

**UNIVERSIDAD HISPANOAMERICANA
ESCUELA DE INGENIERÍA INFORMÁTICA
TESINA PARA OPTAR POR EL GRADO DE
BACHILLERATO EN LA CARRERA DE
INGENIERÍA INFORMÁTICA**

**PROPUESTA DE MEJORA EN LA
METODOLOGÍA DE DESARROLLO DE
SISTEMAS DE INFORMACIÓN DE LA
CONTRALORÍA GENERAL DE LA REPÚBLICA**

Sustentante:

Josué Francisco Solano Morales

TUTOR:

Msc. Alejandro Bogantes Salazar

Agosto, 2019

ÍNDICE DE CONTENIDO

ÍNDICE DE CONTENIDO	II
ÍNDICE DE TABLAS	VI
ÍNDICE DE FIGURAS	VII
DECLARACIÓN JURADA.....	IX
CARTA DE APROBACIÓN DEL TUTOR	X
CARTA DE APROBACIÓN DEL LECTOR	XI
CARTA DE REVISIÓN DEL FILÓLOGO	XII
DEDICATORIA	XV
AGRADECIMIENTO.....	XVI
ABREVIATURAS	XVII
RESUMEN	XVIII
CAPÍTULO I: PROBLEMA DEL PROYECTO.....	1
1.1 ANTECEDENTES Y JUSTIFICACIÓN DEL PROYECTO	2
1.1.1 Marco de referencia empresarial y contextual.....	2
1.1.2 Justificación del proyecto.....	6
1.2 DEFINICIÓN DEL PROBLEMA	7
1.2.1 Problemática	7
1.2.1.1 Diagrama de Ishikawa.....	8
1.2.2 Problema General	9
1.2.3 Problemas Específicos	9
1.3 OBJETIVOS DEL PROYECTO	10
1.3.1 Objetivo general.....	10
1.3.2 Objetivos específicos	10
1.4 ALCANCES Y LIMITACIONES	10

1.4.1 Alcances	10
1.4.2 Limitaciones	11
1.5 CRONOGRAMA DE ACTIVIDADES	12
CAPÍTULO II: MARCO TEÓRICO	14
2.1 CONTROL EN LA HACIENDA PÚBLICA	15
2.1.1 Definición de Control	15
2.1.2 Definición de Hacienda Pública	17
2.1.2 Entidades de Fiscalización Superior (EFS).....	18
2.2 SISTEMAS	19
2.2.1 Generalidades de los sistemas	19
2.2.2 Sistemas de Información	21
2.2.2.1 Generalidades de un Sistema de Información.....	21
2.2.2.2 Clasificación de los Sistemas de Información	22
2.2.2.3 Componentes de un Sistema de Información.....	23
2.2.2.4 Desarrollo de un Sistema de Información	24
2.2.2.5 Evaluación del desarrollo de Sistemas de Información.....	27
2.3 METODOLOGÍAS DE DESARROLLO DE SISTEMAS DE INFORMACIÓN	29
2.3.1 Concepto de Metodología	29
2.3.2 Metodologías tradicionales de desarrollo de sistemas	30
2.3.3 Metodologías Ágiles de desarrollo de sistemas	39
2.4 Gestión de proyectos	61
2.4.1 Generalidades de los proyectos	61
2.4.2 Metodologías de gestión de proyectos.....	67
CAPÍTULO III: MARCO METODOLÓGICO.....	82
3.1 TIPO Y ENFOQUE DE LA INVESTIGACIÓN.....	83

3.1.1 Tipo de investigación.....	83
3.1.2 Enfoque de la investigación	84
3.2 FUENTES Y SUJETOS DE INFORMACIÓN	85
3.2.1 Fuentes primarias.....	85
3.2.2 Fuentes secundarias	85
3.2.3 Sujetos de información	86
3.3 TÉCNICAS Y HERRAMIENTAS DE RECOLECCIÓN DE DATOS	86
3.4 VARIABLES DE INVESTIGACIÓN	87
3.5 DISEÑO DE LA INVESTIGACIÓN.....	89
3.6 MATRIZ DE COHERENCIA	92
CAPÍTULO IV: DIAGNÓSTICO DE LA SITUACIÓN ACTUAL.....	94
4.1 Diagnóstico Administrativo	95
4.2 Diagnóstico de Percepción.....	106
4.2.1 Percepción de los usuarios de los sistemas de información.....	106
4.2.2 Percepción del equipo de desarrollo de sistemas de la UTI.....	113
4.3 Brechas o conclusiones del diagnóstico	118
CAPÍTULO V: PROPUESTA DE PROYECTO	121
5.1 Análisis de la metodología de desarrollo de software	122
5.2 Análisis comparativo entre metodologías utilizadas actualmente en desarrollo de software.	125
5.3 Propuesta de mejora a la metodología de desarrollo de sistemas de la información de la Contraloría General de la República	129
5.3.1 Diagramas del proceso de desarrollo de software propuesto	129
5.3.2 Metodología propuesta.....	132
CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES	180
6.1 Conclusiones	181

6.2 Recomendaciones	182
CAPÍTULO VII: APÉNDICES Y ANEXOS.....	184
7.1 Apéndices.....	185
Apéndice 1: Encuesta aplicada al equipo de desarrollo de software de la UTI	185
BIBLIOGRAFÍA	191

ÍNDICE DE TABLAS

Tabla 1. Métricas relacionadas al Proceso: Gestionar la Identificación y Construcción de Soluciones de COBIT 5.....	28
Tabla 2. Diferencias entre metodologías ágiles y no ágiles	42
Tabla 3. Prácticas de programación extrema.....	48
Tabla 4: Análisis de las características generales de PMBOK y PRINCE2	78
Tabla 5: Relación entre los procesos de PMBOK y los procesos de PRINCE2	79
Tabla 6: Relación entre las áreas de conocimientos definidas en PMBOK y los procesos y componentes de PRINCE2	80
Tabla 7. Sujetos de Información	86
Tabla 8: Variables de Investigación.....	88
Tabla 9: Matriz de coherencia	92
Tabla 10. Resultados más relevantes de la encuesta para medir la satisfacción de los usuarios del Sistema Fipnet	107
Tabla 11: Comparación entre metodologías Cascada, Programación Extrema (XP) y Scrum ...	125

ÍNDICE DE FIGURAS

Figura 1: Estructura Organizacional de la Contraloría General de la República.....	4
Figura 2: Estructura Organizacional de la Unidad de Tecnologías de Información.	6
Figura 3: Diagrama de Ishikawa.	9
Figura 4: Cronograma de Actividades.	13
Figura 5: Componentes de un sistema de información según Stair	23
Figura 6: Ciclo de vida del desarrollo de sistemas de Stair	25
Figura 7: Modelo de la cascada	32
Figura 8: Modelo Incremental	35
Figura 9: Proceso Unificado	38
Figura 10: Proceso de programación extrema	47
Figura 11: Flujo del proceso Scrum.....	58
Figura 12: Representación Genérica del Ciclo de Vida de un Proyecto	65
Figura 13: Interrelación entre los Componentes Clave de los Proyectos de la Guía del PMBOK71	71
Figura 14: Estructura PRINCE2	75
Figura 15: Etapas de la investigación.....	89
Figura 16: Fases de un proyecto	97
Figura 17: Proceso de análisis de requerimientos.....	98
Figura 18: Proceso de diseño conceptual	99
Figura 19: Proceso de diseño detallado.....	100
Figura 20: Construcción y pruebas	101
Figura 21: Proceso de implementación	102
Figura 22: Diagrama del desarrollo de un nuevo sistema	105
Figura 23: Percepción de los usuarios indicando si el sistema Fipnet le permite alcanzar los objetivos de las labores asignadas.....	108
Figura 24: Percepción de los usuarios indicando si el desempeño del sistema Fipnet es satisfactorio	109
Figura 25: Percepción de los usuarios indicando si el sistema Fipnet facilita las labores asignadas	109

