software..	31
2.4	Antecedentes contextuales	34
2.4.1	Misión	34
2.4.2	Visión.....	34
2.4.3	Atribuciones.....	34
2.4.4	Organigrama	35

CAPÍTULO III

DISEÑO DEL MODELO DE PLANIFICACIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE EN LA DNSI DE LA ISTI DE LA SEPS.....	42
3.1. Introducción.....	42
3.2. Estudio comparativo de los modelos de planificación para proyectos de desarrollo de software.....	42
3.2.1. Descripción de modelos y estándares	42
3.2.2. Definición de los Parámetros de Comparación.....	46
3.2.2.1. Criterios vinculados a la planificación de proyectos de software:	46
3.2.2.2. Criterios institucionales para la planificación de proyectos.....	48
3.3. Diseño del modelo de planificación para proyectos de desarrollo de software.	52
3.3.1. Gestión de la integración.....	56
3.3.1.1. Desarrollar el Plan para la gestión de un proyecto de ingeniería de software.....	56
3.3.2. Gestión del alcance.	59
3.3.2.1. Planificar la Gestión del Alcance.....	60
3.3.3. Gestión del cronograma	63
3.3.3.1. Planificar la gestión del cronograma.....	63
3.3.3.2. Registrar y Secuenciar las actividades	65
3.3.3.3. Estimar los recursos y desarrollar el cronograma	68
3.3.4. Gestión de los recursos humanos.	71
3.3.4.1. Planificar la gestión de los recursos humanos.....	71
3.3.5. Gestión de la comunicación	73
3.3.5.1. Planificar la gestión de las comunicaciones.....	74
3.3.6. Gestión del riesgo.....	75
3.3.6.1. Planificar la gestión del riesgo	75

CAPÍTULO IV

APLICACIÓN DEL MODELO DE PLANIFICACIÓN EN UN PROYECTO INSTITUCIONAL.	77
4.1 Introducción	77
4.2. Implementación del modelo de planificación para proyectos de desarrollo de software en la DNSI de la SEPS: Sistema Acopio de Balances.	77
4.2.1. Gestión de la integración.....	78
4.2.1.1. Desarrollar el Plan para la gestión del proyecto de ingeniería de software.	78
4.2.2. Gestión del alcance.	78
4.2.2.1. Planificar la Gestión del Alcance.....	78
4.2.3. Gestión del cronograma	79
4.2.3.1. Planificar la gestión del cronograma.....	79
4.2.3.2. Registrar y Secuenciar las actividades	80
4.2.3.3. Estimar los recursos y desarrollar el cronograma	80
4.2.4. Gestión de los recursos humanos.	81
4.2.4.1. Planificar la gestión de los recursos humanos.....	81
4.2.5. Gestión de la comunicación	82
4.2.5.1. Planificar la gestión de las comunicaciones.....	82
4.2.6. Gestión del riesgo.....	82
4.2.6.1. Planificar la gestión del riesgo	82
CAPÍTULO V	
PRESENTACIÓN DE RESULTADOS	84
1.1 Introducción	84
1.2 Procesamiento de los resultados	84
1.2.1 Análisis y resultados de la aplicación del modelo a los desarrolladores:.....	86
1.2.2 Análisis y resultados de la aplicación del modelo al arquitecto de software:	87

1.2.3	Análisis y resultados de la aplicación del modelo al líder tecnológico:.....	88
1.3	Prueba de hipótesis con chi cuadrado	90
1.3.1	Planteamiento de la hipótesis	90
1.3.2	Cálculo de frecuencias esperadas, correspondientes a cada frecuencia observada. .	90
1.3.3	Cálculo del valor de chi cuadrado.....	93
1.3.4	Cálculo del valor crítico de chi cuadrado.....	94
1.3.5	Comparación entre el valor esperado y el valor crítico.....	95
1.3.6	Conclusiones de los resultados presentados.....	96

CAPÍTULO VI

	CONCLUSIONES Y RECOMENDACIONES.....	98
6.1	Conclusiones	98
6.2	Recomendaciones	100

ÍNDICE DE TABLAS

Tabla 1. Modelos de ciclo de vida de desarrollo de software	20
Tabla 2. Diferencias entre metodologías ágiles y no ágiles	22
Tabla 3. Puntuación de los modelos y normas en base a los criterios establecidos	49
Tabla 4. Resultados comparativos	51
Tabla 5. Detalle del Modelo de Planificación de Proyectos de Desarrollo de Software (MPPDS)	55
Tabla 6. Resultados de aceptación de los desarrolladores.	86
Tabla 7. Resultados de aceptación del arquitecto de software.....	87
Tabla 8. Resultados de aceptación del líder tecnológico	89
Tabla 9. Variable Modelo de Planificación	91
Tabla 10. Variable Proceso de Planificación	91
Tabla 11. Frecuencia esperada para la variable independiente y la variable dependiente	92
Tabla 12. Frecuencia esperada para la variable independiente y la variable dependiente	93
Tabla 13. Frecuencia esperada para la variable independiente y la variable dependiente	94
Tabla 14. Distribución Chi Cuadrado Crítico	95

ÍNDICE DE GRÁFICOS

Gráfico 1. Organigrama de la Superintendencia de Economía Popular y Solidaria.	36
Gráfico 2. Modelo de Planificación de Proyectos de Desarrollo de Software (MPPDS)	55
Gráfico 3. Nivel de aceptación de los desarrolladores sobre el modelo MPPDS	87
Gráfico 4. Nivel de aceptación del arquitecto de software sobre el modelo MPPDS	88
Gráfico 5. Nivel de aceptación del líder tecnológico sobre el modelo MPPDS	89
Gráfico 6. Prueba de Chi Cuadrado dependiente	96

ÍNDICE DE ANEXOS

Para ubicar los anexos correspondientes al presente documento, es necesario ubicarse en la carpeta Anexos del presente CD y ubicarlo por su nombre.

Anexo A: Encuesta proyectos de desarrollo de software

Anexo B: Procesos de Planificación para Proyectos de Desarrollo de Software

Anexo C: Proceso de Ingeniería de Software

Anexo D: Acta de Constitución del proyecto y el Plan de Gestión del proyecto

Anexo E: Cronograma General del Proyecto

Anexo F: Plan de Gestión de un Proyecto de Ingeniería de Software

Anexo G: Documento de Definición de Requerimientos (DDR)

Anexo H: Manual Técnico de Estructuras de Información (MTEI)

Anexo I: Manual Técnico de Formularios de Información (MTFI)

Anexo J: Análisis de requerimientos con componente tecnológico

Anexo K: Estructura de Desglose del Trabajo Work Breakdown Structure (EDTWBS)

Anexo L: Diccionario de la EDT

Anexo M: Planificación del desarrollo de sistemas de software

Anexo N: Plantilla Cronograma Ingeniería de Software

Anexo O: Identificación, Evaluación y Definición de Acciones de Contingencia sobre los riesgos en proyectos con componente tecnológico

Anexo P: Diseño de componentes

Anexo Q: Diseño aplicativo de base de datos

Anexo R: Distribución de componentes de software

Anexo S: Registro de cambios del código fuente de los aplicativos internos

Anexo T: Gestión de Pruebas Unitarias, Integración y Carga

Anexo U: Registro de equipo de proyecto

Anexo V: Identificación, Evaluación y Definición de Acciones de Contingencia sobre los Riesgos en Proyectos con Componente Tecnológico

Anexo W: Encuesta de medición MPPDS

Anexo X: Resultado de la Encuesta proyectos de desarrollo de software

Anexo D1: Acta de Constitución y Plan de gestión del proyecto

Anexo E1: Cronograma General del Proyecto Acopio Integral

Anexo F1: Plan de gestión de un proyecto de ingeniería de software

Anexo H1: Manual Técnico de Estructuras de información

Anexo K1: EDT Estados Financieros

Anexo L1: Diccionario de la EDT Estados Financieros

Anexo N1: Cronograma Sistema Acopio Integral - Estados Financieros

Anexo U1: Registro de equipo de proyecto falta llenar

RESUMEN

La Intendencia de Sistemas y Tecnologías de la Información (ISTI) a través de la Dirección Nacional de Sistemas de Información (DNSI) gestiona y planifica los proyectos de ingeniería de software conforme las necesidades de la Superintendencia de Economía Popular y Solidaria (SEPS), para ello ha sido necesario aplicar metodologías de gestión y planificación de proyectos, sin embargo no existe un estándar o norma que se adapte al proceso de planificación de estos proyectos ya que no se ajustan con las necesidades institucionales, es por esta razón que el presente trabajo de investigación da a conocer la evolución de procedimientos, estándares, guías y normas para la planificación de los proyectos de desarrollo de software en el proceso de ingeniería de software, además presenta un estudio comparativo de estos estándares, guías y normas con la finalidad de diseñar un modelo de planificación para proyectos de desarrollo de software (MPPDS) que optimice el proceso de planificación de los mencionados proyectos, una vez diseñado el modelo propuesto se aplicó en el Sistema Acopio Integral – Estados Financieros en la DNSI de la ISTI. Finalmente se ha realizado el análisis y los cálculos correspondientes a los resultados generados de las encuestas aplicadas al equipo del proyecto Acopio Integral – Estados Financieros para comprobar la hipótesis planteada al inicio de la investigación.

PALABRAS CLAVE:

- **DESARROLLO DE SOFTWARE**
- **MODELO DE PLANIFICACIÓN**
- **INGENIERÍA DE SOFTWARE**
- **SUPERINTENDENCIA DE ECONOMÍA POPULAR Y SOLIDARIA**

ABSTRACT

The Intendencia de Sistemas y Tecnologías de la Información (ISTI) through the Dirección Nacional de Sistemas de Información (DNSI) manages and plans the projects of software engineering according of Superintendencia de Economía Popular y Solidaria (SEPS) needs, for it has been necessary to apply management methodologies and project planning, however there isn't standard or norm that suits the planning process of these projects, however there is no standard or norm that suits the planning process of these projects because do not fit with the institutional needs, for this reason the present investigation discloses the evolution of procedures, rules, guidelines and standards for planning the software development projects in the software engineering process, also it presents a comparative study of these rules, guidelines and standards in order to design a Planning Model for Software Development Projects (PMSDP) wich optimizes the planning of these projects, once designed the proposed model it is applied in the Acopio Integral System - Estados Financieros - Balances, into DNSI of ISTI. Finally, It has performed the analysis and calculations of the results generated from the surveys to the Acopio Integral project team to test the hypothesis posed at the beginning of this investigation.

KEY WORDS:

- **SOFTWARE DEVELOPMENT**
- **PLANNING MODEL**
- **SOFTWARE ENGINEERING**
- **SUPERINTENDENCIA DE ECONOMÍA POPULAR Y SOLIDARIA**

CAPÍTULO I

GENERALIDADES

1.1 Introducción

La Superintendencia de Economía Popular y Solidaria (SEPS) nace con la Ley Orgánica de la Economía Popular y Solidaria y del Sector Financiero Popular y Solidario (LOEPS), publicada mediante Registro Oficial Nro. 444 el 10 de mayo de 2013.

La SEPS inició sus funciones el 05 de junio de 2012, es una entidad técnica de supervisión y control de las organizaciones de la economía popular y solidaria, con personalidad jurídica de derecho público y autonomía administrativa y financiera, que busca el desarrollo, estabilidad, solidez y correcto funcionamiento del sector económico, popular y solidario, con procesos técnicos, transparentes y confiables, para contribuir al bienestar de sus integrantes y de la comunidad en general.

El presente proyecto se focalizará en el diseño de un modelo de planificación para proyectos de desarrollo de software que garantice la sistematización de los procesos y procedimientos establecidos en la institución, de acuerdo a la planificación establecida o con un mínimo de desviación; mediante la gestión de la ejecución de los proyectos de desarrollo de software y con la aplicación de técnicas, métodos y metodologías de desarrollo y de gestión de proyectos. Este modelo permitirá a la institución mejorar sus servicios de acuerdo a los reglamentos y normativas establecidas y de esta manera ofertar a sus usuarios un servicio de calidad.

El modelo que se propone incluirá la documentación necesaria para realizar la planificación de un proyecto de desarrollo de software, basado en la metodología PMI, 5ta. Edición, se centrará en el grupo de procesos “Planificación”, identificando en cada

área de conocimiento los formatos que se adapten para el desarrollo de los proyectos de software.

El tipo de investigación que se utilizará para el presente trabajo es el diseño de campo en el cual se recogen los datos de la realidad en la que se desarrolla la institución. Además se aplicará los métodos: deductivo-inductivo, el método histórico-lógico, análisis-síntesis, sistémico, modelamiento, la observación; y, la técnica de la encuesta, con la finalidad de conocer la realidad mediante la percepción directa de los objetos y fenómenos.

1.2 Justificación e Importancia

El presente proyecto está orientado a dar solución a los inconvenientes presentados en el desarrollo de nuevos proyectos de software en la Dirección Nacional de Sistemas de Información de la Superintendencia de Economía Popular y Solidaria, ya que al ser una Institución nueva aún no se encuentran consolidados los procedimientos estándar para la gestión de proyectos de desarrollo de software y genera que los proyectos sean desarrollados de acuerdo a la experiencia de cada gestor, presentando variaciones en el resultado de los desarrollos. Contar con un modelo para planificación de proyectos de desarrollo de software permitirá desarrollar de manera más organizada y poco a poco ir mejorando los procesos. Desde esta perspectiva es justificable diseñar un modelo de planificación para proyectos de software, que incluya estándares de desarrollo que serán aplicados en la implementación de proyectos informáticos por profesionales especialistas en el área, los mismos que podrán realizar aportes significativos.

1.3 Planteamiento del problema

Diariamente las instituciones públicas y en especial la SEPS tienen que asegurarse de ofrecer un servicio de calidad a sus usuarios, para mantener un nivel alto de confianza en las funciones que le han sido encomendadas. Las organizaciones como cooperativas o asociaciones solicitan a la SEPS información o apoyo en los procedimientos de

adecuación de estatutos, creación de nuevas organizaciones o acopio de información entre otros, ya que no conocen exactamente los procedimientos y normativa que los rige; requiriendo además apoyo técnico tienen inconvenientes al tratar de ingresar la información a la SEPS a través de sus aplicaciones desarrolladas, mejor conocidas como servicios electrónicos, que deben estar disponibles oportunamente para que los usuarios los utilicen de manera satisfactoria.

En algunas ocasiones estos servicios electrónicos no se encuentran disponibles a las organizaciones de acuerdo a la planificación y cronograma establecidos en las etapas iniciales del proyecto. Esto es consecuencia de los constantes cambios en la definición y determinación de requerimientos por parte de los usuarios funcionales, lo que genera muchas modificaciones en las etapas de diseño, desarrollo e implantación e influye directamente en el retraso de las entregas a los usuarios finales. También se debe a una escasa comunicación entre todos los responsables de cada segmento de la cadena involucrada en el proceso de desarrollo de software.

Al no existir un modelo de planificación para proyectos de desarrollo de software, no es posible estimar adecuadamente los tiempos y recursos necesarios para el final exitoso de un proyecto.

1.4 Formulación del problema

¿Cómo optimizar el proceso de planificación de los proyectos de desarrollo de software en la Dirección Nacional de Sistemas de información, de la Intendencias de Sistemas y Tecnologías de la Información de la Superintendencia de Economía Popular y Solidaria?

1.5 Hipótesis

El problema que se desea resolver plantea la siguiente hipótesis:

H1. ¿Con el diseño de un modelo de planificación para proyectos de desarrollo de software, es posible optimizar el proceso de planificación de proyectos de software en la Dirección Nacional de Sistemas de información, de la Intendencia de Sistemas y Tecnologías de la Información de la Superintendencia de Economía Popular y Solidaria?

- **Variables de la investigación**

- **Variable independiente:** Se crea un modelo de planificación para la fase de Planificación para proyectos de desarrollo de software.
- **Variable dependiente:** se optimiza el proceso de planificación para proyectos de desarrollo de software en la DNSI durante el período julio 2015 – noviembre 2015.
- **Conceptualización de la variable independiente:** un modelo de planificación para proyectos de desarrollo de software está descrito como un conjunto de buenas prácticas, normas y parámetros, representados bajo una secuencia específica, que facilitan y apoyan al proceso óptimo de los proyectos de desarrollo de software. Nuestra propuesta de modelo de planificación para la etapa de planificación de proyectos medianos de software es una propuesta genérica que hace referencia a las mejores prácticas ejecutadas bajo las normas y estándares de calidad de gestión de proyectos de software acoplada a la situación y necesidad propia de la DNSI de la SEPS.

- **Indicadores:**

- Planes y procedimientos de proyectos de desarrollo de software.
- Ajustes del proceso de planificación bajo las normas y estándares mundiales.
- Cumplimiento de las actividades planificadas.
- Entrega satisfactoria del producto software.
- Satisfacción del equipo de trabajo.

1.6 Objetivo general

Diseñar un modelo de planificación para proyectos de desarrollo de software en la Dirección Nacional de Sistemas de Información, de la Intendencias de Sistemas y Tecnologías de la Información de la Superintendencia de Economía Popular y Solidaria.

1.6.1 Objetivos específicos

- Elaborar el marco referencial que fundamente el diseño de un modelo de planificación para proyectos de desarrollo de software.
- Desarrollar la propuesta de diseño del modelo de planificación para proyectos de desarrollo de software.
- Aplicar el modelo de planificación para proyectos de desarrollo de software.
- Validar el modelo de planificación para proyectos de desarrollo de software.

CAPÍTULO II

MARCO TEÓRICO

2.1 Introducción

Debido a la creciente globalización, el incremento de la competencia en los mercados de bienes y servicios y la rapidez en el desarrollo de los sistemas y las tecnologías de la información, se considera a la información como un insumo fundamental dentro de las empresas. Es por esta razón que los sistemas de información se han constituido en un elemento clave para la gestión, el crecimiento y la competencia de la organización empresarial, sea ésta de bienes o servicios, pública o privada.

Así mismo es necesario considerar que sin una gestión del proyecto y una efectiva planificación del mismo, se corre el riesgo de demorarse o consumir recursos mayores a los planificados. Esto se ve reflejado en la cantidad de proyectos de desarrollo de software que no han cumplido con las metas propuestas, en términos de tiempo de entrega, funcionalidad o costes.

Para solventar estos riesgos nace la ingeniería de software con la idea de planificación del desarrollo, el mismo que ha generado cambios notables con el paso del tiempo, en el ámbito económico y en la productividad de software.

A partir de la creación de la SEPS y la definición de sus procesos institucionales, el uso de la información ha sido indispensable para consolidar su funcionamiento, es por ello que poco a poco se han ido implementado procesos de ingeniería de software en el desarrollo de los sistemas, de tal manera que permitan la madurez y consolidación de los procesos institucionales, particularmente de los procesos tecnológicos.

2.2 Antecedentes Históricos

2.2.1 Evolución de las metodologías para la planificación de los proyectos de desarrollo de software en el proceso de ingeniería de software en relación al proceso de desarrollo de software

Durante mucho tiempo se ha tratado de instaurar procesos y procedimientos que mejoren la productividad y la calidad de los sistemas de información, a través de una planificación de las tareas que predigan de una forma más exacta los recursos a ser utilizados en el desarrollo de un software. Estos procesos y procedimientos se han convertido en una parte esencial de la ingeniería de software ya que han aportado con su potencial para mejorar el software.

Así mismo se han establecido marcos de trabajo que permitan sistematizar o formalizar las diferentes tareas que se ejecutan dentro del proceso de desarrollo de un software, enfocados en aplicar técnicas, herramientas y procesos de planificación que van desde la gestión de proyectos con metodologías formales para el desarrollo del producto de software hasta la aplicación de los métodos ágiles que permiten involucrar directamente al usuario final en todo el proceso de desarrollo de software y permiten mejorar la entrega de un producto acorde a las necesidades del usuario en el tiempo y con el coste establecidos, a lo largo del tiempo.