Figura 26: Percepción de los usuarios indicando si el sistema Fipnet tiene elementos que facilitan su uso.....	110
Figura 27: Percepción de los usuarios indicando si el sistema SIPP contribuyó a agilizar el proceso de aprobación presupuestaria.....	111
Figura 28: Percepción de los usuarios del sistema SIPP sobre la navegación del sistema	111
Figura 29: Percepción del equipo de desarrollo de sistemas de la UTI sobre si la metodología vigente se adapta a las necesidades institucionales.....	114
Figura 30: Percepción del equipo de desarrollo de sistemas de la UTI sobre si las técnicas para la especificación de requerimientos son las adecuadas.....	114
Figura 31: Percepción del equipo de desarrollo de sistemas de la UTI sobre si la gestión de cambios se realiza de manera adecuada	115
Figura 32: Percepción del equipo de desarrollo de sistemas de la UTI sobre si cuentan con las herramientas necesarias para el desarrollo de aplicaciones.....	116
Figura 33: Percepción del equipo de desarrollo de sistemas de la UTI sobre si existe suficiente flexibilidad para gestionar cambios en procura de los objetivos.....	117

DECLARACIÓN JURADA

Yo Josué Francisco Solano Morales, mayor de edad, portador de la cédula de identidad número 1-1329-0585 egresado de la carrera de Ingeniería Informática de la Universidad Hispanoamericana, hago constar por medio de éste acto y debidamente apercibido y entendido de las penas y consecuencias con las que se castiga en el Código Penal el delito de perjurio, ante quienes se constituyen en el Tribunal Examinador de mi trabajo de tesis para optar por el título de Bachillerato en Ingeniería Informática, juro solemnemente que mi trabajo de investigación titulado: Propuesta de mejora en la metodología de desarrollo de sistemas de información de la Contraloría General de la República es una obra original que ha respetado todo lo preceptuado por las Leyes Penales, así como la Ley de Derecho de Autor y Derecho Conexos número 6683 del 14 de octubre de 1982 y sus reformas, publicada en la Gaceta número 226 del 25 de noviembre de 1982; incluyendo el numeral 70 de dicha ley que advierte; artículo 70. Es permitido citar a un autor, transcribiendo los pasajes pertinentes siempre que éstos no sean tantos y seguidos, que puedan considerarse como una producción simulada y sustancial, que redunde en perjuicio del autor de la obra original. Asimismo, quedo advertido que la Universidad se reserva el derecho de protocolizar este documento ante Notario Público.

En fe de lo anterior, firmo en la ciudad de San José, a los 26 días del mes de julio del año dos mil diecinueve.



Firma del estudiante

Cédula: 1-1329-0585

CARTA DE APROBACIÓN DEL TUTOR

CARTA DEL TUTOR

San José, 01 de agosto de 2019

Carrera Ingeniería Informática
Universidad Hispanoamericana

Estimado señor:

El estudiante Josué Solano Morales, cédula de identidad número 1-1329-0585, me ha presentado, para efectos de revisión y aprobación, el trabajo de investigación denominado "Propuesta de mejora en la metodología de desarrollo de sistemas de información de la Contraloría General de la República", el cual ha elaborado para optar por el grado académico de Bachillerato en Ingeniería Informática.

En mi calidad de tutor, he verificado que se han hecho las correcciones indicadas durante el proceso de tutoría y he evaluado los aspectos relativos a la elaboración del problema, objetivos, justificación; antecedentes, marco teórico, marco metodológico, tabulación, análisis de datos; conclusiones y recomendaciones.

De los resultados obtenidos por el postulante, se obtiene la siguiente calificación:

a)	ORIGINAL DEL TEMA	10%	9%
b)	CUMPLIMIENTO DE ENTREGA DE AVANCES	20%	18%
C)	COHERENCIA ENTRE LOS OBJETIVOS, LOS INSTRUMENTOS APLICADOS Y LOS RESULTADOS DE LA INVESTIGACION	30%	30%
d)	RELEVANCIA DE LAS CONCLUSIONES Y RECOMENDACIONES	20%	20%
e)	CALIDAD, DETALLE DEL MARCO TEORICO	20%	20%
	TOTAL		97%

En virtud de la calificación obtenida, se avala el traslado al proceso de lectura.

Atentamente,

Msc. Alejandro Bogantes Salazar
Cédula identidad: 303940389
Carné Colegio Profesional: 4644

CARTA DE APROBACIÓN DEL LECTOR

CARTA DE LECTOR

San José, 23 de septiembre de 2019

Universidad Hispanoamericana
Sede Llorente
Carrera de Ingeniería Informática

Estimado señor

El estudiante Josué Solano Morales, cédula de identidad 1-1329-0585, me ha presentado para efectos de revisión y aprobación, el trabajo de investigación denominado "PROPUESTA DE MEJORA EN LA METODOLOGÍA DE DESARROLLO DE SISTEMAS DE INFORMACIÓN DE LA CONTRALORÍA GENERAL DE LA REPÚBLICA ", el cual ha elaborado para obtener su grado de BACHILLERATO EN LA CARRERA DE INGENIERÍA INFORMÁTICA.

He revisado y he hecho las observaciones relativas al contenido analizado, particularmente lo relativo a la coherencia entre el marco teórico y análisis de datos, la consistencia de los datos recopilados y la coherencia entre éstos y las conclusiones; asimismo, la aplicabilidad y originalidad de las recomendaciones, en términos de aporte de la investigación. He verificado que se han hecho las modificaciones correspondientes a las observaciones indicadas.

Por consiguiente, este trabajo cuenta con mi aval para ser presentado en la defensa pública.

Atte.



Lic. Jorge Iván Arroyo Torres.
Carné: 110680975

CARTA DE REVISIÓN DEL FILÓLOGO

27 de setiembre del 2019

Señores
Comisión de Trabajos Finales de Graduación
Universidad Tecnológica Costarricense
Escuela de Ingeniería Informática

Estimados señores:

Yo, Noel Molina Blanco, cédula ocho cero cuarenta y seis cero quinientos ochenta y siete, vecino de San Juan de Tibás, de profesión Licenciado en Filología clásica, y que cuento con conocimientos y experiencia en revisión filológica de textos, doy fe de haber revisado el trabajo final de graduación del sustentante Josué Solano Morales, titulado, “Propuesta de mejora en la metodología de desarrollo de sistemas de información de la Contraloría General de la República”, para optar por el grado de Bachillerato en Ingeniería Informática.

Después de la revisión y corrección de la estudiante, considero que el Informe del Trabajo Final de Graduación indicado anteriormente, cuenta con la revisión y corrección filológica en aspectos fundamentales que lo hacen apto para ser presentado al proceso de evaluación de los Trabajos Finales de Graduación en el nivel de Bachillerato-

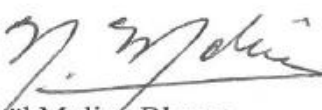
Quedo a su disposición para cualquier consulta en:

Email: noelmolina16@hotmail.com

Teléfono celular: 84199224

Carné Colypro 57465

De ustedes muy atentamente,



Noél Molina Blanco
Carné Colypro 57465

**UNIVERSIDAD HISPANOAMERICANA
CENTRO DE INFORMACION TECNOLOGICO (CENIT)
CARTA DE AUTORIZACIÓN DE LOS AUTORES PARA LA CONSULTA, LA
REPRODUCCION PARCIAL O TOTAL Y PUBLICACIÓN ELECTRÓNICA
DE LOS TRABAJOS FINALES DE GRADUACION**

San José, 27 de setiembre de 2019

Señores:


Universidad Hispanoamericana
Centro de Información Tecnológico (CENIT)

Estimados Señores:

El suscrito Josué Francisco Solano Morales con número de identificación 1-1329-0585 autor del trabajo de graduación titulado “PROPUESTA DE MEJORA EN LA METODOLOGÍA DE DESARROLLO DE SISTEMAS DE INFORMACIÓN DE LA CONTRALORÍA GENERAL DE LA REPÚBLICA” presentado y aprobado en el año 2019 como requisito para optar por el título de Bachillerato en la carrera de Ingeniería Informática; SI autorizo al Centro de Información Tecnológico (CENIT) para que con fines académicos, muestre a la comunidad universitaria la producción intelectual contenida en este documento.

De conformidad con lo establecido en la Ley sobre Derechos de Autor y Derechos Conexos N° 6683, Asamblea Legislativa de la República de Costa Rica.

Cordialmente,



Josué Francisco Solano Morales
Cédula de identidad: 1-1329-0585

**ANEXO 1 (Versión en línea dentro del Repositorio)
LICENCIA Y AUTORIZACIÓN DE LOS AUTORES PARA PUBLICAR Y
PERMITIR LA CONSULTA Y USO**

Parte 1. Términos de la licencia general para publicación de obras en el repositorio institucional

Como titular del derecho de autor, confiero al Centro de Información Tecnológico (CENIT) una licencia no exclusiva, limitada y gratuita sobre la obra que se integrará en el Repositorio Institucional, que se ajusta a las siguientes características:

- a) Estará vigente a partir de la fecha de inclusión en el repositorio, el autor podrá dar por terminada la licencia solicitándolo a la Universidad por escrito.
- b) Autoriza al Centro de Información Tecnológico (CENIT) a publicar la obra en digital, los usuarios puedan consultar el contenido de su Trabajo Final de Graduación en la página Web de la Biblioteca Digital de la Universidad Hispanoamericana
- c) Los autores aceptan que la autorización se hace a título gratuito, por lo tanto, renuncian a recibir beneficio alguno por la publicación, distribución, comunicación pública y cualquier otro uso que se haga en los términos de la presente licencia y de la licencia de uso con que se publica.
- d) Los autores manifiestan que se trata de una obra original sobre la que tienen los derechos que autorizan y que son ellos quienes asumen total responsabilidad por el contenido de su obra ante el Centro de Información Tecnológico (CENIT) y ante terceros. En todo caso el Centro de Información Tecnológico (CENIT) se compromete a indicar siempre la autoría incluyendo el nombre del autor y la fecha de publicación.
- e) Autorizo al Centro de Información Tecnológica (CENIT) para incluir la obra en los índices y buscadores que estimen necesarios para promover su difusión.
- f) Acepto que el Centro de Información Tecnológico (CENIT) pueda convertir el documento a cualquier medio o formato para propósitos de preservación digital.
- g) Autorizo que la obra sea puesta a disposición de la comunidad universitaria en los términos autorizados en los literales anteriores bajo los límites definidos por la universidad en las “Condiciones de uso de estricto cumplimiento” de los recursos publicados en Repositorio Institucional.

SI EL DOCUMENTO SE BASA EN UN TRABAJO QUE HA SIDO PATROCINADO O APOYADO POR UNA AGENCIA O UNA ORGANIZACIÓN, CON EXCEPCIÓN DEL CENTRO DE INFORMACIÓN TECNOLÓGICO (CENIT), EL AUTOR GARANTIZA QUE SE HA CUMPLIDO CON LOS DERECHOS Y OBLIGACIONES REQUERIDOS POR EL RESPECTIVO CONTRATO O ACUERDO.

DEDICATORIA

Dedico este proyecto a todas aquellas personas que durante esta nueva etapa universitaria estuvieron brindándome incondicionalmente su apoyo. Dentro de estas personas quiero destacar a mi familia, que ha sido un pilar en cada uno de los proyectos que he emprendido, así como a mi novia que me acompañó y animó en todo este proceso.

AGRADECIMIENTO

Primeramente, agradezco a Dios la oportunidad de alcanzar una nueva carrera universitaria, brindándome sabiduría para tomar buenas decisiones, fuerzas para no desfallecer en el proceso, pero sobre todo su amor que da sentido a cada día de mi vida.

A mi familia y a mi novia por su apoyo incondicional desde que tomé la decisión de asumir este reto.

A mi profesor tutor por su disposición, compromiso y valiosos aportes para concluir este proyecto de manera exitosa.

A la Unidad de Tecnologías de Información de la Contraloría General de la República por abrir sus puertas en busca de una mejora en sus procesos y brindarme los insumos necesarios para finalizar este proyecto.

A aquellos compañeros comprometidos con el crecimiento profesional, con los que pude compartir y trabajar en diversos cursos.

A profesores y personal administrativo de la Universidad, que con su compromiso hacia la formación de nuevos profesionales, me brindaron su apoyo de una u otra manera para culminar este proceso.

ABREVIATURAS

CGR: Contraloría General de la República

COBIT: Objetivos de Control para Información y Tecnologías Relacionadas (de sus siglas en inglés: Control Objectives for Information and related Technology)

CGTIC: Comité Gerencial de Tecnologías de Información y Comunicaciones

DGA: División de Gestión de Apoyo

EDT: Estructura de desglose de Trabajo

EFS: Entidades de Fiscalización Superior

INTOSAI: Organización Internacional de las Entidades Fiscalizadoras Superiores

IPMA: Asociación Internacional para la Dirección de Proyectos (de sus siglas en inglés: International Project Management Association)

OCDE: Organización para la Cooperación y el Desarrollo Económico

PAO: Plan Anual Operativo

PMI: Instituto de Administración de Proyectos (de sus siglas en inglés: Project Management Institute)

PMBOK: Fundamentos para la dirección de proyectos (de sus siglas en inglés: Project Management Body of Knowledge)

PRINCE: Proyectos en ambientes controlados (de sus siglas en inglés Projects in Controlled Environments)

PTAC: Plan Táctico de Tecnologías de Información y Comunicación

PU: Proceso Unificado

RUP: Proceso Unificado Racional

TI: Tecnologías de Información

TIC: Tecnologías de información y comunicación

UTI: Unidad de Tecnologías de Información

XP: Programación Extrema

RESUMEN

La Contraloría General de la República es un órgano auxiliar de la Asamblea Legislativa, cuya principal tarea es la fiscalización para mejorar la gestión de la Hacienda Pública. En ese sentido, la institución ha definido dentro de su Plan Estratégico 2013-2020 como uno de sus objetivos el siguiente: “Transformar el proceso de fiscalización integral sustentándolo en tecnologías de información para incrementar su confiabilidad y oportunidad”. (Contraloría General de la República, 2013).