Con la finalidad de comprender la influencia de la planificación de proyectos dentro del proceso de ingeniería de software, se presenta a continuación un análisis sobre la evolución de las metodologías para la planificación de los proyectos de desarrollo de software en el proceso de ingeniería de software en relación al proceso de desarrollo de software y de la mano con la evolución del hardware:

- **Sus inicios – 1950 hasta 1965**

En la década de los 50's aparece la primera computadora denominada ENIAC, eran máquinas grandes, pesadas y el software que permitía su funcionamiento era considerado como un componente añadido.

Aparece el término y la actividad de la programación (codificación) que más tarde sería la base del desarrollo de software, el reto mayor era programar los algoritmos para que los computadores ejecuten los procesos y reportes requeridos. De esta manera emerge el rol de programador, el mismo que realiza un trabajo para un usuario y se convierte en un intermediario entre el usuario final y el computador. Este usuario define vagamente los requerimientos y el programador diseña la solución y la desarrolla. La distribución del software a medida era limitada, los programas eran pequeños y se utilizaba el lenguaje ensamblador. El desarrollo en general era a base de prueba y error, es decir se trabaja con la idea de “codificar y corregir”.

Los programadores tenían que volver a codificar todos sus programas para ser ejecutados en estas nuevas máquinas ya que el hardware del computador era específico para la aplicación y las tareas de negocio necesitaban diferentes máquinas. Los programadores no tenían equipos en sus escritorios y tenían que ir a la "sala de máquinas" para ejecutar las tareas que únicamente podían ser realizadas por el personal operativo.

El campo del desarrollo de software era tan nuevo que la idea de planificación o gestión era inexistente, se da mayor énfasis a la tarea de escribir código en un lenguaje de programación. Era casi imposible hacer predicciones de la fecha de finalización del proyecto, los costos y los recursos en general. No existía un planteamiento previo y claro en la definición de requerimientos. No existía documentación de ningún tipo y empezaban a crearse pocos métodos formales los cuales no eran aplicados.

A partir de 1955 y debido a la necesidad de traducir frecuentemente el software viejo para atender las necesidades del nuevo hardware, los lenguajes de programación de tercera generación evolucionaron como FORTRAN, COBOL y ALGOL. Conforme van apareciendo los nuevos lenguajes de programación de alto nivel, los programadores empiezan a preocuparse más por la complejidad de las aplicaciones que por la estructura de los ordenadores.

Algunas empresas vendían el servicio de construcción de software personalizado, sin embargo no existían empresas de desarrollo de software vendiendo paquetes de software hasta que surge la noción de reutilización, la programación modular y la abstracción de datos, con estos nuevos conceptos grupos como SHARE, el grupo de usuario científico de IBM, empiezan a ofrecer catálogos de componentes reutilizables. La academia todavía no enseñaba los principios de las ciencias de la computación y las organizaciones de usuarios comúnmente empiezan a liberar el software.

- **La crisis del software - 1965 hasta 1985**

Debido a la crisis del software surge la necesidad de controlar el proceso de desarrollo de software y para 1968, (G, 1998) en el transcurso de una conferencia de la OTAN en Garmisch nace la ingeniería de software con la idea de crear e implantar procedimientos y metodologías de desarrollo de software, las mismas que han ido evolucionando en forma paralela con la evolución de la programación. Se definen modelos de ciclo de vida del proceso de desarrollo de software para que sea tomada como referencia para ejecutar las actividades implícitas en dicho proceso.

Para 1970 aparece el modelo de desarrollo en cascada propuesto por Winston Royce (Royce, 1970) y para 1985 el Departamento de Defensa de los Estados Unidos, lo estableció como un proceso estandarizado, el mismo incluye las fases de: Análisis de la situación actual, Diseño lógico, Diseño físico, Programación, Implantación y Operación

y mantenimiento; sin embargo, no se presenta una de las fases fundamentales del desarrollo de software que es la fase de Pruebas.

Se empieza a crear las empresas de desarrollo de software principalmente con el objetivo de crear software a la medida, sin embargo los fracasos tanto a nivel de proyectos como de empresas continúan, debido a que las condiciones de contratación no son claras y los precios son fijos, las métricas del trabajo y el alcance del proyecto a realizar no son claros, en algunas ocasiones la corrección de errores genera más errores, o existen errores que son cambios en los requerimientos.

Como se tiene un enfoque cascada, los errores sólo se detectan en la última fase del desarrollo; estos errores en varias ocasiones se refieren a problemas de diseño, o imprecisiones de requerimientos. Al ser detectados al final, se debe volver a hacer mucho del software, e incluso se cancelan los proyectos por la cantidad de trabajo perdido que habría que desarrollar nuevamente. Esto conlleva a que las empresas de software tengan fuertes riesgos y en la mayoría de las ocasiones se quiebran en menos de 5 años.

Entre 1975 y 1985 se genera una etapa de consolidación de la gestión del proceso de desarrollo de software a través del control de las actividades que serán ejecutadas durante el mismo y la aplicación de métricas de medición para estimar los recursos como costo y tiempo, calidad de software, conceptos de reusabilidad. El programador no solo codifica sino que se convierte en un analista, un diseñador y hasta gestor del proyecto según la necesidad. Se da inicio al concepto de ingeniero de software.

El tradicional modelo de cascada evoluciona al método en V, en donde se destaca la importancia de las pruebas tanto a nivel de programación como a nivel de usuario, entre ellas tenemos: unitarias, de integración, de sistema y pruebas de aceptación por parte del usuario. Esto implica a la vez un costo adicional para las empresas en la especialización de las tareas: análisis de requerimientos, diseño del sistema y pruebas. Es decir no sólo se requieren ingenieros que codifiquen sino también analistas e

ingenieros de calidad, además se hace necesaria la adquisición de infraestructura y herramientas para realizar el control y aseguramiento de la calidad de forma adecuada.

En esta misma década 1975 a 1985 los sistemas se empiezan a programar uno a continuación de otro, es decir que dada una entrada se realiza un procedimiento que genera una salida, esto permite identificar claramente los requerimientos permitiendo de esta manera la planificación y control de la ejecución del proyecto; sin embargo, el desarrollo del software es más que entradas, procesos y salidas consecutivas, se refiere principalmente a la interacción con el usuario en la que se presentan varios caminos y una fuerte integración con otros sistemas. Esto genera mayor posibilidad de error en los sistemas, se hacen necesarias las pruebas de los nuevos requerimientos, pruebas para verificar la correcta funcionalidad del sistema original y pruebas de las características del sistema que no son de procesamiento, validaciones, ni salidas del sistema sino referentes a temas técnicos.

En 1962 aparece el concepto de Estructura de Desglose del Trabajo (EDT) como la evolución de los mapas mentales y para 1969 es modificada por el PMI (PMBOK, 2013) como un gran aporte para la gestión de proyectos, esta estructura permite identificar claramente los paquetes de trabajo y tareas a las cuales se debe asignar recursos, permitiendo de esta manera llevar una planificación del proyecto, sin embargo depende de las herramientas utilizadas para la asignación de recursos.

En 1969 nace en los Estados Unidos el PMI, Project Management Institute (PMBOK, 2013), es una organización internacional sin fines de lucro que agrupa a profesionales dedicados a la gestión de proyectos e implícitamente a la planificación y control de los mismos a través de procesos y áreas de conocimiento.

En 1975 se difunde el documento de “The Mythical Man-Month” (Brooks, 1995), un ensayo de Ingeniería de software escrito por Fred Brooks, basado en las experiencias del autor en el proyecto OS/360 en IBM, a través de este ensayo se propone cómo mejorar los procesos en la gestión de un proyecto, incluida la planificación del proyecto.

En 1979 aparece la Métrica del Punto de Función (Pressman, 2010), definida por Allan Albretch de IBM, este método permite medir el tamaño de software en base a la funcionalidad entregada por el usuario e independiente de la tecnología que se utilice para la construcción, de esta manera permite un mayor control y planificación del proyecto.

En 1981 se origina el Modelo Constructivo de Costes (Constructive Cost Model), más conocido como COCOMO, fue desarrollado por B. W. Boehm (Boehm, 2000), exponiéndolo detalladamente en su libro “Software Engineering Economics” (Boehm, Software Engineering Economics, 1981). Este modelo permite la planificación de los proyectos en base a la complejidad del software a través de una jerarquía de modelos de estimación de costes software que incluye submodelos básico, intermedio y detallado.

Esta etapa se caracterizó por la Crisis del software en la cual destacó el estudio del Standish Group (“Reporte del Caos”), el mismo indicaba que solo el 16% de los proyectos de software fueron exitosos. Debido a la falta de la gestión, planificación y control de los proyectos de desarrollo de software se generaron muchos problemas como: la generación de errores multimillonarios por no diseñar una coherente arquitectura antes de iniciar el desarrollo, el mismo que generó un desborde en el presupuesto y en el tiempo estimados, no existía seguridad en el software lo que permitió a los hackers robar identidades, costando tiempo, dinero y reputación; los errores en el software causaron pérdidas incluso de vidas.

Con el uso de los procesos en el desarrollo de software se fue tratando de controlar los problemas generados en esta era, sin embargo aún se mantenían inconvenientes, ya que las actividades variaban dependiendo del software a desarrollarse y de la organización, se debía seleccionar un tipo de modelo de proceso para la ingeniería del software y debía adaptarse según el tipo de proyecto y los requerimientos de validación y control del sistema resultante. Si bien estos procesos nos facilitan la gestión de los

proyectos de desarrollo de software, su implementación puede llegar a ser muy tediosa y con abundante documentación.

- **Inicio de la ingeniería - 1985 hasta 1999**

Con la crisis del software y a la vez su evolución se consolidan las tecnologías de software y se generan varios conceptos sobre la ingeniería de software como: los procesos de desarrollo de software; los métodos y las metodologías; algunas fases adicionales a la codificación como el análisis de requerimientos, diseño del sistema, pruebas y despliegue; la gestión del proyecto, arquitectura de software, arquitectura empresarial y modelamiento de negocio.

En 1984 a través del Departamento de Defensa de los Estados Unidos se crea el Instituto de Ingeniería de Software (SEI, Software Engineering Institute), con la finalidad de impulsar a las empresas de desarrollo de software a mejorar sus desarrollos con el uso de procesos y métricas estandarizadas.

La IEEE en 1984 desarrolla la Norma IEEE 1058.1 , en el cual se especifica qué se debe utilizar para desarrollar los planes para la gestión de proyectos. Esta norma incluye los siguientes estándares:

1983, Glosario Estándar IEEE de Terminología de Ingeniería de Software.- Glosario que se utiliza durante el proceso del desarrollo del proyecto.

1983, Estándar IEEE para Planes de Gestión de la Configuración.- Se definen diferentes versiones por cada uno de sus artefactos.

1983, Estándar IEEE para la Documentación de Pruebas de Software.- Permite identificar si el producto cumple con los requerimientos del usuario y sus expectativas.

1984, Estándar IEEE para Planes de Aseguramiento de Calidad del Software.- Este plan define qué es un software de alta calidad.

1986, Guía IEEE para la Planificación de Aseguramiento de Calidad del Software.- Se asegura la calidad mediante el uso de procedimientos, un marco de trabajo y estándares corporativos.

1986, Estándar IEEE para la Planificación de Verificación y Validación de Software.- Permite identificar si el producto cumple con los requerimientos del usuario y sus expectativas.

1998, IEEE Std 1058 – 1998, Estándar IEEE para la Planificación de la Gestión de un Proyecto Software.- Este estándar describe el formato y el contenido de un plan de gestión de un proyecto software. Entre septiembre de 1984 y septiembre de 1996 se celebraron diez reuniones en los Estados Unidos con la finalidad de crear y actualizar este estándar entre el 13 de diciembre de 1984 y finalmente fue aceptada en septiembre de 1996 por los Componentes de Estándares del IEEE.

1998, IEEE Std 1220 – 1998, Estándar IEEE para la Aplicación y Gestión de Procesos de Ingeniería de Sistemas.- Este estándar describe las actividades de ingeniería de sistemas y el proceso requerido en todo el ciclo de vida del sistema para el desarrollo de sistemas que permitan conocer las necesidades, requisitos de los clientes, y sus límites.

1998, IEEE Std 1490 – 1998 Guía IEEE Adopción del Estándar PMI - Una Guía para el Cuerpo de la Gestión del Proyecto del Conocimiento.- Este documento es la adopción IEEE del Cuerpo de la Gestión del Proyecto de Conocimiento definido por el Instituto de Gestión del Proyecto. Identifica y describe al conocimiento generalmente aceptado con lo que concierne a la gestión del proyecto.

Con el fin de incrementar la calidad de los productos de software fue necesario mejorar el conocimiento de los procesos, es así que se crearon modelos que permitan medir la madurez del proceso para producir software como:

El Modelo de Madurez de Capacidades (CMM, Capability Maturity Model), fue desarrollado en 1987 por la Universidad Carnegie-Mellon para el SEI (Software Engineering Institute), es un modelo para evaluar los procesos críticos relativos al desarrollo e implementación de software. Este modelo ha madurado a una serie de reglas y de buenas prácticas apoyada en una serie de herramientas software y no software para el uso o puesta en marcha de procesos de mejora continua en desarrollo y mantenimiento de sistemas informáticos.

En 1995 evoluciona a CMMI (Capability Maturity Model for Integration, Modelo de Madurez de Capacidades Integrado) y fue publicado por el SEI (Software Engineering Institute), al igual que CMM está dedicado a la ingeniería de sistemas y no utiliza niveles de madurez generales sino que en cada área de proceso una organización puede alcanzar un determinado nivel de madurez.

- **Desde el 2000 hasta la actualidad**

Para el año 2001 en Snowbird, Utah se reunieron críticos de los modelos de mejora de desarrollo de software, en la cual definieron a los métodos que estaban surgiendo como alternativa a las metodologías formales como “Métodos Ágiles”, además resumieron los principios sobre los que se basan los métodos alternativos en cuatro postulados, lo que se denominó como Manifiesto Ágil, en donde se prefieren: individuos e interacciones sobre procesos y herramientas; software funcionando sobre documentación extensiva; colaboración con el cliente sobre negociación contractual, respuesta ante el cambio sobre seguir un plan. Esta corriente promete que los desarrollos se realicen más rápido, con buena calidad y que sean adaptables a los cambios, sin embargo no permiten una planificación adecuada para estimar los recursos

necesarios dentro del proyecto de desarrollo de software. Entre estas metodologías tenemos: Scrum, XP, Kanban, OpenUP, entre otras.

Por otro lado en octubre de 1998 el Instituto Nacional de Normalización de los Estados Unidos (ANSI) acreditó a PMI como desarrollador de estándares a través de procesos que se detallan en la Guía del PMBOK. Este estándar es aplicable a todo tipo de proyectos incluidos los de desarrollo de software, permite la planificación y control del proyecto de una manera más efectiva a través de artefactos de entrada, salida, técnicas y herramientas identificadas dentro de los grupos de procesos y las áreas de conocimiento.

En octubre de 2007 ISO establece su “Comité Técnico ISO/TC 236 Project Management” en Londres, con la finalidad de elaborar la primera Norma Internacional para la Dirección de Proyectos, generando de este modo la Norma ISO 2150 y su publicación oficial se realizó en septiembre de 2012. La aplicación de esta Norma en las organizaciones aporta para mejorar su eficiencia y eficacia a través de una estructura funcional de sistemas de gestión sobre todas las áreas o actividades que deben ejecutarse dentro de un proyecto para que sus profesionales gestionen los proyectos y las operaciones enfocadas a aplicar procesos de mejora continua.

2.3 Antecedentes Conceptuales y Referenciales

2.3.1 Caracterización tecnológica del proceso de ingeniería de software

- **Definición de Software**

Según Roger Pressman (Pressman, 2010), “el software de computadora es el producto que construyen los programadores profesionales y al que después le dan mantenimiento durante un largo tiempo. Incluye programas que se ejecutan en una computadora de cualquier tamaño y arquitectura”.

Según Ian Sommerville (SOMMERVILLE, 2005), software “son programas de ordenador y la documentación asociada. Los productos de software se pueden desarrollar para algún cliente en particular o para un mercado general”

Es un conjunto de instrucciones o programas que cuando se ejecutan proporcionan las funcionalidades para las cuales fueron desarrolladas y permite la manipulación de la información de una forma adecuada.

- **Ingeniería de software**

Según Roger Pressman (Pressman, 2010), “la ingeniería de software está formada por un proceso, un conjunto de métodos (prácticas) y un arreglo de herramientas que permite a los profesionales elaborar software de cómputo de alta calidad”.

Según Ian Sommerville (Sommerville, 2005), “la ingeniería de software es una disciplina de ingeniería que comprende todos los aspectos de la producción de software. La ingeniería de software es parte del proceso de ingeniería de sistemas”.

El término Ingeniería de software surge en 1968 durante una conferencia del Comité de ciencia de la OTAN en Garmisch – Alemania, debido a la necesidad de instaurar y aplicar en el desarrollo de software, principios de ingeniería orientados a generar productos de manera económica, fiable y que funcionen eficientemente sobre máquinas reales. Además con la ingeniería de software se logra la evolución de un nuevo campo de estudio que ha evolucionado para ser una ciencia con principios y teorías en la cual se aplican métodos, metodologías, herramientas, conceptos y prácticas para el análisis de requerimientos, diseño del sistema, construcción y pruebas tecnológicas, pruebas funcionales y despliegue.

El proceso de ingeniería de software abarca tres elementos clave métodos, herramientas y procedimientos, facilita al gestor controlar el proceso de desarrollo de software y suministra, a quienes la practican, las bases para construir software de alta calidad de una forma productiva.

- **Proceso de ingeniería de software**

Es un conjunto de políticas, tecnologías, artefactos y procedimientos que implican creatividad y el juicio de expertos, con la finalidad de analizar, desarrollar, desplegar y mantener un producto de software a través de la transformación de entradas en salidas. Las actividades fundamentales del proceso de desarrollo de software son:

- **Análisis y levantamiento de requerimientos del software:** En esta actividad se define la funcionalidad y restricciones que debe cumplir el software.
- **Diseño:** se diseña el software de acuerdo a la especificación del requerimiento.
- **Implementación:** Se construye el software de acuerdo a la especificación del requerimiento.
- **Pruebas:** el software debe probarse y validarse, para asegurar que cumpla con lo que quiere el cliente.
- **Mantenimiento:** el software debe evolucionar, para adaptarse a las necesidades del cliente.

Además de estas actividades, según Pressman se pueden aplicar un conjunto de "actividades protectoras": seguimiento y control de proyecto de software. Revisiones técnicas formales, entre otras. El proceso de software también tiene elementos, los cuales se presentan a continuación:

- Un marco común del proceso que define un número de actividades del marco de trabajo con independencia de la complejidad y tamaño.
- Un conjunto de tareas que contienen hitos de proyectos, entregables, productos de software y puntos de garantía de calidad.
- Actividades de protección que son independientes de cualquier actividad del marco de trabajo y aparecen durante todo el proceso de desarrollo de software.

- **Modelo del ciclo de vida de desarrollo de software**

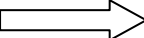
Un modelo de ciclo de vida es un conjunto estructurado de fases y actividades que tienen inicio y fin y se encuentran interrelacionadas entre sí, éstas son requeridas para desarrollar un producto de software. La selección de un modelo de ciclo de vida se puede seleccionar en base a la naturaleza del proyecto y de la aplicación, a los métodos o herramientas a utilizarse, los controles o productos solicitados.

Actualmente se identifican dos tendencias principales para el desarrollo de software: los modelos tradicionales y los modelos ágiles.