Producto de lo anterior, el desarrollo de aplicaciones juega un papel fundamental en ese proceso. La institución, con los recursos que dispone actualmente, debe realizar un esfuerzo para satisfacer, de la mejor forma, una mayor demanda institucional en materia de sistemas de información.

Por lo tanto, la CGR debería contar con una metodología de desarrollo de sistemas de vanguardia, en la que se empleen las mejores prácticas, lo que permitirá alcanzar los resultados de una manera ágil y con la calidad esperada. Sin embargo, producto de la ejecución de una serie de diagnósticos y análisis, se concluyó que la metodología para el desarrollo de sistemas de información vigente, no se adapta a las necesidades actuales de la institución ni está acorde con los recursos disponibles.

Para solventar dicha situación, en el presente proyecto se desarrolla una propuesta de actualización a dicha metodología, la cual incluye prácticas de vanguardia utilizadas en la industria del desarrollo de sistemas de información, por ejemplo, principios de metodologías ágiles.

Asimismo, se propone un mayor involucramiento de los usuarios expertos durante la ejecución de los proyectos de desarrollo de sistemas de información, buscando que realmente se satisfagan las necesidades que originaron el proyecto con un producto de calidad.

Aparte de lo anterior, se busca una respuesta más ágil ante los desafíos de un entorno cambiante, que permita a la Contraloría encontrar en los sistemas de información, herramientas fundamentales para la ejecución de sus funciones en pro del bienestar de la sociedad costarricense.

CAPÍTULO I: PROBLEMA DEL PROYECTO

1.1 ANTECEDENTES Y JUSTIFICACIÓN DEL PROYECTO

1.1.1 Marco de referencia empresarial y contextual

1.1.1.1 Antecedentes del contexto de la institución

La Contraloría General de la República (CGR) fue concebida en el año 1949, cuando la Asamblea Constituyente la incorporó en la Constitución Política vigente, la cual, en el artículo 183 establece lo siguiente: “La Contraloría General de la República es una institución auxiliar de la Asamblea Legislativa en la vigilancia de la Hacienda Pública; pero tiene absoluta independencia funcional y administrativa en el desempeño de sus labores”. (Asamblea Nacional Constituyente, 1949).

La CGR funge un papel primordial en el aparato estatal como órgano de control de la Hacienda Pública. En la Constitución Política se definieron sus principales funciones, las cuales se han venido incrementando por medio de la promulgación de otras leyes, como la Ley Orgánica de la Contraloría General de la República N°7428, Ley General de Control Interno N°8292, Ley de Contratación Administrativa N°7494, Ley de Administración Financiera de la República y Presupuestos Públicos N°8131 y Ley contra la corrupción y el enriquecimiento ilícito en la función pública N°8422.

1.1.1.2 Ideas Rectoras

La función de la CGR como vigilante de la Hacienda Pública se evidencia en sus ideas rectoras, con cuya base, la institución ha definido lo siguiente:

- Visión: “Garantizaremos a la sociedad costarricense, la vigilancia efectiva de la Hacienda Pública”. (Contraloría General de la República, 2013).
- Misión: “Somos el órgano constitucional, auxiliar de la Asamblea Legislativa que fiscaliza el uso de los fondos públicos para mejorar la gestión de la Hacienda Pública y contribuir al control político y ciudadano”. (CGR, 2013, p.5).
- Valores:
 - “Excelencia: Buscar la máxima calidad y desempeño en el trabajo diario.
 - Respeto: Valorar los derechos y formas de pensar de los demás.

- Justicia: Dar a los demás lo que les corresponde de acuerdo con sus derechos y deberes.
- Integridad: Realizar todas las acciones con rectitud.
- Compromiso: Sentir identificación con la Contraloría General de la República y así dar el máximo Esfuerzo”. (CGR, 2013, p.6).

1.1.1.3 Organización

Tal y como lo establece la Constitución Política, la CGR es liderada por un Contralor y un Subcontralor, los cuales son elegidos por la Asamblea Legislativa por un periodo de ocho años.

Asimismo, para operacionalizar las funciones que se le han encomendado, la CGR se ha organizado en cuatro divisiones, las cuales son las siguientes:

- División de Fiscalización Operativa y Evaluativa.
- División Jurídica.
- División de Contratación Administrativa.
- División de Gestión de Apoyo.

Algunas de esas divisiones se subdividen en áreas o unidades, según se puede apreciar en la figura N°1.

Estructura Organizacional de la Contraloría General de la República 1/

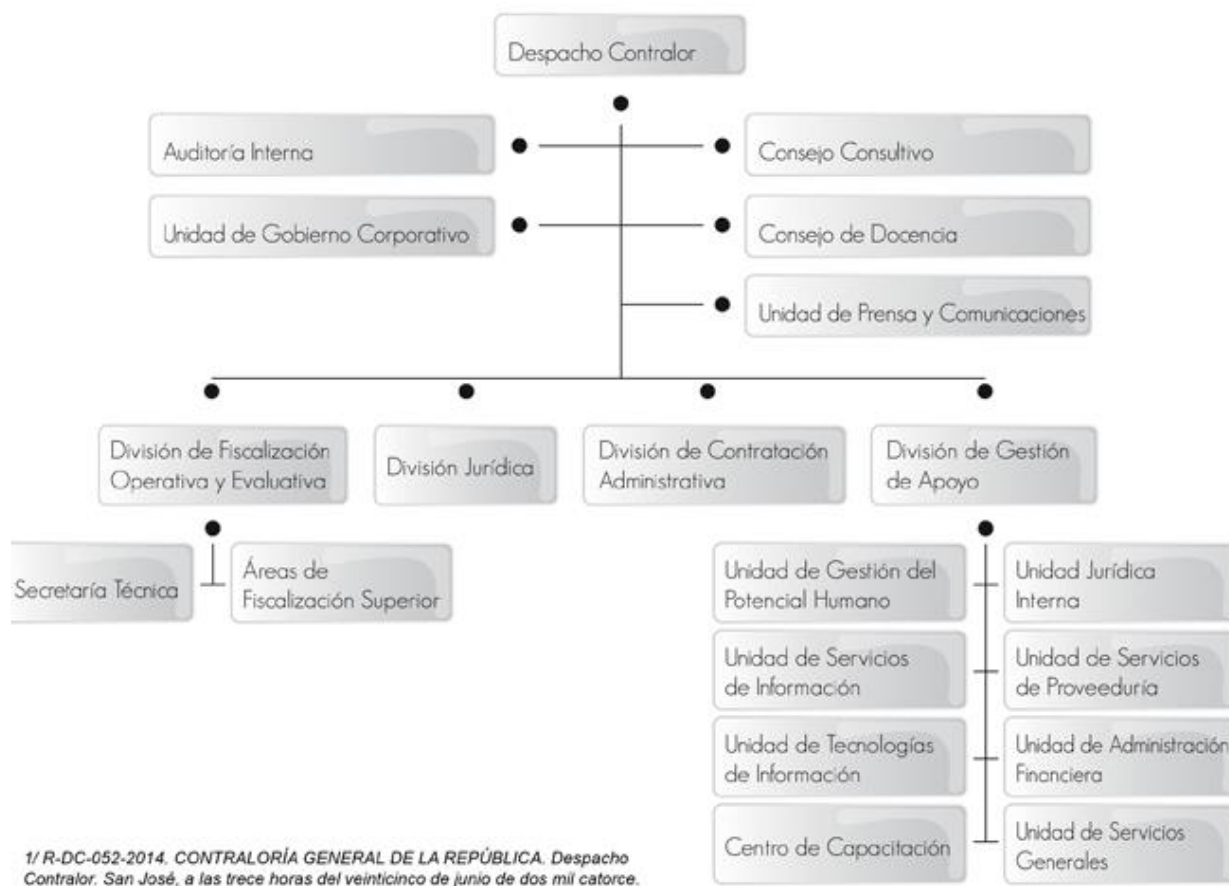


Figura 1: Estructura Organizacional de la Contraloría General de la República.

Fuente: Contraloría General de la República (2018).

El presente proyecto se ubica en la División de Gestión de Apoyo (DGA), que es descrita en el Reglamento Orgánico de la Contraloría General de la República de la siguiente manera:

Agrupar al conjunto de unidades encargadas de adquirir, generar, transferir y potenciar la información y el conocimiento con base en el aprendizaje individual y colectivo, a través del potencial humano y de las tecnologías de información y comunicación; así como de determinar, obtener y proporcionar los recursos necesarios para garantizar el funcionamiento integral del órgano contralor. (Contraloría General de la República, 2011)

1.1.1.4 Unidad de Tecnologías de Información (UTI)

Dentro de las unidades que conforman la DGA se encuentra la Unidad de Tecnologías de Información, para quien se desarrolla la propuesta de metodología de desarrollo de sistemas sobre la cual versa este proyecto.

Según lo establecido en el artículo 27 del Reglamento Orgánico de la Contraloría General de la República, las funciones de la UTI son las siguientes:

Es la unidad encargada de implementar, desarrollar y evolucionar soluciones tecnológicas y de comunicación, para apoyar y facilitar la ejecución de los procesos internos Para ello lidera el proceso de gestión de tecnologías de información y comunicación y participa del proceso de asesoría interna en materia de su competencia. (Contraloría General de la República, 2011)

La UTI está conformada por 27 colaboradores divididos en los siguientes procesos:

- Desarrollo de aplicaciones.
- Seguridad de la información.
- Infraestructura.
- Soporte Técnico.
- Presupuesto y Compras.

Del total de colaboradores de esta unidad, hay un total de 17 ingenieros, cinco funcionarios en labores asistenciales, una secretaria, tres supervisores y un jefe de unidad. El equipo de desarrollo de aplicaciones está conformado por diez de esos ingenieros y un supervisor.

En la figura dos se muestra la estructura organizacional de la UTI:

Estructura Organizacional de la Unidad de Tecnologías de Información

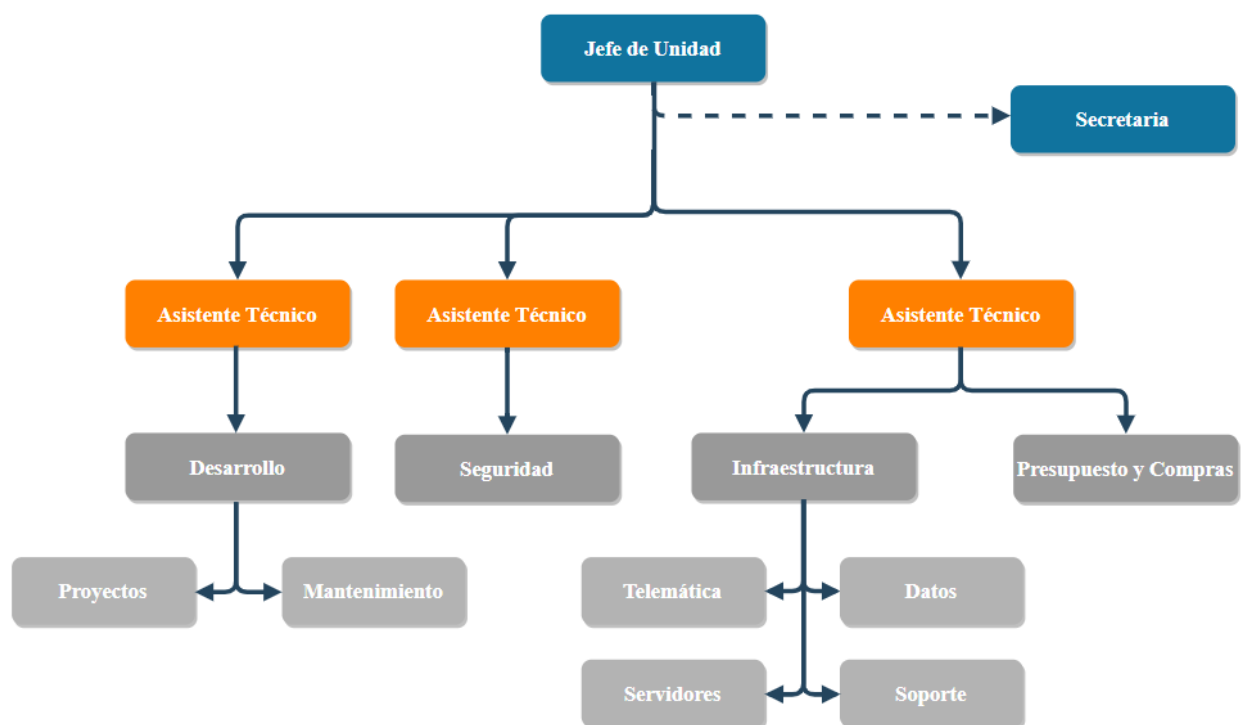


Figura 2: Estructura Organizacional de la Unidad de Tecnologías de Información.

Fuente: Elaboración propia.

1.1.2 Justificación del proyecto

Tal y como se resalta en la misión de la CGR, su principal tarea es la fiscalización para mejorar la gestión de la Hacienda Pública. En ese sentido, la institución ha definido dentro de su Plan Estratégico 2013-2020, como uno de sus objetivos el siguiente: “Transformar el proceso de fiscalización integral sustentándolo en tecnologías de información para incrementar su confiabilidad y oportunidad”. (Contraloría General de la República, 2013). Teniendo en cuenta lo anterior, es palpable que las tecnologías de información juegan un papel trascendental en el plan que está siguiendo la CGR actualmente.

Para lograr ese impacto en la fiscalización, resulta necesario hacer un uso óptimo de los limitados recursos con los que cuenta la institución, ante los grandes desafíos que tiene por delante. Asimismo, debe velarse por ofrecer soluciones tecnológicas modernas y actualizadas, que

colaboren en la automatización de procesos y permitan potenciar las tareas que se realizan actualmente.

El desarrollo de aplicaciones juega un papel fundamental en ese proceso. La institución, con los recursos que cuenta actualmente, debe realizar un esfuerzo en satisfacer de la mejor forma una mayor demanda institucional en materia de sistemas de información.

Dado lo anterior, la CGR debería contar con una metodología de desarrollo de sistemas de vanguardia, mediante el empleo de las mejores prácticas, lo que permitirá alcanzar los resultados de una manera ágil y con la calidad esperada.