A continuación se presenta una matriz con una breve descripción de algunos modelos de ciclo de vida de desarrollo de software:

Tabla 1.
Modelos de ciclo de vida de desarrollo de software

NOMBRE	PROCESO	DESCRIPCIÓN	VENTAJAS	DESVENTAJAS	SE UTILIZA
CASCADA	Secuencia de una serie de fases	Consiste en la ejecución secuencial de una serie de fases que suceden. Cada fase genera documentación para la siguiente. Esta documentación debe ser aprobada. Una fase no comienza hasta que la anterior ha terminado. Requiere disponer de unos complementos u precios al principio del desarrollo.	Facilita la gestión del desarrollo	Los proyectos reales rara vez siguen el flujo secuencial propuesto - Normalmente es difícil para el cliente establecer claramente todos los requisitos y el paradigma lo exige -El cliente debe ser paciente ya que solo hasta las etapas finales de desarrollo estará disponible una versión operativa del software.	Para sistemas bien comprendidos.
ESPIRAL	Ciclos de vida (fases, prototipos)	El modelo define cuatro actividades representadas en cuatro cuadrantes 1.PLANIFICACIÓN 2.ANÁLISIS DE RIESGO. 3.INGENIERÍA 4.-EVALUACIÓN DEL CLIENTE	- Centra su atención en la reutilización de componentes y eliminación de errores en información descubierta en fases iniciales - Los objetivos de calidad son el primer objetivo. - Integra desarrollo con mantenimiento. -Provee un marco de desarrollo de hardware/software.	- La evaluación de riesgos es compleja - Excesiva flexibilidad para algunos proyectos.	Con requerimientos estables y sistemas de seguridad críticos

Continúa 

INCREMENTA	Módulos incrementando funcionalidades	<p>Se realiza construyendo por módulos que cumplen las diferentes funciones del sistema.</p> <p>Esto permite ir aumentando gradualmente las capacidades del software. Al final de cada ciclo se entrega una versión al cliente que contiene una nueva funcionalidad. Este ciclo de vida nos permite realizar una entrega al cliente antes de terminar el proyecto.</p>	<p>Construir un sistema pequeño siempre es menos riesgoso que construir un sistema grande.</p> <ul style="list-style-type: none"> - Como desarrollamos independientemente las funcionalidades, es más fácil relevar. - Si se detecta un error grave, sólo desechamos la última iteración. - No es necesario disponer de los requerimientos de todas las funcionalidades en el comienzo del proyecto 	<p>Poca visibilidad en el proceso.</p> <p>> Los módulos son parciales y no representan el software final completamente.</p>	<p>Este modelo de ciclo de vida no está pensado para cierto tipo de aplicaciones, sino que está orientado a cierto tipo de usuario o cliente.</p> <p>Podremos utilizar este modelo de ciclo de vida para casi cualquier proyecto, pero será verdaderamente útil cuando el usuario necesite entregas rápidas, aunque sean parciales.</p>
PROCESO DE DESARROLLO UNIFICADO	Ciclos en los que se utiliza el lenguaje unificado de modelo UML	<ul style="list-style-type: none"> - Es un conjunto de metodologías adaptables al contexto y necesidades de cada organización - Describe cómo aplicar enfoques para el desarrollo del software, llevando a cabo unos pasos para su realización. - Se centra en la producción y mantenimiento de modelos del sistema. - Pretende implementar las mejores prácticas de ingeniería de Software. - Se caracteriza por ser iterativo e incremental, estar centrado en la arquitectura y guiado por los casos de uso. 	<ul style="list-style-type: none"> - Se adapta a cualquier tipo de proyecto. - Es un modelo muy completo. - Entrega un buen software final. 	<ul style="list-style-type: none"> - Es muy tardado y complejo cada ciclo lo que hace poco fácil de manejar. 	<p>En todo proyecto bien definido</p>

- **Metodologías de desarrollo de software**

Son un marco de trabajo usado para estructurar, planificar y controlar el proceso de desarrollo en sistemas de información. Una metodología de desarrollo de software se refiere a un framework que es usado para estructurar, planear y controlar el proceso de desarrollo en sistemas de información.

Una gran cantidad de métodos han sido desarrollados diferenciándose por su fortaleza y debilidad. El framework para metodología de desarrollo de software consiste en:

- Una filosofía de desarrollo de programas de computación con el enfoque del proceso de desarrollo de software.
- Herramientas, modelos y métodos para asistir al proceso de desarrollo de software.

Estos framework son a menudo vinculados a algún tipo de organización, que además desarrolla, apoya el uso y promueve la metodología. La metodología es a menudo documentada en algún tipo de documentación formal (Sommerville, 1992).

Tabla 2. Diferencias entre metodologías ágiles y no ágiles

METODOLOGÍAS ÁGILES	METODOLOGÍAS TRADICIONALES
Basadas en heurísticas provenientes de prácticas de producción de código	Basadas en normas proveniente de estándares seguidos por el entorno de desarrollo
Especialmente preparados para cambios durante el proyecto	Cierta resistencia a los cambios
Impuestas internamente (por el equipo)	Impuestas externamente
Proceso menos controlado, con pocos principios	Proceso mucho más controlado, con numerosas políticas/ normas
No existe contrato tradicional o al menos es bastante flexible	Existe un contrato prefijado

Continúa 

El cliente es parte del equipo de desarrollo	El cliente interactúa con el equipo de desarrollo mediante reuniones
Grupos pequeños (<10 integrantes) y trabajando e el mismo sitio	Grupos grandes y posiblemente distribuidos
Pocos artefactos	Más artefactos
Pocos roles	Más roles
Menos énfasis en la arquitectura del software.	La arquitectura del software es esencial y se expresa mediante modelos

2.3.2 Caracterización tecnológica de la planificación de los proyectos de desarrollo de software

- **Proyecto**

Proceso único que consiste en un conjunto de actividades coordinadas y controladas, con fechas de inicio y finalización llevadas a cabo para lograr un objetivo, conforme a requisitos y requerimientos específicos, incluyendo limitaciones de tiempo, coste y recursos (ISO 10006).

- **Gestión de proyectos**

Gestión de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto, con el fin de cumplir o exceder las expectativas y necesidades de los involucrados del proyecto.

Dentro del campo de la planificación como parte de la gestión de proyectos de una organización, tenemos algunos referentes, entre los más representativos podemos mencionar los Fundamentos de la Dirección de Proyectos – Guía de PMBOK, cuya finalidad principal es identificar el subconjunto de Fundamentos de la Dirección de Proyectos generalmente reconocido como buenas prácticas.

En base a la Guía del PMBOK se presentan los siguientes grupos de procesos que permitirán una adecuada gestión de un proyecto de ingeniería de software, entre los cuales la de mayor interés para nuestro contexto es el grupo de Planificación:

- Iniciación,
- Planificación,
- Ejecución,
- Seguimiento y control del proyecto, y;
- Cierre del proyecto.

- **Planificación de proyectos.**

La planificación de proyectos proporciona un marco conceptual que permite al gerente de proyectos establecer el alcance total del esfuerzo, definir y refinar los objetivos y desarrollar la línea de acción necesaria para alcanzar dichos objetivos, además permitirá hacer estimaciones razonables de recursos, costo y calendario. Estas estimaciones deben intentar definir los escenarios de mejor caso y peor caso con la finalidad de delimitar los resultados del proyecto.

- **Gestión de proyectos de desarrollo de software**

La gestión de la ingeniería de software es la aplicación de actividades administrativas – planeación, coordinación, medición, monitorización, control y reporte – para asegurar que el desarrollo y el mantenimiento de software sea sistemático, disciplinado y cuantificable. (IEEE610.12-90).

Según José Fernández León, la gestión de un proyecto de desarrollo de software es “una gestión de la calidad del producto que se manifiesta en diversos aspectos de éste y de los procesos implicados en las fases de desarrollo”.

La gestión del proyecto de software es el primer nivel del proceso de ingeniería de software, porque cubre todo el proceso de desarrollo. Para conseguir un proyecto de software fructífero se debe comprender el ámbito del trabajo a realizar, los riesgos en los que se puede incurrir, los recursos requeridos, las tareas a llevar a cabo y el plan a seguir (Knowledge, 2003.).

Dentro de las actividades de gestión de un proyecto de desarrollo de software se presentan las siguientes:

- Planificación: Predeterminación de un curso de acción para alcanzar los objetivos organizacionales.
- Organización: Arreglo de las relaciones entre las unidades de trabajo para el cumplimiento de objetivos y el otorgamiento de responsabilidad y autoridad para obtener esos objetivos.
- Staffing: Selección y entrenamiento de personas para puestos en la organización.
- Dirección: Creación de una atmósfera que apoye y motive a la gente para alcanzar los resultados finales deseados.
- Control: Establecimiento, medición y evaluación del desempeño de las actividades a través de los objetivos planeados.

- **Planificación de proyectos de desarrollo de software.**

Según Gonzalo León Serrano (Sommerville, 2005), la Planificación del desarrollo y del mantenimiento “implica la gestión tanto de los recursos tanto humanos como materiales necesarios para obtener el producto de software incluyendo el entrenamiento requerido por los componentes del equipo de trabajo y los potenciales usuarios del sistema a desarrollar”. La planificación involucra la especificación de objetivos y metas para un proyecto y las estrategias, políticas, planes y procedimientos para alcanzarlos.

Todo proyecto de ingeniería de software debe partir con un buen plan. La planificación es necesaria por la existencia de incertezas sobre el ambiente del proyecto software y sobre fuentes externas. La planificación enfoca su atención en las metas del proyecto, riesgos potenciales y problemas que puedan interferir con el cumplimiento de esas metas. Los principales problemas en la planificación de un proyecto de ingeniería de software incluyen los siguientes puntos:

- Requerimientos incorrectos e incompletos.
- Muchas especificaciones de requerimientos son inestables y sujetas a cambios mayores.
- La planificación no se lleva a cabo por la creencia errónea de que es una pérdida de tiempo y los planes cambiarán de todos modos.
- La planificación de costos y plazos no es actualizada y se basa en necesidades de mercadeo y no de los requerimientos del sistema.
- Es difícil estimar el tamaño y complejidad del proyecto de software de modo de realizar una estimación de costos y plazos realista.
- Los costos y plazos no son re estimados cuando los requerimientos del sistema o el ambiente de desarrollo cambia.
- No se manejan factores de riesgo.
- La mayoría de las organizaciones de desarrollo de software no recolectan datos de proyectos pasados.
- Las compañías no establecen políticas o procesos de desarrollo de software.

2.3.3 Estándares para la planificación del desarrollo de software

2.3.3.1 Guía de los Fundamentos para la Dirección de Proyectos (PMBOK)

La guía del PMBOK “contiene el estándar, reconocido a nivel global y es la guía para la dirección de proyectos. Este estándar es un documento formal que describe las normas, métodos, procesos y prácticas para la dirección de proyectos, evolucionó a partir de las buenas prácticas reconocidas de los profesionales dedicados a la dirección

de proyectos que han contribuido a su desarrollo”. Además esta guía detalla la información mediante la descripción de entradas, salidas, herramientas y técnicas utilizadas en la dirección de proyectos y proporciona una visión de las interacciones entre procesos dentro de diez Áreas de Conocimiento y cinco Grupos de Procesos que se mencionan a continuación:

- **Grupos de Procesos:**

- Inicio,
- Planificación,
- Ejecución,
- Monitoreo y control, y;
- Cierre.

- **Áreas de conocimiento:**

- Integración,
- Alcance,
- Tiempo,
- Costo,
- Calidad,
- Recursos Humanos,
- Comunicación,
- Riesgo,
- Adquisiciones, y;
- Interesados.

- **Grupo de Procesos de Inicio.**

En esta fase se definen el alcance, los objetivos y los recursos globales del proyecto, se establece en forma general qué es lo que se debe hacer, se identifican los interesados que interactuarán e influirán en el resultado del proyecto con la finalidad de alinear sus expectativas con el propósito del proyecto y se seleccionará al gerente del proyecto. Estos datos serán registrados en el Acta de Constitución del proyecto, con la cual se considera autorizado oficialmente su inicio.

- **Grupo de Procesos de Planificación.**

En esta fase el gerente del proyecto establece el alcance total del esfuerzo, define y refina los objetivos y desarrolla la línea de acción necesaria para alcanzar dichos objetivos. Además con el uso de las herramientas y técnicas proporcionadas se puede hacer estimaciones razonables de recursos, costo y calendario. Con la finalidad optimizar la calidad de los procesos del grupo de procesos de Planificación, éstos se interrelacionan de acuerdo a las siguientes áreas de conocimiento:

- **Gestión de la integración.**

Esta área incluye los procesos que permitirán al líder tecnológico la coordinación de los elementos del proyecto, es soporte fundamental para alcanzar los requerimientos y gestionar las expectativas de los interesados del proyecto.

- **Gestión del alcance.**

La gestión del alcance del proyecto permite incluir todo el trabajo requerido para completar con éxito el proyecto, aquí se define qué se incluye y que no se incluye en el proyecto.

- **Gestión del tiempo**

En este proceso se define la duración del plazo del proyecto de ingeniería de software, tomando como base el cronograma general del proyecto que fue realizado por el gerente del proyecto.

- **Gestión de la comunicación**

En este proceso se establecen los lineamientos para que la planificación, distribución, gestión, control y disposición de la información sean oportunos y adecuados a todos los miembros del equipo del proyecto de ingeniería de software.

- **Grupo de Procesos de Ejecución.**

Este grupo de procesos nos permite completar el trabajo que se ha definido en el plan para la dirección del proyecto con la finalidad de cumplir con los objetivos propuestos.

- **Grupo de Procesos de Seguimiento y Control.**

Estos procesos se encargan de controlar, monitorear y regular el proceso de proyecto para identificar si es necesario realizar cambios en el plan para ejecutar los mismos.

- **Grupo de Procesos de Cierre.**

Consta de los procesos que permiten dar por finalizado todas las actividades y tareas de los grupos de procesos ejecutados y cerrar formalmente una fase o el proyecto.

2.3.3.2 Directrices para la Dirección y Gestión de Proyectos (Norma ISO 21500)

La norma ISO 21500, es un estándar internacional desarrollado por la Organización Internacional de Normalización (ISO), explica los principios básicos y lo que constituye una buena práctica en los proyectos de gestión, busca orientar a las empresas en su

gestión. La norma ISO 21500 da soporte a uno de los principales motores económicos mundiales: los proyectos. Según datos del Project Management Institute se prevé que para el año 2016 habrá 32,6 millones de empleados de todo el mundo trabajando en proyectos de diversa índole. Saber gestionar los proyectos es un asunto que adquiere cada vez una mayor importancia. Su objetivo es ayudar a Directores de Proyecto, principiantes o experimentados, a aplicar las mejores prácticas en la gestión de sus proyectos, mejorando los resultados de negocio y cumpliendo sus objetivos con éxito.

Según esta guía, la dirección de proyectos se distribuye en cinco Grupos de Procesos y diez Temas, como se indica a continuación:

- **Grupos de Procesos:**

- Inicialización,
- Planificación,
- Implementación,
- Control, y;
- Cierre.

- **Temas de conocimiento:**

- Integración
- Interesados,
- Alcance,
- Recursos Humanos,
- Tiempo,
- Costo,
- Riesgo,
- Calidad,
- Adquisiciones, y;

- Comunicación.

Esta norma describe los procesos antes mencionados y establece entradas y salidas, sin embargo no establece técnicas y herramientas para ejecutar los procesos.

2.3.3.3 Norma IEEE 1058-1998 para la Planificación de Gestión de Proyectos Software.

1998, IEEE Std 1058 – 1998 (Ruiz, 1999), Estándar IEEE para la Planificación de la Gestión de un Proyecto Software.- Este estándar describe el formato y el contenido de un plan de gestión de un proyecto software. Entre septiembre de 1984 y septiembre de 1996 se celebraron diez reuniones en los Estados Unidos con la finalidad de crear y actualizar este estándar entre el 13 de diciembre de 1984 y finalmente fue aceptada en septiembre de 1996 por los Componentes de Estándares del IEEE.

Este estándar debe contener los siguientes elementos fundamentales para la planificación de la Gestión de un proyecto software: página de título, hoja de revisión, prefacio, tabla de contenidos, lista de figuras, lista de tablas y la siguiente estructura:

- **Introducción.**

En esta sección se pretende elaborar un resumen general de los objetivos del proyecto, los productos a ser desarrollados, actividades y productos de trabajo principales, hitos, recursos requeridos y una agenda general del contenido y presupuesto, se debe listar los términos y referencias de los acuerdos del proyecto, se debe especificar los planes de las actualizaciones planificadas y no planificadas del plan de gestión de proyectos de software, se debe ofrecer una lista de los documentos y otras fuentes de información referenciadas en el plan de gestión de proyectos de software y se define los términos y acrónimos necesarios para comprender adecuadamente el plan.

- **Organización del Proyecto.**

Esta sección del estándar hace referencia a las relaciones entre las funciones principales del proyecto y las actividades del mismo, la estructura para la gestión interna del proyecto, los límites administrativos y de gestión entre el proyecto y las entidades de la organización y los individuos responsables de las funciones y actividades.

- Modelos de Procesos.
- Estructura Organizativa.
- Fronteras e interfaces organizativas.
- Responsabilidades.

- **Procesos de Gestión.**

Estos procesos describen la filosofía, objetivos y prioridades de las actividades de gestión del proyecto, afirman los supuestos, dependencias y restricciones en los cuales está basado el proyecto, identifican y valoran los factores de riesgos asociados al proyecto, definen mecanismos que pueden ser usados para supervisar y controlar el proyecto y especifican el número y perfil de personas requeridas para ejecutar el proyecto.

- Objetivos y prioridades de Gestión.
- Suposiciones, dependencias y restricciones.
- Gestión de Riesgos.
- Mecanismos de supervisión y control.
- Plan de Personal.

- **Procesos Técnicos**

Estos procesos especifican la tecnología, plataforma, arquitectura y metodologías de desarrollo para el proceso de ingeniería de software, contiene o referencia el plan de documentación del proyecto software y contiene los planes para las funciones de soporte para el proyecto software.

- Metodologías, Técnicas y Herramientas.
- Documentación Software.
- Funciones de Apoyo al proyecto.

- **Plan de Desarrollo.**

Este plan define los paquetes de trabajo para las tareas y actividades que deben completarse en orden para satisfacer los acuerdos del proyecto, detalla las relaciones entre los paquetes de trabajo y la dependencia de acontecimientos externos al proyecto, facilita en función del tiempo, estimaciones de los recursos necesarios para completar el plan de gestión de proyectos de software, describe la disposición de las herramientas y recursos de las diferentes funciones del proyecto, actividades y tareas, y; brinda una agenda para que las funciones, actividades y tareas del proyecto se desarrollen en un determinado orden y se ejecuten en los plazos comprometidos.

- Paquetes de Trabajo.
- Dependencias.
- Recursos.
- Presupuesto y distribución de recurso
- Calendario.

De manera opcional puede contener componentes adicionales, el índice y los apéndices. Este estándar establece el formato y contenido de los planes para la gestión de los proyectos de desarrollo de software, sin embargo no especifica las técnicas exactas que pueden ser usadas en el desarrollo de los planes de proyectos.

2.4 Antecedentes contextuales

2.4.1 Misión

La SEPS es una entidad técnica de supervisión y control que busca el desarrollo, estabilidad, solidez y correcto funcionamiento del sector económico popular y solidario, con procesos técnicos, transparentes y confiables, para contribuir al bienestar de sus integrantes y de la comunidad en general. (GOBIERNO, 2012)

2.4.2 Visión

Para el año 2017 la SEPS tiene como enfoque institucional, ser reconocida a nivel nacional e internacional como referente en supervisión y control de la Economía Popular y Solidaria, con procesos transparentes, confiables y de excelencia, que impulsan la consolidación de sus organizaciones en procura del Buen Vivir. (GOBIERNO, 2012)

2.4.3 Atribuciones

Las atribuciones que la Ley le otorga a la Superintendencia, respecto de las organizaciones de la Economía Popular y Solidaria (EPS) y del Sistema Financiero Popular y Solidario (SFPS), son:

- Ejercer el control de sus actividades económicas;
- Velar por su estabilidad, solidez y correcto funcionamiento;
- Otorgar personalidad jurídica a estas organizaciones;
- Fijar tarifarios de servicios;
- Autorizar las actividades financieras que dichas organizaciones desarrollen;
- Levantar estadísticas;
- Imponer sanciones; y,
- Expedir normas de carácter general.

Artículo 147.- Ley Orgánica de la Economía Popular y Solidaria y del Sector Financiero Popular y Solidario. Adicionalmente, el Reglamento señala las siguientes:

- Cumplir y hacer cumplir las regulaciones dictadas por los órganos competentes;
- Conocer y aprobar reformas a los estatutos de las organizaciones;
- Registrar nombramientos de directivos y representantes legales de las organizaciones;
- Registrar la adquisición o pérdida de la calidad de integrante de las organizaciones;
- Proponer regulaciones a los órganos encargados de dictarlas; y,
- Revisar informes de auditoría de las organizaciones.

Artículo 154.- Reglamento General de la Ley Orgánica de la Economía Popular y Solidaria y del Sector Financiero Popular y Solidario. (GOBIERNO, 2012).

2.4.4 Organigrama

Actualmente la SEPS se encuentra estructurada como se muestra en la siguiente figura:

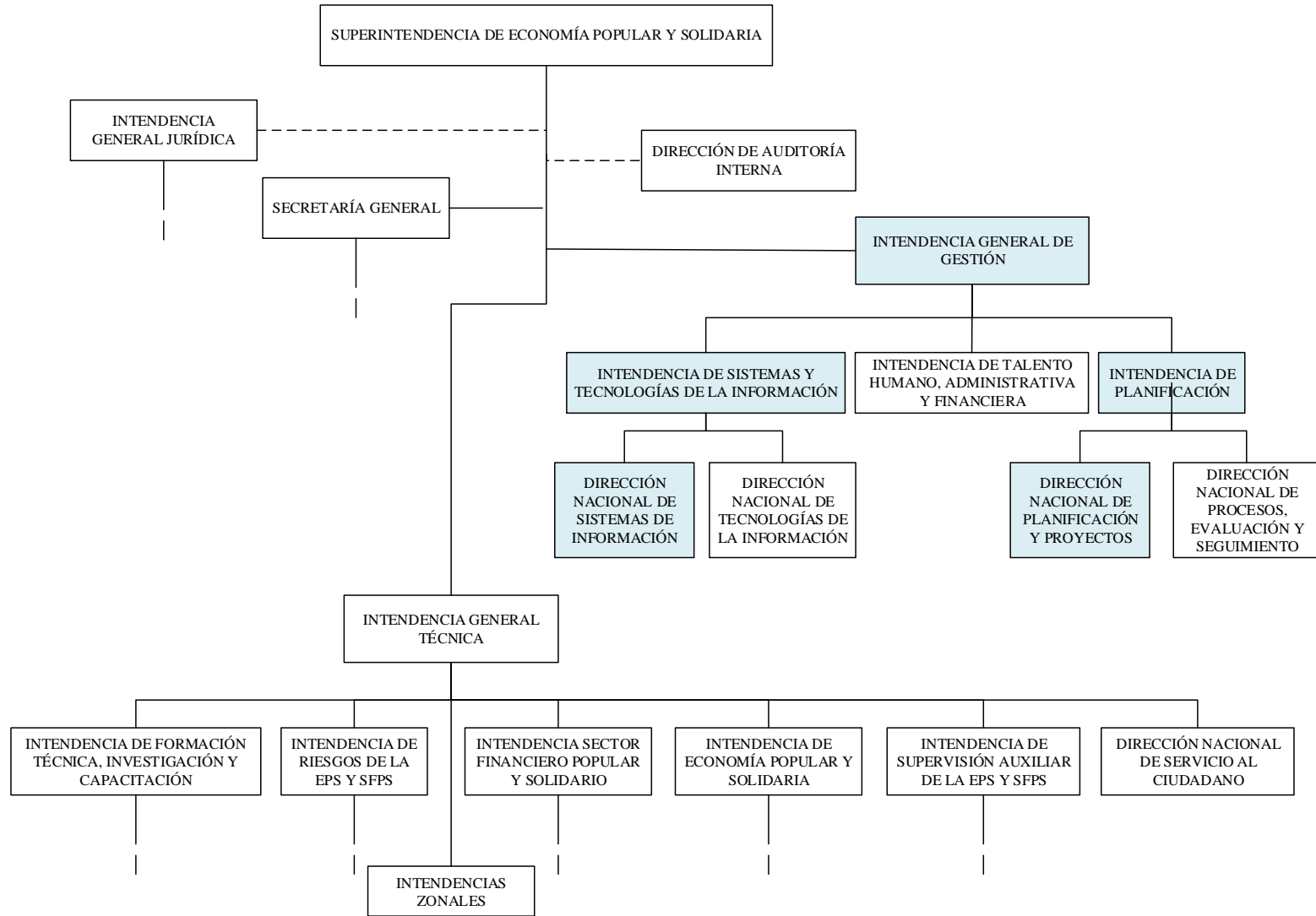


Gráfico 1. Organigrama de la Superintendencia de Economía Popular y Solidaria.

En este organigrama podemos identificar en color azul, que una de las áreas que conforman la SEPS es la Intendencia General de Gestión, la misma que se encuentra organizada en: la Intendencia de Sistemas y Tecnologías de la Información, la Intendencia de Talento Humano, Administrativa y Financiera y la Intendencia de Planificación.

Dentro de la Intendencia de Planificación se encuentra la Dirección Nacional de Planificación y Proyectos que se encarga de gestionar todos los proyectos que se desarrollan en la SEPS; y la Dirección Nacional de Procesos, Evaluación y Seguimiento que conjuntamente con las Intendencias Funcionales se encarga de definir y diagramar los procesos y procedimientos establecidos para cada proyecto.

Por su parte, la Intendencia de Sistemas y Tecnologías de la Información se encuentra organizada en la Dirección Nacional de Tecnologías de Información que se encarga de ofrecer el soporte necesario en lo referente a infraestructura, conectividad e implementación de ambientes de desarrollo tecnológico; y la Dirección Nacional de Sistemas de Información en la cual se ejecutan los proyectos de sistemas de información, incluidos los nuevos proyectos de desarrollo de software.

Con la finalidad de justificar el problema que será sujeto de investigación, se ha elaborado un instrumento (encuesta), Ver **Anexo A**, que fue aplicado a la DNSI a 14 personas distribuidos de acuerdo a los siguientes tipos de roles:

- Líderes tecnológicos (expertos).
- Arquitectos de software.
- Desarrolladores Senior.

Obteniendo los siguientes resultados:

1. Se han presentado inconvenientes en los proyectos de desarrollo de software en la Superintendencia de Economía Popular y Solidaria?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	8
NO	0
Ocasionalmente	6

Resultado: Del 100 % de las respuestas obtenidas, un 57,14 % han confirmado que se presentan inconvenientes en los proyectos de desarrollo de software y un 42,85 % ha confirmado que ocasionalmente se presentan inconvenientes en los proyectos de desarrollo de software.

2. Se ha realizado una estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	8
NO	1
Ocasionalmente	5

Resultado: Del 100 % de las personas encuestadas, las respuestas corresponden: el 57,14 % SI, el 7,14 % NO y el 35,71 % Ocasionalmente, en lo referente a si se ha realizado una estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software.

3. Se han ocasionado retrasos en la entrega de los productos de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	10

NO	2
Ocasionalmente	2

Resultado: Del 100 % de las personas encuestadas, las respuestas corresponden: el 71,42 % SI, el 14,28 % NO y el 14,28 % Ocasionalmente, en lo referente a si se ha realizado una estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software.

4. Se han presentado inconformidades de parte de las Intendencias funcionales con relación al producto entregado?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	7
NO	1
Ocasionalmente	6

Resultado: Del 100 % de las personas encuestadas, las respuestas corresponden: el 50 % SI, el 7,14 % NO y el 42,85 % Ocasionalmente, en lo referente a si se ha realizado una estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software.

5. Existen procedimientos implementados en la SEPS para realizar la planificación de los proyectos de desarrollo de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	7
NO	4
Ocasionalmente	3

Resultado: Del 100 % de las personas encuestadas, las respuestas corresponden: el 50 % SI, el 28,57 % NO y el 21,42 % Ocasionalmente, en lo referente a si se ha realizado una estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software.

6. Cree que sería necesaria la aplicación de procedimientos para la planificación de los proyectos de desarrollo de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	14
NO	0
Ocasionalmente	0

Resultado: Del 100 % de las respuestas obtenidas, en su totalidad han confirmado que se presentan inconvenientes en los proyectos de desarrollo de software.

7. Cree conveniente el uso de modelos de planificación para los proyectos de desarrollo de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	14
NO	0
Ocasionalmente	0

Resultado: Del 100% de las respuestas obtenidas, en su totalidad han confirmado que se presentan inconvenientes en los proyectos de desarrollo de software.

8. Estaría dispuesto a aplicar un modelo de planificación para los proyectos de desarrollo de software?

ALTERNATIVA RESPUESTA	CANTIDAD
SI	13
NO	0
Ocasionalmente	1

Resultado: Del 100 % de las respuestas obtenidas, un 92,85 % han confirmado que se presentan inconvenientes en los proyectos de desarrollo de software y un 7,14 % ha confirmado que ocasionalmente se presentan inconvenientes en los proyectos de desarrollo de software.

Los resultados generados a través de la encuesta, el 57,14 % confirma que existen inconvenientes en el desarrollo de los proyectos de software especialmente en el grupo de procesos de Planificación, se demuestra mediante las siguientes interrogantes: estimación poco precisa de los recursos a ser utilizados en el proyecto de desarrollo de software, retrasos en la entrega de los productos de software, no existen procedimientos para realizar la planificación de los proyectos de desarrollo de software, entre otros.

De igual manera al investigar sobre la necesidad de implementar un modelo de planificación con procedimientos que guíen el proceso de desarrollo de software se demuestra que el 100% de las personas encuestadas está de acuerdo con la implementación de un modelo de planificación para los proyectos de desarrollo de software, quedando justificado de esta manera el diseño de un modelo de planificación de proyectos de desarrollo de software en la Dirección Nacional de Sistemas de Información de la Intendencia de Sistemas y Tecnologías de la Información de la Superintendencia de Economía Popular y Solidaria.

CAPÍTULO III

DISEÑO DEL MODELO DE PLANIFICACIÓN DE PROYECTOS DE DESARROLLO DE SOFTWARE EN LA DNSI DE LA ISTI DE LA SEPS.

3.1. Introducción.

De acuerdo al capítulo anterior se ha identificado que existen inconvenientes en el desarrollo de los proyectos de software, especialmente en el grupo de procesos de Planificación ya que se han presentado estimaciones poco precisas de los recursos a ser utilizados, retrasos en la entrega de los productos de software y no se han establecido procedimientos para realizar la planificación de los proyectos de desarrollo de software. Así mismo se ha detectado la necesidad de implementar un modelo de planificación con procedimientos que guíen el proceso de desarrollo de software y de esta manera mejorar el proceso de planificación de un proyecto de desarrollo de software, ya que constituye el pilar fundamental para la ejecución, control y gestión de un proyecto de desarrollo de software ya que permite la definición de los recursos que van a ser utilizados para llevar a cabo un proyecto, en este caso de desarrollo de software. Además permite establecer las tareas, el alcance, el ámbito y el tiempo destinado para la ejecución de dichas tareas.

En este capítulo se realizará una selección de los modelos y estándares que cumplen con las condiciones necesarias para la presente investigación, se procederá a comparar dichos modelos y estándares utilizando métodos establecidos para obtener el modelo de referencia para la propuesta, finalmente se desarrolla el modelo propuesto para la planificación de proyectos de desarrollo de software de la DNSI.

3.2. Estudio comparativo de los modelos de planificación para proyectos de desarrollo de software.

3.2.1. Descripción de modelos y estándares

En referencia a los capítulos 1 y 2 del presente documento se ha identificado que los modelos o estándares que cumplen con los procesos para la planificación de un proyecto de desarrollo de software y que mejor se ajustan a la realidad institucional son los siguientes:

- **Guía de los Fundamentos para la Dirección de Proyectos (PMBOK V5).**

La Guía del PMBOK es un marco de referencia de la gestión de proyectos desarrollado por el Project Management Institute (PMI), constituye un estándar reconocido internacionalmente (ANSI/PMI 99-001-2008) que provee los fundamentos de la gestión de proyectos que son aplicables a un amplio rango de proyectos, incluyendo construcción, software, ingeniería, etc.

Contiene 5 grupos de procesos y 10 áreas de conocimiento generalmente aceptados como mejores prácticas en la gestión de proyectos. Los grupos de procesos se traslapan e interactúan a través del ciclo de vida de un proyecto o de una fase. Los grupos de procesos contienen a su vez procesos que se describen en términos de: Entradas, Herramientas y Técnicas y Salidas.

Para el presente análisis y comparación nos enfocaremos en el proceso de Planificación, las entradas, salidas, técnicas y herramientas a utilizar y su alineación con los objetivos y estrategias institucionales.

El proceso de Planificación de esta guía nos permite mantener el control de la planificación del desarrollo de software como parte de la planificación general del proyecto, en consecuencia este proceso es aplicable tanto en la planificación del ciclo de vida del proyecto como en la planificación del desarrollo del producto.

El proceso de planificación presenta una información completa y detallada en cuanto a las entradas, salidas, técnicas y herramientas para la planificación del proyecto de desarrollo de software y existe la suficiente información de acceso público.

Permite la combinación con las metodologías ágiles de desarrollo de software utilizadas actualmente en la Institución. La metodología ágil que actualmente utiliza la Institución es SCRUM.

La guía del PMBOK es utilizada como un estándar a nivel institucional para la gestión de proyecto, es por esta razón que su uso se alinea con la estrategia institucional establecida para la planificación del proyecto. En la SEPS se ha ejecutado un Plan de capacitación institucional sobre las buenas prácticas del PMBOK permitiendo de esta manera que exista un nivel de conocimiento aceptable y claridad en sus contenidos.

- **Norma ISO 21500, Directrices para la Dirección y Gestión de Proyectos.**

La norma **ISO 21500** ha sido escrita como una guía que busca orientar a las empresas en su gestión. Su objetivo es ayudar a Directores de Proyecto, principiantes o experimentados a aplicar las mejores prácticas en la gestión de sus proyectos, mejorando los resultados de negocio y concluyendo sus objetivos con éxito.

De igual manera para el presente análisis y comparación nos enfocaremos en el proceso de Planificación, las técnicas y herramientas a utilizar y su alineación con los objetivos y estrategias institucionales.

ISO 21500 proporciona una descripción detallada y muy explicativa de los conceptos y procesos que se consideran relevantes en esta línea.

El proceso de Planificación de esta guía nos permite mantener el control de la planificación del desarrollo de software como parte de la planificación general del

proyecto, sin embargo no se ajusta completamente a la alineación estratégica actualmente utilizada en la SEPS. Por otro lado este proceso es aplicable en la planificación del ciclo de vida del proyecto y no en la planificación del desarrollo del producto.

El proceso de planificación presenta una información completa y detallada en cuanto a las entradas, salidas, sin embargo en cuanto a las técnicas y herramientas para la planificación del proyecto de desarrollo de software no presenta mayor detalle, existe la suficiente información de acceso público.

- **Norma IEEE 1058-1998 para la Planificación de Gestión de Proyectos Software.**

Es un estándar IEEE que describe el formato y el contenido de los planes para la gestión de proyectos de software. No especifica técnicas exactas que puedan ser usadas en el desarrollo de los planes de proyectos. Esta norma se aplica a todo tipo de proyecto de software; la aplicación no está limitada a los proyectos que desarrollan versiones operativas de nuevos productos. La aplicación de esta norma no está limitada por el tamaño de los proyectos

Este proceso de Planificación únicamente permite mantener el control de la planificación del desarrollo de software pero no la planificación general de cualquier proyecto, es aplicable tanto en la planificación del ciclo de vida del proyecto como en la planificación del desarrollo del producto.

El proceso de planificación no presenta una información completa y detallada en cuanto a técnicas y herramientas para la planificación del proyecto de desarrollo de software y no existe la suficiente información de acceso público.

A continuación se realizará un estudio comparativo de estos modelos y estándares con la finalidad de definir cuál de ellos se ajusta para la propuesta del modelo en la presente investigación.

3.2.2. Definición de los Parámetros de Comparación

La planificación constituye la principal causa del éxito o fracaso de un proyecto de desarrollo de software, es por ello que es fundamental utilizar entradas, salidas, técnicas y herramientas que sean aplicadas de forma rigurosa sobre el proceso de gestión del proyecto de desarrollo de software para establecer los recursos a ser utilizados en el proyecto de acuerdo al alcance de sus requerimientos. La planificación del proyecto de desarrollo de software debe estar alineada directamente con la planificación general del proyecto, las estrategias, portafolio y programas institucionales.

A continuación se presentan los criterios que el modelo para la planificación de proyectos de desarrollo de software debe cumplir acorde con las necesidades institucionales y mejores prácticas que nos ofrecen los modelos seleccionados. Estos datos han sido tomados de las siguientes fuentes: “Comparison between ISO 21500 and PMBOK Guide 5th Edition” y del “Primer Congreso internacional de Gerencia de Proyectos”.

3.2.2.1. Criterios vinculados a la planificación de proyectos de software:

a) Alineación estratégica, portafolio, programa y proyecto.- La planificación del proceso de desarrollo de software debe estar acorde a la planificación general del proyecto:

- La alineación es detallada.
- La alineación no es muy clara.
- No existe alineación.

b) Área de aplicación.- El modelo se aplica en la planificación del proyecto y en la planificación del proceso de desarrollo:

- Aplicable a la planificación del ciclo de vida del proyecto y del producto.
- Aplicable únicamente a la planificación del ciclo de vida del proyecto o del producto.
- No es aplicable en la planificación del ciclo de vida del proyecto ni del producto.

c) Completitud.- Completitud con relación a las entradas, salidas, técnicas y herramientas para el proceso de planificación de los proyectos de desarrollo de software:

- Detalla las entradas, salidas, técnicas y herramientas para la planificación del proyecto de desarrollo de software.
- Menciona a detalle las entradas, salidas, técnicas y herramientas para la planificación del proyecto de desarrollo de software.
- No detalla las entradas, salidas, técnicas y herramientas para la planificación del proyecto de desarrollo de software.

d) Acceso a la información.- Existen los medios de consulta y la suficiente información para la planificación de proyectos de desarrollo de software:

- Se tiene acceso público a la información.
- Se tiene acceso limitado a la información.
- No se tiene acceso a la información.

e) Adaptabilidad con las metodologías ágiles.- Se puede combinar con las metodologías ágiles de desarrollo de software utilizadas actualmente en la Institución:

- Es adaptable.

- Tiene muchas restricciones para ser adaptado.
- No es adaptable.

3.2.2.2. Criterios institucionales para la planificación de proyectos

Además se han tomado en cuenta criterios institucionales que tienen gran influencia en la selección del modelo de planificación:

a) Alineación con los objetivos institucionales.- Estrategia institucional establecida para la planificación de los proyectos de desarrollo de software:

- Alineación estratégica, portafolio, programa y proyecto, bien detallado.
- Alineación estratégica, portafolio, programa y proyecto mencionado sin detallar.
- No menciona la alineación estratégica, portafolio, programa y proyecto.

b) Dominio de la información.- Si existe un nivel de conocimiento aceptable y claridad en sus contenidos:

- La documentación es clara.
- La documentación no es suficientemente clara.
- La documentación es confusa.

Con la finalidad de permitir la selección del modelo de planificación que mejor se adapte a las necesidades de los proyectos de desarrollo de software de la Dirección Nacional de Sistemas de Información de la Intendencia de Sistemas y Tecnologías de la Información, se ha asignado las puntuaciones: 1 para B (Bueno), 2 para M (Medio) y 3 para A (Alto). A continuación se presenta en la Tabla 3 el detalle de la puntuación asignada a los modelos y normas en base a los criterios establecidos:

Tabla 3.

Puntuación de los modelos y normas en base a los criterios establecidos

CRITERIOS	PMBOK V5			ISO 21500			IEEE 1058.1		
	A	M	B	A	M	B	A	M	B
Alineación Estratégica, Portafolio, Programa y Proyecto	3				2				N/A
Área de aplicación	3				2		3		
Compleitud	3				2			2	
Acceso a la información	3			3				2	
Adaptabilidad con metodologías ágiles	3			3			3		
Alineación con los objetivos institucionales	3				2				1
Dominio de la información	3				2				1
TOTAL	21			16			12		

De los resultados expuestos se puede concluir que la Guía del PMBOK se adapta mejor a las necesidades institucionales ya que tiene una mejor calificación comparada con la Norma ISO 21500 y la Norma IEEE 1058.1.