Para contribuir con la situación descrita anteriormente, con el presente proyecto se busca coadyuvar a la consecución de los objetivos institucionales en materia de desarrollo de sistemas, proponiendo una actualización a la metodología de desarrollo de sistemas que utiliza la CGR hace más de una década. Además, se busca lograr una mayor satisfacción de las necesidades de los usuarios, con una mayor oportunidad, flexibilidad e innovación, sin dejar de lado los controles y documentación que deben llevar los proyectos de desarrollo.

1.2 DEFINICIÓN DEL PROBLEMA

1.2.1 Problemática

La Unidad de Tecnologías de Información debe velar porque las aplicaciones que desarrollen satisfagan los requerimientos de los usuarios y sean efectivas para apoyar las funciones que éstos desempeñen.

Para lograr esto, es necesario que la metodología de desarrollo de sistemas existente, brinde herramientas para facilitar un desenlace exitoso de los proyectos que se emprendan. En este punto reside la principal problemática por resolver con el presente proyecto.

Actualmente se cuenta con una Metodología de Desarrollo de Sistemas que tiene más de una década de existir. A pesar de que en el año 2016 se realizó una actualización, ésta no contempló aspectos de fondo, ni incorporó mejores prácticas para el desarrollo de aplicaciones.

Asimismo, los involucrados en el proceso de desarrollo de sistemas, no conocen la totalidad de la metodología, lo que provoca omisiones en tareas clave en todas las etapas del proyecto. Esto provoca que se dificulte comprobar si se cumplieron los requerimientos del usuario, si realmente hacen más eficientes las funciones para las cuales el sistema fue desarrollado, lo mismo que medir la cantidad de recursos invertidos, así como realizar labores de control en la gestión de los proyectos.

En adición a lo anterior, existen deficiencias en la documentación de los proyectos, ya que en algunos casos no se tiene claro su objetivo, por lo que se omite. Además, las herramientas propuestas en la metodología para una adecuada documentación pueden ser interpretada de diversas maneras, lo que implica que en algunos proyectos contengan altos niveles de detalle, mientras que en otros puede ser escasa.

En la parte técnica, es importante destacar que no se contempla la estandarización del código, lo que eventualmente favorecería la materialización de algunos riesgos importantes, principalmente si se dieran cambios en los equipos de desarrollo.

Las situaciones antes descritas repercuten de forma negativa en la razón de ser del proceso de desarrollo de sistemas, ya que podrían ofrecerse soluciones que no satisfagan las necesidades de los usuarios, que no estén alineados a los objetivos institucionales y que no generen el valor agregado que se espera recibir.

1.2.1.1 Diagrama de Ishikawa

A continuación, se ilustra el problema mediante un diagrama de Ishikawa, también conocido como “Causa – Efecto”

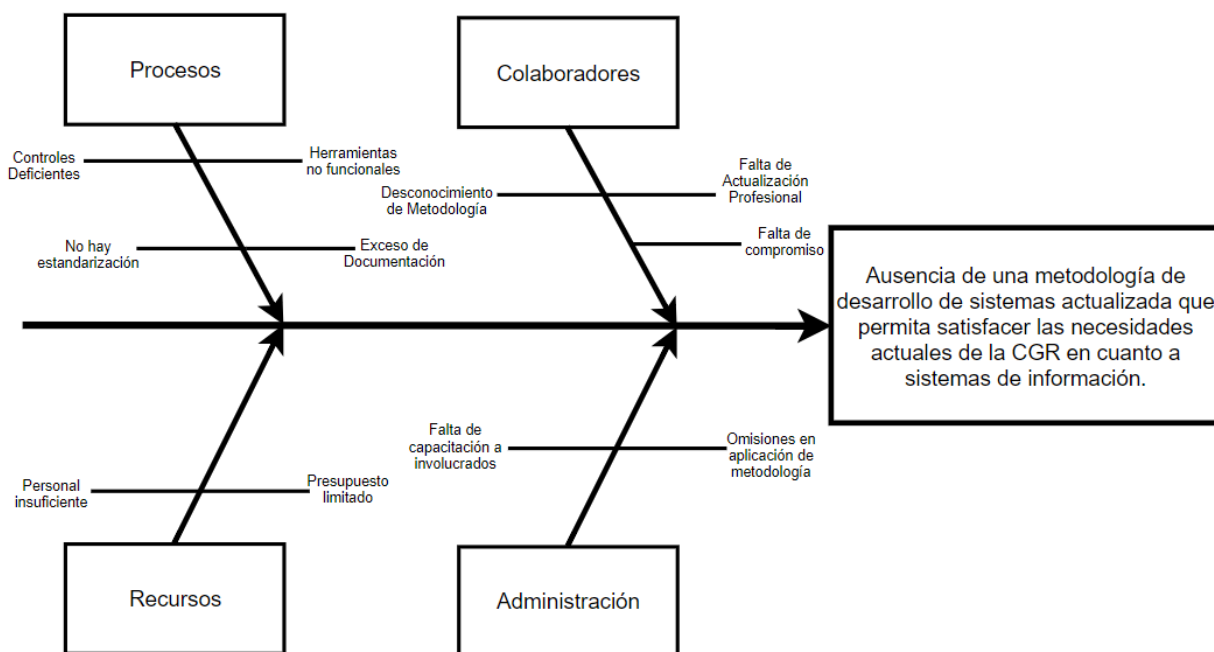


Figura 3: Diagrama de Ishikawa.

Fuente: Elaboración propia.

1.2.2 Problema General

¿Cómo podría la Contraloría General de la República (CGR) atender sus necesidades actuales en el proceso de desarrollo de sistemas de información?

1.2.3 Problemas Específicos

- ¿Cuáles son las necesidades actuales de la CGR en el proceso de desarrollo de sistemas de información?
- ¿De qué manera la metodología de desarrollo de sistemas que se utiliza actualmente en la CGR responde a las necesidades institucionales?
- ¿Qué tan alineada se encuentra la metodología de desarrollo de sistemas que se utiliza actualmente en la CGR, a las mejores prácticas, estándares y metodologías vigentes en la industria?

- ¿Cómo podría la CGR satisfacer las necesidades identificadas en el proceso de desarrollo de sistemas de información?

1.3 OBJETIVOS DEL PROYECTO

1.3.1 Objetivo general

Proponer una mejora a la metodología de desarrollo de sistemas de la Contraloría General de la República con el fin de que responda a las necesidades actuales de la Institución, mediante un análisis detallado del proceso de desarrollo de sistemas de información.

1.3.2 Objetivos específicos

- Diagnosticar la situación actual del proceso de desarrollo de sistemas de información, mediante la identificación procedimientos que se ejecutan actualmente y la normativa aplicable, con el fin de identificar las necesidades de la Institución.
- Ejecutar un análisis de la metodología de desarrollo de sistemas que se utiliza actualmente en la Institución, mediante la aplicación de técnicas para recolectar la información, con el fin de evaluar si responde a las necesidades institucionales.
- Identificar las mejores prácticas, estándares y metodologías para el desarrollo de *software* que podrían implementarse para satisfacer las necesidades de la institución, mediante un análisis comparativo.
- Desarrollar una propuesta de mejora a la metodología de desarrollo de sistemas de información que responda a las necesidades actuales de la Institución.