Esta comparación se la ha realizado aplicando el Método de Estudio de Similitud entre Modelos y Estándares (MESME) (Orna, 2013). Aplicando este método a nuestro caso de estudio, se han realizado las siguientes acciones:

1) Seleccionar estándares y modelos.- se ha tomado como referencia el cumplimiento de los siguientes criterios:

- El modelo o estándar cumple con el proceso de planificación de proyectos de desarrollo de software o puede ser adaptado al proceso de ingeniería de software.
- El modelo o estándar es reconocido a nivel mundial por las organizaciones de control del proceso de desarrollo de software.
- El modelo o estándar fue propuesto por un organismo de control a nivel mundial.
- Se tiene libre acceso a la información proporcionada sobre este modelo o estándar.

2) Seleccionar modelo de referencia:

- El modelo o estándar contiene información actualizada.
- El modelo o estándar fue emitido por una institución relevante.
- El modelo o estándar tiene un enfoque dirigido a la planificación de proyectos.
- El modelo o estándar puede ser adaptado a la planificación de los proyectos de desarrollo de software.
- El modelo o estándar es compatible con otras buenas prácticas.

3) Seleccionar el proceso a analizar:

- El modelo o estándar hace referencia a los procesos de planificación.
- El modelo o estándar presenta técnicas y herramientas para la estimación de recursos.
- Permite la identificación y evaluación de los riesgos.
- Toma en consideración el ámbito del proyecto.

4) Establecer el nivel de detalle y crear una plantilla de correspondencia.- Se ha determinado el nivel de detalle de acuerdo al cumplimiento de cada criterio según los siguientes parámetros:

- A (3) Alto grado de cumplimiento del criterio evaluado.
- M (2) Mediano grado de cumplimiento del criterio evaluado.
- B (1) Bajo grado de cumplimiento del criterio evaluado.

5) Identificar la similitud entre modelos.

Tabla 4.

Resultados comparativos

CRITERIOS	PMBO			ISO			IEEE		
	K V5			21500			1058.1		
	A	M	B	A	M	B	A	M	B
Cumple con el proceso de planificación de proyectos de desarrollo de software o puede ser adaptado al proceso de ingeniería de software.	3			3			3		
Es reconocido a nivel mundial por las organizaciones de control del proceso de desarrollo de software.	3			3			3		
Fue propuesto por un organismo de control a nivel mundial.	3			3			3		
Se tiene libre acceso a la información proporcionada sobre este modelo o estándar.	3				2			2	
Contiene información actualizada.	3				2			2	
Fue emitido por una institución relevante.	3			3			3		
Tiene un enfoque dirigido a la planificación de proyectos.	3			3				2	
Puede ser adaptado a la planificación de los proyectos de desarrollo de software.	3				3			3	
Compatible con otras buenas prácticas.	3			3			3		
Hace referencia a los procesos de planificación.	3			3			3		
Presenta técnicas y herramientas para la estimación de recursos.	3				3			3	
Permite la identificación y evaluación de los riesgos.	3			3			3		
Toma en consideración el ámbito del proyecto.	3			3			3		
TOTAL	39			37			36		

6) Presentar resultados.

Como podemos observar en la tabla 4, la Guía PMBOK cumple con todos los criterios, mientras que la norma ISO 21500 y la norma IEEE 1058.1 cumplen en un alto grado, validando los tres estándares como modelos de referencia para el presente estudio.

Seguidamente se diseña del modelo de planificación para proyectos de desarrollo de software en la DNSI de la ISTI, basado en la Guía PMBOK, norma que según el estudio realizado y presentado en este apartado se ajusta a las condiciones del contexto en el cual se basa la presente investigación.

3.3. Diseño del modelo de planificación para proyectos de desarrollo de software.

Para la gestión de los proyectos de desarrollo de software se han aplicado técnicas y herramientas utilizadas en base a la experiencia de cada líder tecnológico, esto ha generado que se obtengan resultados muy diversos en el producto final y en la entrega del sistema al área funcional, en especial difiere en la precisión de la planificación establecida al inicio del proyecto, ya que dependía de la experiencia y las estrategias que cada líder tecnológico aplique al proyecto.

Es por esta razón que, con la finalidad de mejorar la planificación de los proyectos de desarrollo de software, estandarizar y formalizar la gestión de los mismos, se realizaron varias reuniones con el equipo de trabajo, de las cuales 5 de ellas sirvieron para las siguientes actividades puntuales:

1. En la primera reunión con la participación del Director Nacional de Sistemas de Información y los líderes tecnológicos se seleccionó la Guía del PMBOK como base para la gestión de proyectos tecnológicos por ser el más completo y por ser el estándar institucional.

2. En la segunda reunión con la participación del Director Nacional de Sistemas de Información y los líderes tecnológicos se definió que para el fin propuesto de mejorar la planificación de los proyectos de desarrollo de software se centrará el análisis en el grupo de procesos de Planificación de la Guía del PMBOK.
3. En la tercera reunión con la participación del Director Nacional de Sistemas de Información, los líderes tecnológicos y los arquitectos de software se definió el proceso de desarrollo de software para los proyectos de desarrollo de software que consiste en las siguientes fases: análisis de requerimientos, diseño del sistema, construcción y pruebas tecnológicas, pruebas funcionales y despliegue.
4. En la cuarta reunión con la participación del Director Nacional de Sistemas de Información, los líderes tecnológicos y los arquitectos de software se definió que la Guía del PMBOK debía ser acoplada con el proceso de desarrollo de software definido en el punto 3.
5. En la quinta reunión con la participación del Director Nacional de Sistemas de Información, los líderes tecnológicos y los arquitectos de software se definió que se desarrollarán procedimientos que permitan controlar y dar seguimiento al proceso de desarrollo de software dentro de las áreas de conocimiento y los grupos de procesos seleccionados, los mismos que serán utilizados por los líderes tecnológicos, los arquitectos y los desarrolladores.

Con la finalidad de institucionalizar los procedimientos se realizaron reuniones de trabajo con la Intendencia de Planificación y Procesos y en los casos necesarios con las Intendencias funcionales.

Luego de varias reuniones, revisión, corrección y aprobación de dichos procedimientos se inició con la formalización y la aplicación de los mismos en un proyecto real tomado como piloto, el proyecto de Acopio Integral – Estados Financieros - Balances.

De la misma manera, en referencia a los capítulos II y III de la presente investigación, se ha elaborado la propuesta de un modelo de planificación para proyectos

de desarrollo de software en la DNSI de la ISTI, con la finalidad de abarcar los proyectos nuevos de desarrollo de software, tomando como referencia la Guía del PMBOK 5ta. EDICIÓN y los procesos de ingeniería de software. Además se han definido los procesos y áreas del PMBOK que se presenta en la plantilla “Procesos de Planificación para Proyectos de Desarrollo de Software”, en el **Anexo B**, y se ha establecido el diagrama de procesos de ingeniería de software que la Institución utiliza, el cual puede visualizarse en la plantilla “Proceso de Ingeniería de Software”, en el **Anexo C**.

El modelo propuesto se encuentra constituido por seis áreas de conocimiento distribuido en ocho procesos, cada proceso contiene entradas, herramientas y técnicas, y salidas. Las entradas son los artefactos necesarios para dar inicio a un proceso; las herramientas son los documentos que han sido elaborados por el equipo de líderes tecnológicos, arquitectos y funcionarios de la Intendencia de Planificación de la Institución con la finalidad de estandarizar y optimizar el proceso de planificación, entre estos documentos tenemos el Plan de gestión para un proyecto de ingeniería de software, los procedimientos y las plantillas de trabajo correspondientes a cada proceso; las técnicas son los diferentes mecanismos o métodos que utilizaremos como apoyo para la ejecución de cada proceso; y, por último las salidas son los artefactos como resultado de la aplicación de cada proceso.

En el gráfico 2 y tabla 5, se presenta el modelo de planificación para proyectos de desarrollo de software para la Dirección Nacional de Sistemas de Información de la Intendencia de Sistemas y Tecnologías de la Información:

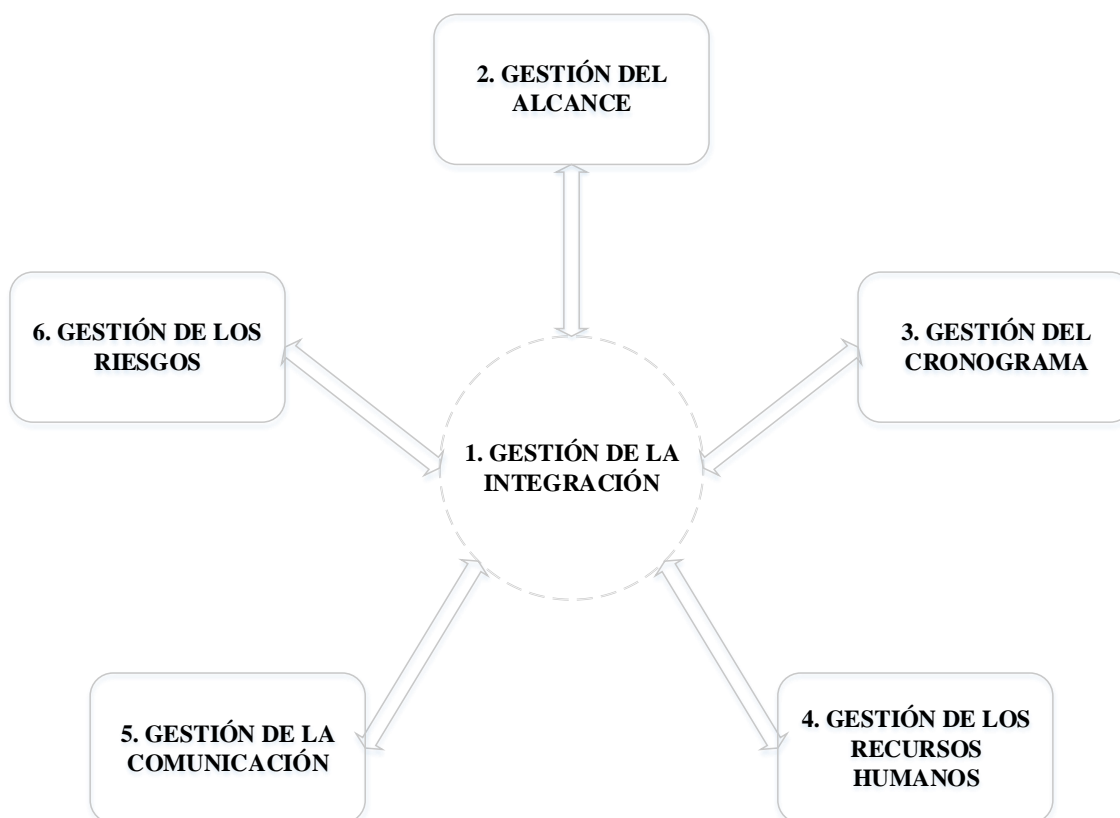


Gráfico 2. Modelo de Planificación de Proyectos de Desarrollo de Software (MPPDS)

Tabla 5.

Detalle del Modelo de Planificación de Proyectos de Desarrollo de Software (MPPDS)

ÁREA DE CONOCIMIENTO	PROCESO	PLAN / PROCEDIMIENTO
1. GESTIÓN DE LA INTEGRACIÓN	Desarrollar el Plan para la gestión del proyecto de ingeniería de software.	Plan de Gestión de un proyecto de ingeniería de software.
2. GESTIÓN DEL ALCANCE	Planificar la Gestión del Alcance.	Análisis de requerimientos con componente tecnológico.

Continúa →

3. GESTIÓN DEL CRONOGRAMA	Planificar / Actualizar la gestión del cronograma.	Planificación del desarrollo de sistemas de información.
	Registrar y Secuenciar las actividades.	Planificación del desarrollo de sistemas de información.
	Estimar los recursos y desarrollar el cronograma.	Análisis de requerimientos con componente tecnológico.
		Identificación, Evaluación y Definición de Acciones de Contingencia sobre los riesgos en proyectos con componente tecnológico.
		Diseño de componentes.
		Diseño aplicativo de base de datos.
		Distribución de componentes de software.
		Pruebas unitarias, integración y carga.
4. GESTIÓN DE LOS RECURSOS HUMANOS	Planificar la gestión de los recursos humanos.	Formato para el registro del equipo de proyecto.
5. GESTIÓN DE LA COMUNICACIÓN	Planificar la gestión de las comunicaciones.	Formato para el registro del equipo de proyecto.
6. GESTIÓN DE LOS RIESGOS	Planificar la gestión de los riesgos.	Identificación, Evaluación y Definición de Acciones de Contingencia sobre los riesgos en proyectos con componente tecnológico.

3.3.1. Gestión de la integración.

Esta área incluye los procesos que permitirán al líder tecnológico la coordinación de los elementos del proyecto, es soporte fundamental para alcanzar los requerimientos y gestionar las expectativas de los interesados del proyecto. En esta área se desarrollarán los siguientes procesos:

3.3.1.1. Desarrollar el Plan para la gestión de un proyecto de ingeniería de software.

El plan para la gestión de un proyecto de ingeniería de software define cómo el proyecto se ejecuta, monitorea, se controla y se cierra. Además en este proceso se definirán, prepararán y coordinarán todos los planes secundarios que componen el plan para la gestión de un proyecto de ingeniería de software.

Entradas

Acta de Constitución del proyecto y Plan de Gestión del proyecto.- Este documento será elaborado previo al inicio del proyecto de ingeniería de software. El objetivo de este documento es definir el inicio, los límites del proyecto y el compromiso formal de los involucrados del proyecto, se definen los siguientes contenidos:

1. Nombre del proyecto,
2. Descripción del proyecto,
3. Equipo de trabajo,
4. Unidad Ejecutora y Coejecutora,
5. Beneficios Cualitativos,
6. Beneficiarios,
7. Restricciones del Proyecto,
8. Hitos,
9. Riesgos del Proyecto,
10. Presupuesto total del proyecto.

En el **Anexo D** se muestra la plantilla del “Acta de Constitución del Proyecto y el Plan de Gestión del proyecto”.

Cronograma general del proyecto: Este documento será elaborado previo al inicio del proyecto de ingeniería de software. El objetivo de este documento es definir el cronograma general del proyecto por parte del patrocinador y los involucrados del proyecto, en el **Anexo E** se muestra la plantilla del “Cronograma General del Proyecto”.

Herramientas y Técnicas:

Plan de gestión de un proyecto de ingeniería de software.- En el **Anexo F** se presenta el “Plan de Gestión de un Proyecto de Ingeniería de Software”, para desarrollar el plan para la gestión del proyecto de ingeniería de software, en el cual se han descrito los siguientes artefactos:

- ✓ Gestión del alcance.
- ✓ Gestión del cronograma.
- ✓ Gestión de los recursos humanos.
- ✓ Gestión de las comunicaciones.
- ✓ Gestión de los riesgos.

Juicio de expertos.- Se utilizará el juicio de expertos para realizar las siguientes actividades: desarrollar los detalles técnicos y de gestión, definir los recursos y roles necesarios para ejecutar el proyecto, establecer qué documentos mantendrán un control de cambios, definir las prioridades en las tareas a ejecutar con la finalidad de asignar el trabajo adecuado en el momento adecuado.

Técnicas de facilitación.- guiarán el desarrollo del Plan para la dirección del proyecto. Se hará uso de las técnicas de tormenta de ideas, resolución de conflictos y reuniones con el equipo, con la finalidad de apoyar al equipo y ejecutar las actividades del proyecto.

Salidas

Plan de gestión de un proyecto de ingeniería de software.- en este documento se describirá el modo en que el proyecto será ejecutado, monitoreado y controlado. Además se integrarán y se consolidarán todos los planes y líneas base secundarios de los procesos de planificación como se presenta en el **Anexo F**.

El Plan de Gestión del proyecto de ingeniería de software incluirá:

- Gestión de la integración.
- Gestión del alcance.
- Gestión del cronograma.
- Gestión de los recursos humanos.
- Gestión de las comunicaciones.
- Gestión de los riesgos.

El Plan para la gestión de un proyecto de ingeniería de software será elaborado en el apartado correspondiente a la gestión de la integración y queda pendiente su elaboración en los apartados de las otras áreas de conocimiento, ya que es un documento transversal en el modelo.

3.3.2. Gestión del alcance.

La gestión del alcance del proyecto permite incluir todo el trabajo requerido para completar con éxito el proyecto, aquí se define qué se incluye y que no se incluye en el proyecto. Actualmente las Intendencias funcionales de la SEPS con la asesoría tecnológica de los líderes tecnológicos de la Intendencia de Sistemas y Tecnologías de la Información definen el alcance del proyecto, el mismo que se encuentra definido en uno de los siguientes documentos: Documento de Definición de Requerimientos (DDR), ver **Anexo G**; Manual Técnico de Estructuras de Información (MTEI), ver **Anexo H**; Manual Técnico de Formularios de Información (MTFI), ver **Anexo I**.

El DDR aprobado se constituye en la línea base del alcance del proyecto, el cual puede cambiarse únicamente mediante procedimientos formales de control de cambios y con la aprobación de todos los interesados en el proyecto. En la SEPS el control de cambios se realiza mediante la actualización del DDR y el envío a la ISTI a través de un memorando.

En esta área de conocimiento se da inicio al proyecto de ingeniería de software, siempre y cuando la(s) Intendencia(s) solicitante(s) envíe los siguientes documentos obligatorios:

- a) Acta de Constitución del proyecto y Plan de Gestión (estos dos documentos pueden estar unificados).
- b) Cronograma general del proyecto.
- c) Documento de Definición de Requerimientos (DDR) o Manual Técnico de Estructuras de Información (MTEI), Manual Técnico de Formularios de Información (MTFI).
- d) Memorando de solicitud de requerimiento.

Una vez recibidos estos documentos inicia el análisis de los requerimientos por parte del equipo tecnológico para lo cual se hará uso del procedimiento “Análisis de requerimientos con componente tecnológico”, ver **Anexo J**.

3.3.2.1. Planificar la Gestión del Alcance.

Este proceso permite subdividir los entregables y el trabajo del proyecto en componentes más pequeños y más fáciles de manejar. El objetivo principal es proporcionar una visión estructurada de lo que se debe entregar.

Entradas

Acta de Constitución y Plan de Gestión del proyecto: se refiere a los documentos que fueron definidos por el Sponsor y los involucrados del proyecto con la finalidad de formalizar y dar inicio a la ejecución del proyecto, tenga componente tecnológico o no.

Cronograma General del Proyecto: es el cronograma definido por el Sponsor y los involucrados con la finalidad de estimar de manera general todo el trabajo y recursos necesarios para el desarrollo del proyecto, tenga componente tecnológico o no.

Documento de Definición de requerimientos: Es el documento definido por el gerente del proyecto y las Intendencias funcionales, en el que consta el requerimiento funcional que se convertirá en la línea base del alcance del proyecto.

Herramientas y Técnicas

Procedimiento Análisis de requerimientos con componente tecnológico: El uso de este procedimiento permitirá realizar el análisis de los requerimientos del proyecto hasta el nivel de paquetes, como se muestra en el **Anexo J**.

Descomposición: con esta técnica se permitirá dividir el alcance y entregables del proyecto de tal manera que cualquier proyecto sea simple o complejo sea manejable. En el nivel más bajo de la EDT se definirá el paquete de trabajo para el cual se definirán las propiedades esenciales como se indica en el procedimiento “Análisis de requerimientos con componente tecnológico”. El nivel de detalle de los paquetes dependerá de la complejidad del proyecto y la descomposición será definida de acuerdo a los siguientes parámetros:

- ✓ Identificar y analizar los entregables y el trabajo relacionado. Se recomienda una revisión completa del alcance del proyecto para asegurar la consistencia entre los EDT y los requerimientos del proyecto.
- ✓ Estructurar y organizar la EDT. Se recomienda definir los entregables principales del producto y los entregables predecesores necesarios para el proyecto.
- ✓ Descomponer los entregables de la EDT en los niveles adecuados dependiendo del tamaño y complejidad del proyecto de ingeniería de software.
- ✓ Desarrollar y asignar códigos de identificación a los componentes de la EDT.

Juicio de expertos: esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica permitirá descomponer los entregables del proyecto en componentes más pequeños y manejables.