1.4 ALCANCES Y LIMITACIONES

1.4.1 Alcances

- El primer entregable del proyecto corresponde a un diagnóstico que describa la situación actual del proceso de desarrollo de sistemas de información de la CGR y el marco normativo que le aplica. Con esto se busca identificar las necesidades de la Institución en esa materia.

- El segundo entregable es un análisis de la metodología de desarrollo de sistemas que se utiliza actualmente en la Institución, con el fin de determinar si responde a las necesidades identificadas en el diagnóstico.
- Un tercer entregable corresponde a un análisis comparativo entre las metodologías, mejores prácticas y estándares más utilizados en la industria de desarrollo de aplicaciones.
- El último entregable es una propuesta de mejora a la metodología de desarrollo de sistemas de información que se utiliza actualmente, tomando en cuenta las necesidades identificadas y el análisis comparativo de mejores prácticas, estándares y metodologías más utilizadas en la industria.

1.4.2 Limitaciones

A continuación, se especifican las principales limitaciones para la ejecución del proyecto por categoría:

1.4.2.1 Limitaciones legales

- Al ser la CGR una institución pública, se rige bajo un marco normativo que podría limitar la implementación de algunas prácticas.
- La CGR cuenta con el Manual General de Fiscalización, así como otra normativa específica que rige todos los procesos que se realizan en la institución, por lo que la propuesta por elaborar, debe estar alineada a ese marco jurídico.

1.4.2.2 Limitaciones de tiempo

- Se hace una propuesta de metodología, sin embargo, no se implementará por el tiempo previsto para el desarrollo de este proyecto.
- El horario oficial de la CGR es de 7:30 a.m. a 3:30 p.m. por lo que para realizar entrevistas y recabar información presencial es necesario adaptarse a dicho horario.

1.4.2.3 Limitaciones organizacionales

- La propuesta de metodología debe adaptarse a los recursos actuales con que cuenta la CGR, especialmente el recurso humano, ya que por restricciones presupuestarias no se tiene contemplado abrir nuevas plazas para contratar más funcionarios.

1.5 CRONOGRAMA DE ACTIVIDADES

A continuación, se presenta el cronograma propuesto para el desarrollo del proyecto con un desglose de las principales actividades por ejecutar y su duración en semanas.

Actividad / Semanas	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21	
Planteamiento del Tema																						
Antecedentes y Justificación	■																					
Definición del Problema		■																				
Definición de Objetivos		■																				
Establecimiento de Alcances y Limitaciones			■																			
Definición del Cronograma			■																			
Elaboración del Marco Teórico																						
Consulta de fuentes				■																		
Desarrollo de Conceptos					■	■																
Definición del Marco Metodológico																						
Establecer tipo de investigación							■															
Definir fuentes de información							■															
Elegir técnicas de recolección de datos							■															
Definir categorías de análisis								■														
Diseño de la Investigación								■														
Completar Matriz de coherencia									■													
Ejecutar el diagnóstico de la situación actual																						
Descripción de la situación actual										■												
Recolección de datos											■	■										
Determinación de brechas													■									
Diseño y Desarrollo de Propuesta																						
Elaborar propuesta de Metodología de Desarrollo														■	■	■	■					
Conclusiones y Recomendaciones																						
Redactar conclusiones																				■		
Redactar recomendaciones																				■		
Revisión Final del Tutor																						
Revisión del Tutor																					■	■
Aplicación de ajustes sugeridos																					■	■

Figura 4: Cronograma de Actividades.

Fuente: Elaboración propia

CAPÍTULO II: MARCO TEÓRICO

En el presente capítulo se brinda al lector una base para la comprensión de lo que se abarca en el proyecto. Lo anterior, por medio de la selección y desarrollo de una serie de conceptos y teorías relacionadas con la temática principal.

Como parte de los conceptos que se desarrollan, se encuentra el de la fiscalización de la Hacienda Pública, el cual puede entenderse como el negocio principal de la CGR, y de ahí la importancia de que el lector pueda tener una noción sobre esta temática.

Adicional a lo anterior, y dado que el principal producto de este proyecto es una metodología, debe tenerse claro este concepto, el cual lleva también a entender qué es un proceso y un sistema, específicamente un sistema de información. En este tema se abarcarán estándares y mejores prácticas en el desarrollo de *software*, lo cual es ampliamente tratado en la metodología propuesta. Asimismo, se desarrollan conceptos referentes a la evaluación de sistemas de información.

Dentro de los conceptos administrativos que se abarcarán en este capítulo, se encuentra el de los Planes de Tecnologías de Información, así como su importancia para la organización. Sin embargo, los conceptos que más se amplían son los de gestión de proyectos, especialmente proyectos de tecnologías de información.

2.1 CONTROL EN LA HACIENDA PÚBLICA

Tal y como su nombre lo dice, la Contraloría General de la República es el órgano de Control de la Hacienda Pública. Para comprender mejor su razón de ser, es importante desarrollar el concepto de Control y de Hacienda Pública, los cuales se exponen a continuación.

2.1.1 Definición de Control

La Real Academia Española (2014) define control como “comprobación, inspección, fiscalización, intervención”. Como se aprecia, el término fiscalización está contenido en dicha definición, y este es uno de los más utilizados en la jerga de la CGR.

Más enfocado en la administración pública, se pueden encontrar la definición propuesta por la Organización Internacional de las Entidades Fiscalizadoras Superiores (INTOSAI), la cual indica lo siguiente:

“El control no representa una finalidad en sí mismo, sino una parte imprescindible de un mecanismo regulador que debe señalar, oportunamente, las desviaciones normativas y las infracciones de los principios de legalidad, rentabilidad, utilidad y racionalidad de las operaciones financieras, de tal modo que puedan adoptarse las medidas correctivas convenientes en cada caso, determinarse la responsabilidad del órgano culpable, exigirse la indemnización correspondiente o adoptarse las determinaciones que impidan o, por lo menos, dificulten, la repetición de tales infracciones en el futuro”. (Organización Internacional de las Entidades Fiscalizadoras Superiores, 2013)

A su vez, en el documento “La modernización del estado: El camino a (sic) seguir”, la Organización para la Cooperación Económica y el Desarrollo (OCDE) define control como:

“es el proceso diseñado para la adecuada comprobación de la eficiencia y eficacia de las operaciones, de la fiabilidad de la información proporcionada y del cumplimiento de la normativa vigente”. (Organización de Cooperación y Desarrollo Económico, 2006).

Como se aprecia en las definiciones anteriores, el control visto desde el ámbito del sector público, es un proceso con el cual se verifica que las operaciones o gestiones de una entidad, no se desvíen de su verdadera razón de ser.

En el artículo 1 de la Ley Orgánica de la Contraloría General de la República se define el rol de la CGR como órgano auxiliar de la Asamblea Legislativa en el control superior de la Hacienda Pública y rector del sistema de fiscalización, por lo que, ya abarcado el concepto de control, se desarrolla a continuación qué se entiende por Hacienda Pública.

2.1.2 Definición de Hacienda Pública

La definición de Hacienda Pública para el ordenamiento jurídico costarricense se encuentra en la Ley Orgánica de la Contraloría General de la República, precisamente en el artículo ocho donde se define de la siguiente manera:

“La Hacienda Pública estará constituida por los fondos públicos, las potestades para percibir, administrar, custodiar, conservar, manejar, gastar e invertir tales fondos y las normas jurídicas, administrativas y financieras, relativas al proceso presupuestario, la contratación administrativa, el control interno y externo y la responsabilidad de los funcionarios públicos.