La estructura de la EDT se ha elaborado en base al enfoque descendente y utilizando las fases del ciclo de vida de un proyecto de desarrollo de software. Se representará con mínimo tres niveles de desglose de acuerdo a las siguientes directrices:

- ✓ Como primer nivel se indicará el nombre del proyecto con su respectiva versión.
- ✓ Como segundo nivel se utilizará las fases del ciclo de desarrollo de software: análisis de requerimiento, diseño del sistema, construcción del sistema., pruebas e implantación
- ✓ Como tercer nivel se registrarán los entregables del producto y del proyecto.

Salidas

Estructura de Desglose del Trabajo / Work Breakdown Structure (EDT/WBS): esta estructura es generada por el procedimiento Análisis de requerimientos con componente tecnológico, cuya finalidad es proporcionar una descripción más detallada, entendible y manejable del requerimiento funcional solicitado a través del DDR. El responsable de la coordinación y elaboración de la EDT es el líder tecnológico del proyecto.

Con el uso de las herramientas y técnicas para la creación de la EDT se ha elaborado la plantilla “EDT_PRY”, que se presenta en el **Anexo K**.

Diccionario de la EDT: este documento permitirá registrar la información detallada de los entregables y actividades de los componentes de la EDT a través del documento “Diccionario de la EDT” como se presenta en el **Anexo L**.

3.3.3. Gestión del cronograma

En este proceso se define la duración del plazo del proyecto de ingeniería de software, tomando como base el cronograma general del proyecto que fue realizado por el gerente del proyecto.

En esta área se desarrollarán los siguientes procesos:

3.3.3.1. Planificar la gestión del cronograma.

En este proceso se definen los procedimientos y toda la documentación para planificar, gestionar, ejecutar y controlar el tiempo del proyecto de ingeniería de software. Este proceso nos permite gestionar el cronograma del proyecto de ingeniería de software en el tiempo establecido. Esta planificación forma parte del Plan de Gestión de un proyecto de ingeniería de software puede ser general o detallado de acuerdo a las necesidades del proyecto de ingeniería de software.

Entradas

Plan de gestión de un proyecto de ingeniería de software.- En este documento se describirá el modo en que el proyecto será ejecutado, monitoreado y controlado. Además se integra y se consolidan todos los planes y líneas base secundarios de los procesos de planificación.

Acta de Constitución del proyecto.- Esta acta será elaborada fuera del proyecto de ingeniería de software. El objetivo de este proceso es definir el inicio y los límites del

proyecto, con este documento se logrará el compromiso formal de los involucrados del proyecto.

EDT.- esta estructura es generada por el procedimiento Análisis de requerimientos con componente tecnológico, cuya finalidad es proporcionar una descripción más detallada, entendible y manejable del requerimiento funcional solicitado a través del DDR. El responsable de la coordinación y elaboración de la EDT es el líder tecnológico del proyecto.

Herramientas y Técnicas

Procedimiento Planificación del desarrollo de sistemas de software: Se hará uso de este procedimiento para registrar/actualizar en el cronograma las actividades definidas en el diccionario de la EDT aprobada, se asignará una fecha de inicio y fin tentativa del desarrollo del proyecto de ingeniería de software. El nombre institucional de este procedimiento es Planificación del desarrollo de sistemas de información ya que puede ser utilizado para otro tipo de proyectos incluidos los de desarrollo de software, sin embargo para fines de estudio en la presente investigación se le denominará “Planificación del desarrollo de sistemas de software” y se presenta en el **Anexo M**.

Juicio de expertos.- esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica permitirá además, decidir entre la combinación o no de métodos de estimación y conciliación de diferencias entre los mismos.

Técnicas analíticas.- Se utilizará como base del cronograma el modelo de ingeniería de software de la SEPS “Plantilla Cronograma Ingeniería de Software” Ver **Anexo N**, incluye el proceso de desarrollo de software y del proyecto. Se recomienda técnicas de ejecución rápida como realizar tareas en paralelo tomando en cuenta que no

afecte al proyecto o a incrementar los riesgos. Se aplicará el procedimiento “Planificación del desarrollo de sistemas de software” como se presenta en el **Anexo M**.

Reuniones.- Se pueden realizar reuniones de trabajo con el equipo tecnológico, el comité de arquitectura y con quienes sea necesario para aclarar los requerimientos y optimizar el cronograma.

Salidas

Plan de gestión del cronograma.- Se realiza en base al procedimiento “Planificación del desarrollo de sistemas de software”, ver **Anexo F**. En este plan se registrarán los siguientes parámetros:

- ✓ Tareas
- ✓ Duración en horas
- ✓ Fecha inicio
- ✓ Fecha fin
- ✓ Secuencia
- ✓ Nombre de recurso

3.3.3.2. Registrar y Secuenciar las actividades

En este proceso se registran en el cronograma del proyecto de ingeniería de software, las actividades definidas en el Diccionario de la EDT descrita en el proceso Planificar la gestión del alcance. Se les asigna una prioridad y la secuencia en la que deben ser registradas. Dependiendo de la necesidad se puede detallar el cronograma a nivel de tareas.

Entradas

Plan de gestión del cronograma.- Este documento se encuentra descrito como salida en el proceso Planificar la gestión del cronograma.

EDT.- Este documento se encuentra descrito como salida en el proceso Planificar la gestión del alcance.

Herramientas y Técnicas

Procedimiento Planificación del desarrollo de sistemas de software: Se hará uso de este procedimiento para secuenciar en el cronograma las actividades definidas en el diccionario de la EDT aprobada y se identificarán los roles para la ejecución de estas actividades, el procedimiento “Planificación del desarrollo de sistemas de software” se presenta en el **Anexo M**.

Descomposición.- Esta técnica permite dividir el alcance y entregables del proyecto como actividades o como tareas, de tal manera que cualquier proyecto sea simple o complejo sea manejable. El Diccionario de la EDT es la base para el desarrollo de la lista final de actividades o tareas. Se hace necesaria la intervención de los miembros del equipo del proyecto que contribuyan a obtener resultados mejores y más precisos.

Planificación gradual.- Esta técnica permite planificar la ejecución de las actividades o tareas a corto plazo y es un proceso iterativo. Dependiendo del ciclo de vida del proyecto las tareas pueden ser descritas en diferentes niveles de detalle. De acuerdo al avance del proyecto se puede ir detallando los próximos eventos.

Juicio de expertos.- esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica permitirá que los expertos con experiencia y habilidad tanto en el negocio

como en el desarrollo y la arquitectura, puedan aportar experiencia en la secuenciación, priorización y detalle de las tareas que deben ser ejecutadas.

Salidas

Lista de actividades.- En esta lista se incluye las actividades definidas en el Diccionario de la EDT, sin embargo se puede detallar las tareas correspondientes al nivel que sea necesario en la etapa del proyecto en el que se encuentre, pueden estar organizadas en actividades generales del ciclo de vida de desarrollo de software: análisis y levantamiento de requerimientos, diseño del sistema, construcción y pruebas, pruebas funcionales y despliegue. Cada actividad / tarea debe tener un nombre único que la identifique dentro del cronograma y que represente el trabajo que se debe realizar. Las actividades serán registradas en el **Anexo N**, “Plantilla Cronograma Ingeniería de Software”.

Atributos de las actividades.- Las actividades / tareas deberán tener definidos los siguientes atributos: identificador, modo de tarea, nombre de la tarea o actividad, duración, fechas de inicio y fin, secuencia, persona asignada para realizar el trabajo y porcentaje de avance. Los atributos de duración en horas, fechas de inicio y fin, secuencia y persona asignada para realizar el trabajo pueden sufrir cambios de forma, se pueden añadir o detallar a mayor nivel el trabajo, dependiendo de las decisiones del líder técnico. Los atributos de las actividades serán completados en el **Anexo N**, “Plantilla Cronograma Ingeniería de Software”.

Lista de hitos.- En el cronograma se establecerán los hitos que deben ser cumplidos dentro del proyecto de ingeniería de software, los mismos serán obligatorios e indican el cumplimiento del avance del trabajo. Los hitos deberán tener los mismos atributos que las actividades / tareas pero con una duración nula. Los hitos serán registrados en el **Anexo N**, “Plantilla Cronograma Ingeniería de Software”.

3.3.3.3. Estimar los recursos y desarrollar el cronograma

En este proceso se estimarán los recursos humanos y de tiempo que van a ser utilizados para ejecutar las actividades establecidas en el cronograma. La estimación será elaborada de manera iterativa y se logrará definir de manera más precisa el tiempo de duración del trabajo y por consiguiente del proyecto de ingeniería de software. Se analizarán todos los atributos de las actividades y se incorporarán en la herramienta del cronograma.

Entradas

Plan de gestión del cronograma.- Este documento se encuentra descrito como salida en el proceso Planificar la gestión del cronograma.

EDT.- Este documento se encuentra descrito como salida en el proceso Planificar la gestión del alcance.

Lista de actividades.- Este documento se encuentra descrito como salida en el proceso Registrar y Secuenciar las actividades.

Atributos de las actividades.- Este documento se encuentra descrito como salida en el proceso Registrar y Secuenciar las actividades.

Recursos requeridos para las actividades.- En esta matriz se presenta los perfiles de usuario requeridos para el trabajo que debe ser realizado para la ejecución de las tareas de acuerdo al proyecto.

Herramientas y Técnicas

Procedimiento Planificación del desarrollo de sistemas de información: Se hará uso de este procedimiento para secuenciar en el cronograma las actividades definidas en el diccionario de la EDT aprobada y se identificarán roles para la ejecución de estas actividades. En el **Anexo M** se muestra el documento de este procedimiento

Procedimiento Identificación, Evaluación y Definición de Acciones de Contingencia sobre los riesgos en proyectos con componente tecnológico.- Este procedimiento ayudará a estimar los tiempos conforme los riesgos que se podían presentar en el proyecto, el objetivo principal será mitigar los riesgos y no permitir que se materialicen. En el **Anexo O** se muestra el documento de este procedimiento.

Procedimiento Diseño de componentes.- Con este procedimiento se logrará estimar de manera precisa las tareas detalladas a realizar ya que se identificarán exactamente los componentes necesarios para el proyecto tanto a nivel de base de datos, de arquitectura y de sistema. En el **Anexo P** se muestra el documento de este procedimiento.

Procedimiento Diseño aplicativo de base de datos.- Con este procedimiento se logró estimar de manera precisa las tareas detalladas para elaborar el modelo relacional de base de datos. En el **Anexo Q** se muestra el documento de este procedimiento.

Procedimiento Distribución de componentes de software.- Con este procedimiento se logrará estimar de manera precisa las tareas detalladas que se ejecutarán por parte de los arquitectos y desarrolladores haciendo uso de las diferentes bondades que presta la orientación a objetos como es la reutilización de componentes. En el **Anexo R** se muestra el documento de este procedimiento.

Procedimiento Registro de cambios del código fuente de los aplicativos internos.- Este procedimiento permitirá optimizar el tiempo de desarrollo y la organización de los equipos de desarrollo de los diferentes proyectos y de esta manera no interfieran los

tiempos entre los equipos de desarrollo. En el **Anexo S** se muestra el documento de este procedimiento.

Procedimiento Gestión de Pruebas Unitarias, Integración y Carga.- Este procedimiento permitirá estimar de manera precisa las pruebas necesarias para ejecutarlas y el tiempo para cada una de ellas, de tal manera que se garantice la calidad del producto final. En el **Anexo T** se muestra el documento de este procedimiento.

Juicio de expertos.- esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica con el apoyo de información histórica en el caso de existir permite la generación de información de la duración máxima y mínima recomendada que puede provenir de proyectos similares. Este juicio permite decidir si es conveniente combinar métodos de estimación o no.

Estimación análoga.- esta técnica permitirá utilizar parámetros de un proyecto similar ejecutado, como base para estimar los mismos parámetros definidos en dicho proyecto anterior, por lo general será utilizada para estimar la duración de las actividades / tareas. Es necesario tomar en cuenta que los tiempos no deben ser exactos ya que diferentes factores pueden influir. Puede ser utilizada en parte del proyecto o en su totalidad, o combinada con otras técnicas.

Técnicas grupales de toma de decisiones.- Esta técnica involucra y genera compromiso en los miembros del equipo, permitirá obtener información adicional y una estimación más precisa sobre el proyecto de ingeniería de software.

Salidas

Estimación de la duración de las actividades.- es el valor cuantitativo que se le asigna a una actividad o tarea, la duración de estas estimaciones no incluyen retrasos

pero si pueden incluir un rango de resultados posibles, por ejemplo: 10 días +- 2 días, significa que la duración será entre 8 y 12 días. La duración de las actividades será registrada en el **Anexo N**, “Plantilla Cronograma Ingeniería de Software”.

Desarrollo del cronograma del proyecto de ingeniería de software.- se refiere al cronograma presentado en el **Anexo N** completo con las actividades, hitos, duración, fechas de inicio y finalización por cada actividad, orden de secuencia y responsables. El cronograma puede mantenerse como preliminar hasta que se confirme la asignación de recursos

Actualizaciones al plan para la gestión del proyecto.- Luego de desarrollar el cronograma y de acuerdo a la necesidad del proyecto de ingeniería de software si es necesario se deberá actualizar el plan para la gestión del proyecto de ingeniería de software presentado en el **Anexo F**.

Actualizaciones a los documentos del proyecto.- Entre los documentos que deben ser actualizados tenemos: recursos requeridos para las actividades, atributos de las actividades, calendarios, entre otros.

3.3.4. Gestión de los recursos humanos.

3.3.4.1. Planificar la gestión de los recursos humanos

En este proceso se identificarán y documentarán los roles y las responsabilidades dentro del proyecto de ingeniería de software.

Entradas

Plan para la gestión del proyecto.- en este documento se describirá el modo en el que el proyecto será ejecutado, monitoreado y controlado. Además se integra y se consolidan todos los planes y líneas base secundarios de los procesos de planificación.

Recursos requeridos para las actividades.- las personas asignadas para la ejecución de las tareas tienen un efecto sobre la duración del trabajo asignado, dependiendo del porcentaje de asignación y del conocimiento requerido para realizar el trabajo.

Herramientas y Técnicas

Registro de equipo de proyecto.- Este documento permite el registro de los involucrados en el proyecto de desarrollo de software, así como los datos para la gestión de las comunicaciones, Ver **Anexo U**, registra los siguientes parámetros:

1. Número de recurso.
2. Apellidos y nombres.
3. Función/Rol en el proyecto.
4. Departamento.
5. Correo electrónico.
6. Número de teléfono.

Juicio de expertos.- esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica con el apoyo de información histórica en el caso de existir permite la generación de información de la duración máxima y mínima recomendada que puede provenir de proyectos similares. Este juicio permite decidir si es conveniente combinar métodos de estimación o no.

Reuniones.- se pueden realizar reuniones de trabajo con el equipo tecnológico, el comité de arquitectura y con quienes sea necesario para aclarar los requerimientos y optimizar el cronograma.

Salidas

Plan para la gestión de los recursos humanos.- este plan forma parte del plan para la gestión de un proyecto de ingeniería de software, define el modo en el que se debería definir, dirigir y asignar los recursos humanos del proyecto. En esta etapa del proceso se añade la gestión de los recursos humanos al “Plan de Gestión de un Proyecto de Ingeniería de Software” que se presenta en el **Anexo F**.

Registro de equipo de proyecto.- Este documento permite el registro de los involucrados en el proyecto de desarrollo de software, así como los datos para la gestión de las comunicaciones, como se presenta en el **Anexo U**, registra los siguientes parámetros:

- Número de recurso.
- Apellidos y nombres.
- Función/Rol en el proyecto.
- Departamento.
- Correo electrónico.
- Número de teléfono.

3.3.5. Gestión de la comunicación

En este proceso se establecen los lineamientos para que la planificación, distribución, gestión, control y disposición de la información sean oportunos y adecuados a todos los miembros del equipo del proyecto de ingeniería de software.

3.3.5.1. Planificar la gestión de las comunicaciones

En este proceso se establecerá los medios adecuados para las comunicaciones entre el equipo del proyecto en base a las necesidades de información de los interesados.

Entradas

Plan para la gestión del proyecto.- en este documento se describirá el modo en el que el proyecto será ejecutado, monitoreado y controlado. Además se integra y se consolidan todos los planes y líneas base secundarios de los procesos de planificación.

Herramientas y Técnicas

Registro de equipo de proyecto.- Este documento permite el registro de los involucrados en el proyecto de desarrollo de software, así como los datos para la gestión de las comunicaciones, Ver **Anexo U**, registra los siguientes parámetros:

- Número de recurso.
- Apellidos y nombres.
- Función/Rol en el proyecto.
- Departamento.
- Correo electrónico.
- Número de teléfono.

Reuniones.- Se pueden realizar reuniones de trabajo con el equipo del proyecto y con quienes sea necesario para establecer medios de comunicación ágiles pero que contengan cierto grado de formalidad.

Salidas

Plan de gestión de las comunicaciones.- describe la forma en la que se planificarán, ejecutarán, controlarán y monitorearán las comunicaciones del proyecto, forma parte del plan para la gestión del proyecto. En esta etapa del proceso se añade la gestión de las comunicaciones al “Plan de Gestión de un Proyecto de Ingeniería de Software” que se presenta en el **Anexo F**.

Registro de equipo de proyecto.- Este documento permite el registro de los involucrados en el proyecto de desarrollo de software, así como los datos para la gestión de las comunicaciones, como se presenta en el **Anexo U**, registra los siguientes parámetros:

7. Número de recurso.
8. Apellidos y nombres.
9. Función/Rol en el proyecto.
10. Departamento.
11. Correo electrónico.
- Número de teléfono.

3.3.6. Gestión del riesgo

Gestionar el riesgo es identificar, evaluar y definir acciones de contingencia para los riesgos que se presenten en la ejecución de un proyecto.

3.3.6.1. Planificar la gestión del riesgo

El proceso de planificar la gestión del riesgo permite que se dé igual importancia a la gestión del riesgo como al proyecto. Dependiendo del tipo de riesgo y de su materialización para que afecte a un proyecto al punto de cerrar el mismo.

Entradas

Plan para la gestión del proyecto.- en este documento se describirá el modo en el que el proyecto será ejecutado, monitoreado y controlado. Además se integra y se consolidan todos los planes y líneas base secundarios de los procesos de planificación.

Herramientas y Técnicas

Procedimiento Identificación, Evaluación y Definición de Acciones de Contingencia sobre los Riesgos en Proyectos con Componente Tecnológico.- Este procedimiento permitirá la identificación de los riesgos, se evalúa las causas, probabilidades e impacto y se definen acciones de contingencia hasta la elaboración y socialización del informe técnico de riesgos. En el **Anexo V** se muestra el documento de este procedimiento.

Juicio de Expertos.- esta técnica se basa en el conocimiento, la capacitación y la experiencia en proyectos o áreas de negocio similares, en la SEPS se cuenta con un grupo de líderes tecnológicos que han ejecutado proyectos en áreas de negocio similares. Esta técnica permitirá identificar, evaluar y establecer planes de contingencia para mitigar o transferir el riesgo.

Reuniones.- Se pueden realizar reuniones de trabajo con el equipo tecnológico, el comité de arquitectura, los gerentes, los interesados en el proyecto y con quienes sea necesario para identificar, evaluar y establecer planes de contingencia para mitigar o transferir el riesgo.

Salidas

Plan de gestión de riesgos.- describe el modo en que se identificará, evaluará y establecerá planes de contingencia para ejecutar las actividades de gestión de riesgos, forma parte del plan de gestión del proyecto, forma parte del plan para la gestión del proyecto. En esta etapa del proceso se añade la gestión de riesgos al “Plan de Gestión de un Proyecto de Ingeniería de Software” que se presenta en el **Anexo F**.

CAPÍTULO IV

APLICACIÓN DEL MODELO DE PLANIFICACIÓN EN UN PROYECTO INSTITUCIONAL.

4.1 Introducción

En este capítulo se presenta la aplicación del modelo de planificación para proyectos de desarrollo de software propuesto para la DNSI de la ISTI, en la etapa de planificación del sistema Acopio – Balances. Para cada proceso se detallan las entradas, herramientas, técnicas y salidas con sus correspondientes entregables.

Previo a la fase de Planificación, los proyectos ejecutan la fase de Inicio, en la cual se definieron los siguientes documentos:

1. Acta de Constitución.
2. Plan de gestión.
3. Cronograma general del proyecto.

Una vez finalizada la fase de Inicio se continúa con la fase de Planificación en la cual se implementará el modelo de planificación descrito a continuación:

4.2. Implementación del modelo de planificación para proyectos de desarrollo de software en la DNSI de la SEPS: Sistema Acopio de Balances.

Para la implementación del modelo de planificación para proyectos de desarrollo de software en el sistema Acopio Balances se ha utilizado el Modelo para la Planificación de Proyectos de Desarrollo de Software (MPPDS) presentado en el gráfico 2.

4.2.1. Gestión de la integración.

4.2.1.1. Desarrollar el Plan para la gestión del proyecto de ingeniería de software.

Entradas.- para desarrollar el plan para la gestión del proyecto de ingeniería de software como entradas se recibieron el Acta de Constitución y el Plan de Gestión del Proyecto, que se presentan en el **Anexo D1**, y el Cronograma General del Proyecto Acopio Integral que se presenta en el **Anexo E1**. Estos documentos fueron definidos por el patrocinador y las Intendencias involucradas en el proyecto Acopio integral.

Herramientas y Técnicas.- para desarrollar el plan para la gestión del proyecto de ingeniería de software se ha elaborado el documento Plan de gestión de un proyecto de ingeniería de software el cual se presenta en el **Anexo F1**, además de ha utilizado la técnica de juicio de expertos y la técnica de facilitación.

Salidas: En el **Anexo F1** se presenta el Plan de gestión de un proyecto de ingeniería de software para el caso de estudio citado.

4.2.2. Gestión del alcance.

4.2.2.1. Planificar la Gestión del Alcance.

Entradas: las entradas para planificar la gestión del alcance fueron el *Acta de Constitución y el Plan de gestión del proyecto*, y el *Cronograma general del proyecto “Sistema de Acopio integral”*, descritos en el proceso anterior Desarrollar el plan para la gestión del proyecto de ingeniería de software. De la misma manera la salida del proceso anterior, el *Plan de gestión de un proyecto de ingeniería de software*, se constituye en una entrada para este proceso.

Manual Técnico de Estructuras de información (MTEI): En este documento se encuentra el requerimiento funcional sobre la estructura de Balances, elaborado por la

Intendencia de Estudios, Estadísticas y Normas en coordinación con las Intendencias involucradas, esta entrada se presenta en el **Anexo H1** para el caso de estudio citado.

Herramientas: La herramienta utilizada para planificar la gestión del alcance es el procedimiento *Análisis de requerimientos con componente tecnológico*, como se presenta en el **Anexo J** para el caso de estudio citado, se hizo uso de este procedimiento para realizar el análisis de los requerimientos de las estructuras de datos de Balances y generar la EDT con su diccionario. La técnica que se utilizó fue de *Descomposición*, con la cual se ha dividido el alcance, entregables y paquetes de trabajo del sistema Acopio Balances.

Salidas: se ha actualizado el plan para la gestión del proyecto de ingeniería de software con la *Gestión del alcance* para el sistema Acopio Integral y se ha generado la *Estructura de Desglose del Trabajo (EDT)* y el *Diccionario de la EDT* para el sistema Acopio Balances, tal como se presenta en el **Anexo K1** y **Anexo L1** respectivamente para el caso de estudio citado.

4.2.3. Gestión del cronograma

4.2.3.1. Planificar la gestión del cronograma.

Entradas: las entradas para planificar la gestión del cronograma fueron el *Acta de Constitución* y el *Plan de gestión del proyecto*, el *Cronograma general del proyecto*, el *Plan de gestión del proyecto de ingeniería de software* del sistema de Acopio integral, descritos en el proceso Desarrollar el plan para la gestión de un proyecto de ingeniería de software, además la *EDT* y el *Diccionario de la EDT*, descritos en el proceso anterior, Planificar la gestión del alcance.

Herramientas y Técnicas: La herramienta utilizada para planificar la gestión del cronograma es el procedimiento *Planificación del desarrollo de sistemas de software* que se presenta en el **Anexo M** para el caso de estudio citado. Las técnicas que se

utilizaron fueron: juicio de expertos, técnicas analíticas y reuniones con el equipo del proyecto.

Salidas: se ha actualizado el plan para la gestión del proyecto de ingeniería de software con la *Gestión del cronograma* para el sistema Acopio Integral, tal como se presenta en el **Anexo F1** para el caso de estudio citado.

4.2.3.2. Registrar y Secuenciar las actividades

Entradas: las entradas para registrar y secuenciar las actividades del proyecto fueron la *EDT*, descrita en el proceso Planificar la gestión del alcance y el *Plan de gestión del cronograma*, descrito en el proceso Planificar la gestión del cronograma.

Herramientas y Técnicas: La herramienta utilizada para planificar la gestión del cronograma es el procedimiento *Planificación del desarrollo de sistemas de software* que se presenta en el **Anexo M** para el caso de estudio citado. Las técnicas que se utilizaron fueron: descomposición, planificación gradual y el juicio de expertos.

Salidas: se ha elaborado la *Lista de actividades*, los *Atributos de las actividades* y la *Lista de hitos*, para el sistema Acopio Integral – Estados Financieros, tal como se presenta en el **Anexo N1** para el caso de estudio citado.

4.2.3.3. Estimar los recursos y desarrollar el cronograma

Entradas: las entradas para registrar y secuenciar las actividades del proyecto fueron el *Plan de gestión del cronograma*, descrito en el proceso Planificar la gestión del cronograma; la *Lista de actividades*, los *Atributos de las actividades*, descritos en el proceso Registrar y Secuenciar las actividades, la *EDT*, descrita en el proceso Planificar la gestión del alcance y los *Recursos requeridos para las actividades*, que se presenta en el Anexo 26 para el caso de estudio citado.

Herramientas y Técnicas: Las herramientas utilizadas para estimar los recursos y desarrollar el cronograma fueron los procedimientos: *Planificación del desarrollo de sistemas de información* que se presenta en el **Anexo M**, *Identificación, Evaluación y Definición de Acciones de Contingencia sobre los riesgos en proyectos con componente tecnológico* que se presenta en el **Anexo O**, *Diseño de componentes* que se presenta en el **Anexo P**, *Diseño aplicativo de base de datos* que se presenta en el **Anexo Q**, *Distribución de componentes de software* que se presenta en el **Anexo R**, *Registro de cambios del código fuente de los aplicativos internos* que se presenta en el **Anexo S** y *Gestión de Pruebas Unitarias, Integración y Carga* que se presenta en el **Anexo T**. Las técnicas que se utilizaron fueron: *juicio de expertos, estimación análoga y técnicas grupales de toma de decisiones*.

Salidas: se ha elaborado el cronograma del sistema Acopio Integral – Estados Financieros, como se presenta en el **Anexo N1** para el caso de estudio citado. En cuanto a la *Actualizaciones al plan para la gestión del proyecto*, no fue necesario realizar actualizaciones al documento.

4.2.4. Gestión de los recursos humanos.

4.2.4.1. Planificar la gestión de los recursos humanos

Entradas: las entradas para planificar la gestión de los recursos humanos fueron el *Plan de gestión de ingeniería de software*, descrito en el proceso Planificar la gestión del proyecto de ingeniería de software y los *Recursos requeridos para las actividades*, descrito en el proceso Estimar los recursos y desarrollar el cronograma.

Herramientas y Técnicas: La herramienta utilizada para planificar la gestión de los recursos humanos fue *Registro de equipo de proyecto*, que se presenta en el **Anexo U**. Las técnicas que se utilizaron fueron: *organigramas, juicio de expertos y reuniones con el equipo del proyecto*.

Salidas: se ha actualizado el plan para la gestión del proyecto de ingeniería de software con la *Gestión de los recursos humanos*, descrito en el proceso Desarrollar el plan para la gestión del proyecto de ingeniería de software para el sistema Acopio integral como se presenta en el **Anexo F1** para el caso de estudio citado y se ha elaborado el *Registro de equipo de proyecto*, como se presenta en el **Anexo U1** para el caso de estudio citado.

4.2.5. Gestión de la comunicación

4.2.5.1. Planificar la gestión de las comunicaciones

Entradas: las entradas para planificar la gestión de las comunicaciones fueron el *Plan de gestión del proyecto de ingeniería de software*, descrito en el proceso Planificar la gestión del proyecto de ingeniería de software y el *Registro de equipo de proyecto*, descrito en el proceso Planificar la gestión de los recursos humanos.

Herramientas y Técnicas: La herramienta utilizada para planificar la gestión de las comunicaciones fue el *Registro de equipo de proyecto* que se presenta en el **Anexo U**. La técnica que se utilizó fue las *reuniones* con el equipo del proyecto.

Salidas: se ha actualizado el plan para la gestión del proyecto de ingeniería de software con la *Gestión de la comunicación*, descrito en el proceso Desarrollar el plan para la gestión del proyecto de ingeniería de software para el sistema de Acopio integral como se presenta en el **Anexo F1** para el caso de estudio citado y se ha elaborado el *Registro de equipo de proyecto*, descrito en el proceso Planificar la gestión de las comunicaciones.

4.2.6. Gestión del riesgo

4.2.6.1. Planificar la gestión del riesgo

Entradas: la entrada para planificar la gestión de los riesgos fue el *Plan de gestión del proyecto de ingeniería de software*, descrito en el proceso Planificar la gestión del proyecto de ingeniería de software.

Herramientas y Técnicas: La herramienta utilizada para planificar la gestión del riesgo es el procedimiento *Identificación, evaluación y definición de acciones de contingencia sobre los riesgos en proyectos con componente tecnológico* que se muestra en el **Anexo V**. Las técnicas que se utilizaron fueron: juicio de expertos y reuniones con el equipo del proyecto.

Salidas: se ha actualizado el *Plan de Gestión de un Proyecto de Ingeniería de Software* con la *Gestión del riesgo*, descrito en el proceso Desarrollar el plan para la gestión del proyecto de ingeniería de software para el sistema Acopio integral tal como se presenta en el **Anexo F1** para el caso de estudios citado.

CAPÍTULO V

PRESENTACIÓN DE RESULTADOS

1.1 Introducción

En este capítulo se presentará el resultado de la aplicación del modelo de planificación para proyectos de desarrollo de software sobre el sistema Acopio – Balances, para ello se ha utilizado el instrumento elaborado (encuesta), el mismo que nos permitirá evaluar los objetivos planteados en la presente investigación.

La encuesta ha sido aplicada a dos grupos de personas distribuidos de la siguiente manera:

- ✓ Equipo del proyecto Acopio integral (líder tecnológico, arquitecto y desarrolladores).
- ✓ Equipo de líderes tecnológicos de la SEPS.

1.2 Procesamiento de los resultados

Para la recolección de datos y la valoración de los resultados se ha definido la siguiente estructura:

1. La encuesta contiene veinte y tres preguntas cerradas y una abierta para conocer el criterio de quienes utilizarán el modelo propuesto.
2. Por cada pregunta se seleccionará una valoración entre 1 y 5 para cada pregunta:

✓ Nunca (en ninguna ocasión)	1
✓ Raramente (en raras ocasiones)	2
✓ Ocasionalmente (en ocasiones puntuales)	3
✓ Frecuentemente (en la mayor parte de las ocasiones)	4
✓ Siempre (en todas las ocasiones)	5

3. Para seleccionar la respuesta de cada pregunta se deberá marcar con una X en la valoración correspondiente a la descripción, de acuerdo al punto anterior.

1	2	3	4	5
---	---	---	---	---

4. Se deberá sumar el número de X marcadas por cada columna de valoración.
5. Se deberá multiplicar este número por el valor de la columna respectiva.
6. Se deberá sumar el total de cada columna y dividirlos para 1,6.

Para calificar al modelo propuesto y poder interpretar los resultados obtenidos del análisis, se ha clasificado a los resultados en cuatro grupos:

Menos del 25 por ciento: El modelo propuesto no cumple con la optimización del proceso de planificación de proyectos de desarrollo de software, razón por la cual no se debe implementar el modelo en la Institución.

Entre 25 y 49 por ciento: El modelo propuesto cumple con la optimización del proceso de planificación de proyectos de desarrollo de software, sin embargo presenta deficiencias en su aplicabilidad y resultados.

Entre 50 y 74 por ciento: El modelo propuesto se cumple con la optimización del proceso de planificación de proyectos de desarrollo de software, sin embargo presenta algunos inconvenientes en su aplicabilidad y resultados.

Más de 75 por ciento: El modelo propuesto cumple a satisfacción con la optimización del proceso de planificación de proyectos de software y es factible de ser aplicado a todos los proyectos de desarrollo de software de la Institución.

El cuestionario aplicado valora parámetros de optimización del proceso de planificación y factibilidad de aplicación, bajo estos parámetros se ha realizado el presente análisis:

1.2.1 Análisis y resultados de la aplicación del modelo a los desarrolladores:

En esta sección se realizará el análisis e interpretación de los resultados generados por los desarrolladores del equipo del sistema Acopio integral, a quienes se les aplicó la encuesta que se encuentra en el **Anexo W**.

El cálculo se ha realizado ejecutando la suma total de las valoraciones obtenidas y dividiendo este valor para 2,4 ya que se ha multiplicado el valor de dos personas por el factor 1,2. En la Tabla 5 y en el Gráfico 3 se presentan los resultados:

Tabla 6.

Resultados de aceptación de los desarrolladores.

Valoración	1	2	3	4	5
Total de respuestas	0	0	0	16	30
Resultado	0	0	0	64	150
TOTAL	214				
PORCENTAJE DE ACEPTACIÓN (ST/2,4)	89,17				

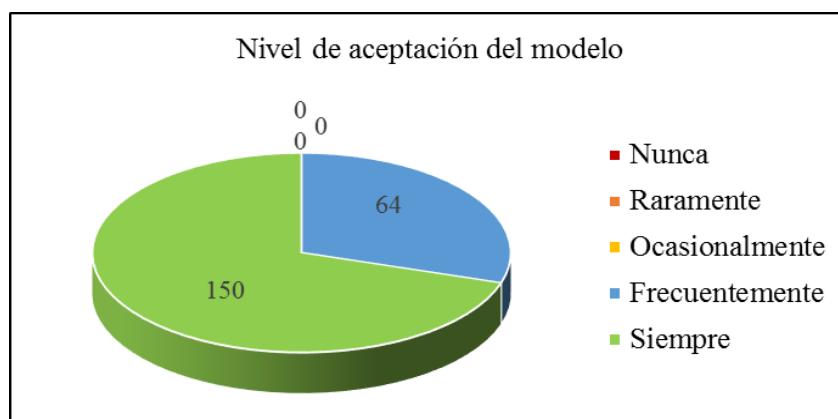


Gráfico 3. Nivel de aceptación de los desarrolladores sobre el modelo MPPDS

El gráfico anterior demuestra que en base a los resultados generados en las encuestas aplicadas a los desarrolladores, el modelo propuesto (MPPDS) presenta un 89,17 % de aceptación ya que consideran que el modelo propuesto cumple a satisfacción con la optimización del proceso de planificación de proyectos de software y es factible de ser aplicado a todos los proyectos de desarrollo de software de la Institución, con lo cual se concluye que el modelo es utilizado en el proceso de planificación de proyectos de desarrollo de software por parte de los desarrolladores.

1.2.2 Análisis y resultados de la aplicación del modelo al arquitecto de software:

En esta sección se realizará el análisis e interpretación de los resultados generados por el arquitecto de software de la DNSI de la ISTI, a quien se le aplicó la encuesta que se encuentra en el **Anexo W**.

El cálculo se realizó ejecutando la suma total de las valoraciones obtenidas y dividiendo este valor para 1,2, ya que se trata de una persona. En la Tabla 6 y en el Gráfico 4 se presentan los resultados:

Tabla 7.

Resultados de aceptación del arquitecto de software

Valoración	1	2	3	4	5
Total de respuestas	0	0	0	10	13
Resultado	0	0	0	40	65
TOTAL	105				
PORCENTAJE DE ACEPTACIÓN (ST/1,2)	87,5				

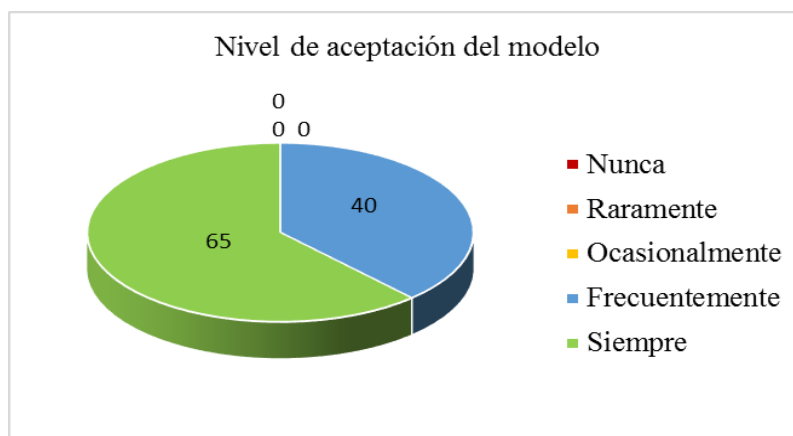


Gráfico 4. Nivel de aceptación del arquitecto de software sobre el modelo MPPDS

El gráfico anterior demuestra que en base a los resultados generados en las encuestas aplicadas al arquitecto de software, el modelo propuesto (MPPDS) presenta un 87,5 % de aceptación ya que consideran que el modelo propuesto cumple a satisfacción con la optimización del proceso de planificación de proyectos de software y es factible de ser aplicado a todos los proyectos de desarrollo de software de la Institución, con lo cual se concluye que el modelo es utilizado en el proceso de planificación de proyectos de desarrollo de software por parte del arquitecto de software.

1.2.3 Análisis y resultados de la aplicación del modelo al líder tecnológico:

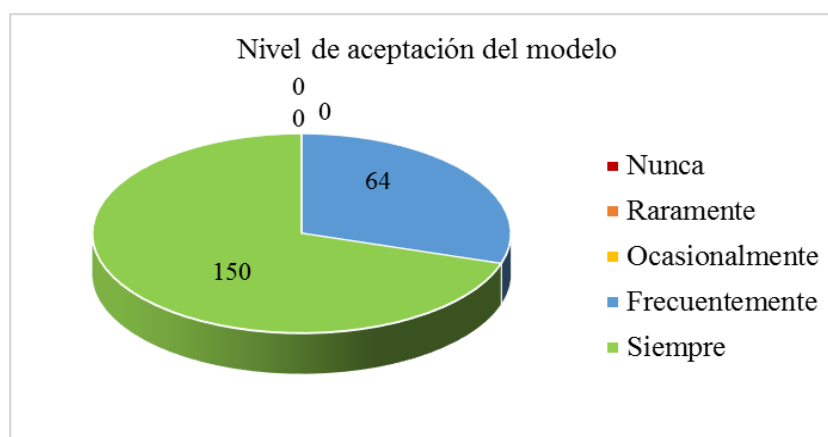
En esta sección se realizará el análisis e interpretación de los resultados generados por el líder tecnológico de la DNSI de la ISTI, a quien se le aplicó la encuesta que se encuentra en el **Anexo W**.

El cálculo se realizó ejecutando la suma total de las valoraciones obtenidas y dividiendo este valor para 1,2, ya que se trata de una persona. En la Tabla 7 y en el Gráfico 5 se presentan los resultados:

Tabla 8.

Resultados de aceptación del líder tecnológico

Valoración	1	2	3	4	5
Total de respuestas	0	0	0	4	19
Resultado	0	0	0	16	95
TOTAL	111				
PORCENTAJE DE ACEPTACIÓN (ST/1,2)	92,5				

**Gráfico 5.** Nivel de aceptación del líder tecnológico sobre el modelo MPPDS

El gráfico anterior demuestra que en base a los resultados generados en las encuestas aplicadas al líder tecnológico, el modelo propuesto (MPPDS) presenta un 92,5 % de aceptación ya que considera que el modelo propuesto cumple a satisfacción con la optimización del proceso de planificación de proyectos de software y es factible de ser aplicado a todos los proyectos de desarrollo de software de la Institución, con lo cual se concluye que el modelo es utilizado siempre en los procesos de planificación de proyectos de desarrollo de software por parte del líder tecnológico.

1.3 Prueba de hipótesis con chi cuadrado

1.3.1 Planteamiento de la hipótesis

La prueba de hipótesis con Chi cuadrado permitirá la comparación de dos atributos para determinar si existe una relación entre ellos, de esta manera se conocerá si una variable depende de la otra para su cumplimiento o ejecución.

a) **Hipótesis de investigación:** Si se implementa un modelo de planificación para la fase de planificación de los proyectos de desarrollo de software durante el período 2015, se optimiza el proceso de planificación para los proyectos de desarrollo de software.

b) **Variable independiente:** El modelo de planificación para la fase de planificación de proyectos de desarrollo de software.

c) **Variable dependiente:** La optimización del proceso de planificación para los proyectos de desarrollo de software en la DNSI de la SEPS.

✓ **Hipótesis Nula (H_0):** El modelo de planificación y la optimización del proceso de planificación de proyectos de desarrollo de software son independientes.

✓ **Hipótesis Alternativa (H_A):** El modelo de planificación y la optimización del proceso de planificación de proyectos de desarrollo de software son dependientes.

1.3.2 Cálculo de frecuencias esperadas, correspondientes a cada frecuencia observada.

a) Frecuencia observada

Como resultado de las encuestas aplicadas al equipo del proyecto (desarrolladores, arquitecto de software y líder tecnológico), en la Tabla 8 y Tabla 9, se presentan los datos de la variable independiente y de la variable dependiente

respectivamente., los datos utilizados para generar estos resultados se encuentran en el **Anexo X.**

Tabla 9.

Variable Modelo de Planificación

MODELO DE PLANIFICACIÓN				
VALORACIÓN	DESARROLLADO R	ARQUITECTO DE SOFTWARE	LÍDER TECNOLÓGIC O	TOTA L
1. Nunca	0	0	0	0
2. Raramente	0	0	0	0
3. Ocasionalmente	0	0	0	0
4. Frecuentemente	8	3	2	13
5. Siempre	14	8	9	31

Tabla 10.

Variable Proceso de Planificación

PROCESO DE PLANIFICACIÓN				
VALORACIÓN	DESARROLLADOR	ARQUITECTO DE SOFTWARE	LÍDER TECNOLÓGICO	TOTAL
1. Nunca	0	0	0	0
2. Raramente	0	0	0	0
3. Ocasionalmente	0	0	0	0
4. Frecuentemente	8	7	2	17
5. Siempre	16	5	10	31

b) Frecuencia esperada

En la Tabla 10 se presenta el cálculo de la intersección entre los datos de la variable independiente y la variable dependiente:

Tabla 11.

Frecuencia esperada para la variable independiente y la variable dependiente

	MODELO DE PLANIFICACIÓN						
	Valoración	1	2	3	4	5	Total
PROCESO DE PLANIFICACIÓN	1	0	0	0	13	31	44
	2	0	0	13	13	31	57
	3	0	0	0	13	31	44
	4	17	17	17	30	48	129
	5	31	31	31	44	62	199
	Total	48	48	61	113	203	473

En la Tabla 11 se presenta el cálculo de la frecuencia esperada de los datos de la variable independiente y la variable dependiente, en función de la siguiente ecuación:

$$E_{i,j} = \frac{\sum_{i=1}^m O_{i,j} * \sum_{j=1}^n O_{i,j}}{\sum_{i=1}^m \sum_{j=1}^n O_{i,j}}$$

En donde: m: número de columnas

n: número de filas
 i: posición de la columna
 j: posición de la fila
 O: frecuencia observada

Tabla 12.

Frecuencia esperada para la variable independiente y la variable dependiente

	MODELO DE PLANIFICACIÓN						
	Valoración	1	2	3	4	5	Total
PROCESO DE PLANIFICACIÓN	1	4,47	4,47	5,67	10,51	18,88	44,00
	2	5,78	5,78	7,35	13,62	24,46	57,00
	3	4,47	4,47	5,67	10,51	18,88	44,00
	4	13,09	13,09	16,64	30,82	55,36	129,00
	5	20,19	20,19	25,66	47,54	85,41	199,00
	Total	48,00	48,00	61,00	113,00	203,00	473,00

1.3.3 Cálculo del valor de chi cuadrado

Para comprobar la dependencia entre la variable dependiente (Proceso de Planificación) y la variable independiente (Modelo de Planificación) se realizará el cálculo de la prueba Chi Cuadrado como se presenta en la Tabla 12, en función de la siguiente ecuación:

$$\chi^2 = \sum_{i=1}^m \sum_{j=1}^n \frac{(O_{i,j} - E_{i,j})^2}{E_{i,j}}$$

En donde: m: número de columnas
 n: número de filas

i: posición de la columna

j: posición de la fila

O: frecuencia observada

E: frecuencia esperada

x: valor Chi Cuadrado

Tabla 13.

Frecuencia esperada para la variable independiente y la variable dependiente

	MODELO DE PLANIFICACIÓN						
	VALORACIÓN	1	2	3	4	5	TOTALES
PROCESO DE PLANIFICACIÓN	1	4,47	4,47	5,67	0,59	7,77	22,97
	2	5,78	5,78	4,34	0,03	1,75	17,68
	3	4,47	4,47	5,67	0,59	7,77	22,97
	4	1,17	1,17	0,01	0,02	0,98	3,34
	5	5,78	5,78	1,11	0,26	6,41	19,35
	TOTALES	21,66	21,66	16,81	1,49	24,69	86,32

1.3.4 Cálculo del valor crítico de chi cuadrado

A continuación se presenta la Tabla de Distribución de Chi Crítico, para lo cual se ha establecido el nivel de significancia de 0,05, de igual manera se calculará los grados de libertad para Chi Crítico, tal como se indica a continuación:

Nivel de significancia: $\alpha = 0,05$

Grados de libertad: $g.l. = (n_c - 1) * (n_f - 1)$

$g.l. = (5 - 1) * (5 - 1)$

$g.l. = 4 * 4$

g.l. = 16

A continuación se presenta en la Tabla 13 el valor de Chi Cuadrado de acuerdo al nivel de significancia y los grados de libertad encontrados:

Tabla 14.

Distribución Chi Cuadrado Crítico

v/p	0,001	0,0025	0,005	0,01	0,025	0,05	0,1
1	10,8274	9,1404	7,8794	6,6349	5,0239	3,8415	2,7055
2	13,8150	11,9827	10,5965	9,2104	7,3778	5,9915	4,6052
3	16,2660	14,3202	12,8381	11,3449	9,3484	7,8147	6,2514
4	18,4662	16,4238	14,8602	13,2767	11,1433	9,4877	7,7794
5	20,5147	18,3854	16,7496	15,0863	12,8325	11,0705	9,2363
6	22,4575	20,2491	18,5475	16,8119	14,4494	12,5916	10,6446
7	24,3213	22,0402	20,2777	18,4753	16,0128	14,0671	12,0170
8	26,1239	23,7742	21,9549	20,0902	17,5345	15,5073	13,3616
9	27,8767	25,4625	23,5893	21,6660	19,0228	16,9190	14,6837
10	29,5879	27,1119	25,1881	23,2093	20,4832	18,3070	15,9872
11	31,2635	28,7291	26,7569	24,7250	21,9200	19,6752	17,2750
12	32,9092	30,3182	28,2997	26,2170	23,3367	21,0261	18,5493
13	34,5274	31,8830	29,8193	27,6882	24,7356	22,3620	19,8119
14	36,1239	33,4262	31,3194	29,1412	26,1189	23,6848	21,0641
15	37,6978	34,9494	32,8015	30,5780	27,4884	24,9958	22,3071
16	39,2518	36,4555	34,2671	31,9999	28,8453	26,2962	23,5418
17	40,7911	37,9462	35,7184	33,4087	30,1910	27,5871	24,7690

1.3.5 Comparación entre el valor esperado y el valor crítico

En el siguiente gráfico 6, se evidencia que el valor de Chi Observado se encuentra en la zona de rechazo para la aceptación de la hipótesis nula (H_0):

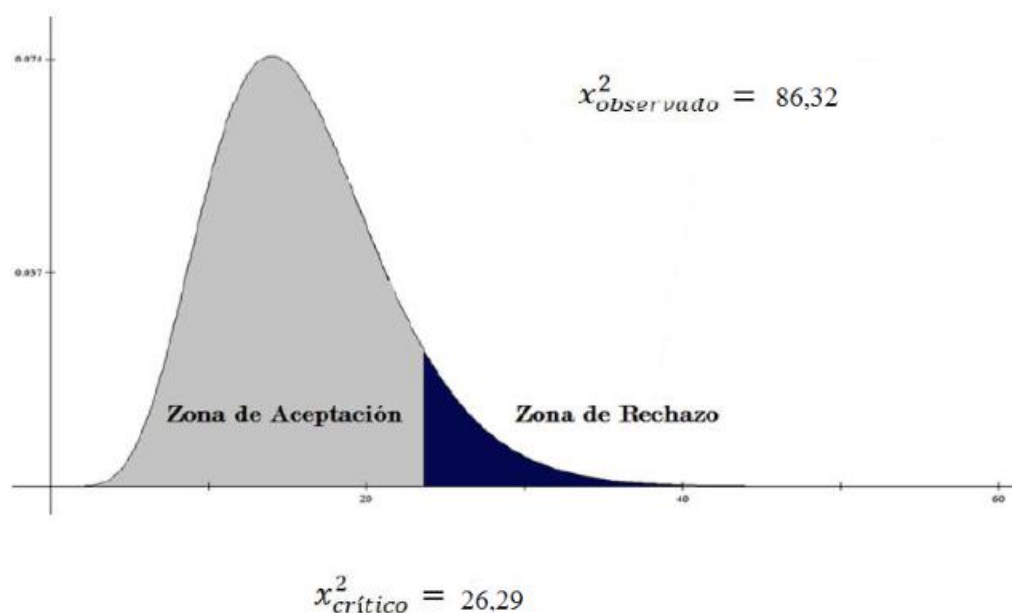


Gráfico 6. Prueba de Chi Cuadrado dependiente

Con este resultado se ha verificado el cumplimiento de la hipótesis planteada, en base al valor observado, el valor crítico y la regla de decisión:

$$86,32 > 26,29$$

Regla de decisión:

- ✓ Se acepta la hipótesis nula (H_0), si $x^2_{observado} < x^2_{crítico}$.
- ✓ Se acepta la hipótesis alternativa (H_A), si $x^2_{observado} > x^2_{crítico}$.

De acuerdo a las afirmaciones anteriores en el presente trabajo de investigación se acepta la hipótesis alternativa (H_A) ya que el valor de Chi Cuadrado es mayor que el de Chi Crítico.

1.3.6 Conclusiones de los resultados presentados

Una vez realizado el análisis y los cálculos correspondientes a los resultados generados de las encuestas aplicadas al equipo del proyecto, se ha validado la aceptación

de la hipótesis planteada a través de la prueba de hipótesis Chi Cuadrado, la misma que concluye que la optimización del proceso de planificación para proyectos de desarrollo de software depende de la aplicación de un Modelo de Planificación para proyectos de desarrollo de software destinado a la ejecución de tareas mediante procedimientos que permitan generar los resultados esperados. Además se ha comprobado que es factible la aplicación del modelo a todos los proyectos de desarrollo de software de la Institución ya que cumple con la optimización del proceso de planificación de proyectos de software.

La aplicación del modelo implica el uso de un plan, procedimientos, técnicas, generación de artefactos y otros documentos con la finalidad de establecer los parámetros del proceso de planificación en la gestión de proyectos de desarrollo de software que permitan su optimización.

CAPÍTULO VI

CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

- ✓ La Superintendencia de Economía Popular y Solidaria a través de la Intendencia de Sistemas y Tecnologías de la Información y la Dirección Nacional de Sistemas de Información, ejecuta la planificación de los proyectos de desarrollo de software bajo mínimos parámetros estandarizados siendo el proceso normal el establecido por la experiencia de cada líder tecnológico. Es por esta razón que se ha propuesto la creación de un modelo de planificación para proyectos de desarrollo de software que se adapte a los procesos institucionales y que optimice dicho proceso.
- ✓ La evolución de las normas, estándares y modelos de gestión de proyectos y planificación de proyectos de desarrollo de software, han aportado para mejorar la eficiencia y eficacia en la gestión y planificación de proyectos de desarrollo de software a través de artefactos de entrada, salida, técnicas y herramientas identificadas dentro de los grupos de procesos y las áreas de conocimiento.
- ✓ La estructura de las normas, estándares y modelos, han permitido representar de manera efectiva las áreas de conocimiento, los procesos, las entradas, herramientas, técnicas y salidas que deben ser ejecutadas por el equipo del proyecto, con la finalidad de realizar una adecuada planificación de los proyectos de desarrollo de software.
- ✓ El modelo de Planificación para Proyectos de Desarrollo de Software (MPPDS) está basado en la Guía de buenas prácticas PMBOK, quinta edición, por tal razón este modelo se encuentra definido en áreas de conocimiento organizado en grupos de procesos, a la vez cada proceso tiene

entradas, herramientas, técnicas y salidas. El modelo está adaptado a las necesidades institucionales, los procedimientos implementados serán utilizados en las fases de análisis y levantamiento de requerimientos, diseño del sistema, y construcción y pruebas tecnológicas en los proyectos de desarrollo de software mientras que las técnicas pueden ser utilizadas de acuerdo a la necesidad de cada proyecto.

- ✓ El modelo de Planificación propuesto optimiza el proceso de planificación de proyectos de desarrollo de software en la Institución, por consiguiente su implementación permite solventar los inconvenientes presentados en la ejecución y entrega de los productos de desarrollo de software, mejorando la satisfacción de las áreas funcionales requirentes, de la misma manera puede ser robustecido conforme se vayan consolidando los procesos institucionales.
- ✓ Con el desarrollo de la presente investigación se ha detectado una problemática que es común para la mayoría de Instituciones públicas, ya que no existe un modelo o metodología que se pueda utilizar en su totalidad para el proceso de planificación de los proyectos de desarrollo de software, sin embargo con el análisis realizado se ha comprobado que gracias a estándares y normas se puede optimizar el proceso de planificación.
- ✓ El modelo de Planificación propuesto es factible de ser implementado para todos los proyectos de desarrollo de software de la Dirección Nacional de Sistemas de Información de la Intendencia de Sistemas y Tecnologías de la Información de la Superintendencia de Economía Popular y Solidaria y para su uso se han generado procedimientos, artefactos y entregables que permiten la optimización del proceso de planificación de los proyectos de desarrollo de software.
- ✓ Una vez realizado el análisis y los cálculos correspondientes a los resultados generados de las encuestas aplicadas al equipo del proyecto, se ha validado que el Modelo de Planificación para proyectos de desarrollo de software propuesto ha permitido la optimización del proceso de planificación para proyectos de desarrollo de software, ya que con el uso de los artefactos definidos en el plan, los procedimientos, técnicas y otros documentos, se ha

logrado que el sistema Acopio Balances, genere el producto planificado con el uso de los recursos estimados y a satisfacción del área requirente.

- ✓ Con la implementación del modelo MPDDS se ha logrado una madurez en cuanto al proceso de desarrollo de software, ya que el mismo ha sido institucionalizado y será de uso exclusivo para los proyectos de desarrollo de software de la DNSI.

6.2 Recomendaciones

- ✓ Se recomienda el uso del Modelo de Planificación para Proyectos de Desarrollo de Software (MPPDS) durante el proceso de planificación, con la finalidad de optimizar este proceso y atender las necesidades institucionales a satisfacción.
- ✓ Se recomienda que el modelo sea adaptado en el proceso de planificación de los proyectos de desarrollo de software haciendo uso de los documentos propuestos en el modelo, los mismos que permiten la convivencia con las metodologías de desarrollo ágil aplicadas en la Institución.
- ✓ Se sugiere que para la implementación del modelo MPPDS se considere utilizar todos los procedimientos establecidos para cada fase del proceso de desarrollo de software ya que en su conjunto permiten la planificación clara y precisa de la ejecución del proyecto.
- ✓ Se sugiere que cuando el modelo MPPDS sea aplicado por un equipo de proyecto de desarrollo de software, previamente el modelo sea dado a conocer a todos los miembros del equipo ya que el mismo requiere en su momento de la intervención activa de cada miembro del equipo.
- ✓ La aplicación del modelo MPPDS principalmente en la elaboración de la EDT y su Diccionario requiere mayor dedicación de tiempo al proceso de planificación por parte del equipo del proyecto, sin embargo al lograr una planificación más clara y precisa se han obtenido los resultados esperados y a satisfacción de las Intendencias funcionales solicitantes.

- ✓ Se sugiere que para la organización de los equipos de proyecto se aplique los formatos establecidos para el efecto ya que permiten tener un control y mantener la comunicación entre los equipos.
- ✓ Se recomienda que para iniciar la implementación del MPPDS todos los líderes tecnológicos apliquen el modelo por lo menos a uno de sus proyectos a cargo para familiarizarse con el mismo y facilitar la implementación total en todos los proyectos de desarrollo de software.
- ✓ Con la finalidad de profundizar en la presente investigación, se recomienda indagar y utilizar métodos y modelos de estimación de proyectos referentes a la ingeniería de software para la estimación de los recursos, mismo que será motivo de una nueva investigación.

Referencias bibliográficas

- Boehm, B. (1981). *Software Engineering Economics*. Prentice Hall.
- Boehm, B. (2000). *Software Cost Estimation in COCOMO II*. Prentice Hall.
- Brooks, F. (1995). *The Mythical Man-Month*. ADDISON-WESLEY.
- Grassi, J. P. (s.f.). Estudio sobre metodologías para la gestión de proyectos de software. Propuesta y aplicaciones. San Andrés.
- León, G. (1998). *Ingeniería de Sistemas de Software*. Madrid: ISDEFE.
- Navarro, J. (2006). *Planificación de un proyecto de software*. Obtenido de Planificación de un proyecto de software: www.slideshare.net/hrubenleiva21/planificacin-de-proyectos-de-software
- Orna, C. (2013). Modelo de Calidad para la etapa de Planificación de proyectos medianos de software. *Modelo de Calidad para la etapa de Planificación de proyectos medianos de software*. Latacunga, Cotopaxi, Ecuador.
- PMBOK. (2013). *Guía de los Fundamentos para la Dirección de Proyectos*. Pensilvania, EE.UU.: GLOBAL STANDARD.
- Pressman, R. (2010). *Ingeniería de Software - Un enfoque práctico* (SÉPTIMA ed.). México: MCGRAW HILL.
- Royce, W. (1970). *Managing the Development of Large Software Systems: Concepts and Techniques*. WESCON.
- Ruiz, F. (1999). *Norma IEEE 1058.1 para la Planificación de Gestión de Proyectos de Software*.
- Slideshare. (07 de 06 de 2012). *Planificación de Proyectos de Software*. Obtenido de <http://www.slideshare.net/hrubenleiva21/planificacin-de-proyectos-de-software>: <http://www.slideshare.net/hrubenleiva21/planificacin-de-proyectos-de-software>
- Sommerville, I. (2005). *INGENIERÍA DEL SOFTWARE* (SEPTIMA ed.). Madrid: Addison Wesley.



VICERRECTORADO DE INVESTIGACIÓN, INNOVACIÓN Y TRANSFERENCIA DE TECNOLOGÍA

MAESTRÍA EN INGENIERÍA DE SOFTWARE

CERTIFICACIÓN

Se certifica que el presente trabajo fue desarrollado por la señorita: **SILVIA CONSUELO ORDÓÑEZ ESOBAR.**

En la ciudad de Latacunga, a los **30 días del mes de marzo de 2016.**

Ing. Lucas Garcés

DIRECTOR DE TESIS

Aprobado por:

Ing. Xavier Montaluisa

COORDINADOR DE MAESTRÍA

Dr. Rodrigo Vaca

SECRETARIO ACADÉMICO