Respecto a los entes públicos no estatales, las sociedades con participación minoritaria del sector público o las entidades privadas, únicamente formarán parte de la Hacienda Pública los recursos que administren o dispongan, por cualquier título, para conseguir sus fines y que hayan sido transferidos o puestos a su disposición, mediante norma o partida presupuestaria, por los Poderes del Estado, sus dependencias y órganos auxiliares, el Tribunal Supremo de Elecciones, la administración descentralizada, las universidades estatales, las municipalidades y los bancos del Estado. Los recursos de origen distinto de los indicados no integran la Hacienda Pública; en consecuencia, el régimen jurídico aplicable a esas entidades es el contenido en las Leyes que las crearon o los ordenamientos especiales que las regulan.

El patrimonio público será el universo constituido por los fondos públicos y los pasivos a cargo de los sujetos componentes de la Hacienda Pública. Serán sujetos componentes de la Hacienda Pública, el Estado y los demás entes u órganos públicos, estatales o no, y las empresas públicas, así como los sujetos de Derecho Privado, en cuanto administren o custodien fondos públicos por cualquier título, con las salvedades establecidas en el párrafo anterior”. (Asamblea Legislativa de la República de Costa Rica, 1994).

En la definición anterior, se aclara que el concepto de Hacienda Pública va más allá de sólo recursos económicos, incluye potestades, normativa, inclusive la responsabilidad de los funcionarios públicos. Con base en ello, puede comprenderse que el objeto de control de la CGR

va más allá de los recursos del Estado, abarca un ámbito más complejo que incluye mayor número de elementos.

2.1.2 Entidades de Fiscalización Superior (EFS)

Los órganos encargados de velar por el control de la hacienda pública en un país, son denominados Entidades de Fiscalización Superior. Por ejemplo, la EFS de Costa Rica es la Contraloría General de la República. No necesariamente en todos los países reciben el nombre de Contralorías, ya que por ejemplo en Brasil y en algunos países europeos recibe el nombre de Tribunal de Cuentas.

La OCDE define a las EFS de la siguiente forma:

“Las Entidades Fiscalizadoras Superiores (EFS) son los principales organismos de auditoría del sector público en un país. Su tarea principal consiste en examinar si los recursos públicos son gastados de forma económica, eficiente y eficaz, de conformidad con las normas y reglamentos vigentes. El buen funcionamiento de las EFS es importante para confirmar que los controles están operando de manera efectiva, identificar el despilfarro y sugerir formas en las que las entidades gubernamentales pueden funcionar mejor. Para ello, elaboran informes de auditoría objetivos y rigurosos, destinados a lograr un cambio positivo en la forma en que los gobiernos administran los recursos públicos. Al asegurar que el dinero se gaste bien, las EFS pueden contribuir al desarrollo nacional sostenible y a reducir el fraude y la pobreza. Una EFS eficaz puede hacer más fácil el trabajo de los socios para el desarrollo, al garantizar que los fondos del socio se están utilizando para los fines propuestos y que los objetivos y resultados acordados están siendo alcanzados. Las EFS son una parte clave del sistema de gestión financiera del sector público de un país, ya que brindan tranquilidad a los parlamentos, ciudadanos y socios para el desarrollo, entre otros, de que los gobiernos están manejando de manera apropiada los fondos públicos”. (Organización para la Cooperación y el Desarrollo Económicos, 2011, p.11).

Tal y como se aprecia en el párrafo anterior, más allá de controlar el uso de los recursos públicos, las EFS contribuyen al desarrollo de un país, promoviendo la movilidad social y el bienestar de la sociedad.

Las EFS se encuentran organizadas a nivel mundial por medio de la Organización Internacional de las Entidades Fiscalizadoras Superiores (INTOSAI). Asimismo, existe una serie de organizaciones regionales de EFS tales como:

- Organización de Entidades Fiscalizadoras Superiores de África (AFROSAI)
- Organización Árabe de Entidades Fiscalizadoras Superiores (ARABOSAI)
- Organización de Entidades Fiscalizadoras Superiores del Caribe (CAROSAI)
- Organización de las Entidades Fiscalizadoras Superiores de Europa (EUROSAI)
- Organización Latinoamericana y del Caribe de Entidades Fiscalizadoras Superiores (OLACEFS)
- Asociación de Entidades Fiscalizadoras Superiores del Pacífico (PASAI)

2.2 SISTEMAS

Los conceptos de sistema y sistemas de información se traen a colación dado que al formularse en este proyecto una metodología para su desarrollo, deben estar debidamente fundamentados.

2.2.1 Generalidades de los sistemas

Pese a que en todo lo que se encuentra alrededor pueden encontrarse distintos tipos de sistemas, en algunas ocasiones este concepto se delimita solamente a sistemas de información. Es por eso que, en los siguientes acápite, se explica el concepto de sistemas en general, así como sus características.

2.2.1.1 Concepto de sistema

Tomando en consideración lo establecido en el diccionario de la Real Academia Española (2014) se define como un sistema como un “Conjunto de cosas que relacionadas entre sí ordenadamente contribuyen a determinado objeto”.

Existen múltiples tipos de sistemas, con el factor común de que son un conjunto de elementos relacionados entre sí para la consecución de un objetivo. Como se apreciará más adelante, aplica también para los sistemas de información.

Otra definición en la misma línea es la que es la que plantea Domínguez (2012) a continuación:

“podemos decir que un sistema es la organización de partes interactuantes e interdependientes que se encuentran unidas y relacionadas para formar una célula compleja.

Con esto nos referimos a un grupo de elementos que realizan actividades para alcanzar un objetivo común, ya sea operando sobre los datos, la energía o la materia para suministrar información”. (p. 10).

2.2.1.2 Características de los sistemas

El mundo actual, y se puede decir en un sentido más amplio que el universo, está conformado por una serie de sistemas, los cuales, según Domínguez (2015) comparten las siguientes características:

- “Propósito u objetivo: todo sistema tiene uno o varios propósitos u objetivos y sus elementos tratan siempre de cumplir esos objetivos.
- Ambiente: cuando nos referimos a esta característica estamos hablando de todo lo externo con respecto al sistema. El sistema ejerce una influencia casi nula sobre dicho ambiente, y solamente interviene sobre él cuando requiere materia, energía o información.
- Recursos: no son más que todos los medios que pueden ser utilizados por el sistema para cumplir sus objetivos. Los recursos que el sistema puede o no tomar para su beneficio se encuentran en el ambiente.
- Componentes: cuando hablamos de los componentes nos referimos a las tareas o actividades que se puedan llevar a cabo para el cumplimiento de los objetivos.
- La administración: tiene dos funciones elementales: 1. La planificación, donde se cubren todos los aspectos relevantes, como: objetivos, ambiente, utilización de

recursos, sus componentes y actividades. 2. El control, que consiste en el análisis de los planes y la planificación de algunos cambios.

- Globalismo o totalidad: en todo sistema encontramos una naturaleza orgánica en la que cada acción provoca un cambio en una o más unidades del sistema. Tomando en cuenta que existe una relación de interdependencia entre las partes de un sistema, con cada nueva acción el sistema sufre algún cambio y el ajuste sistemático es continuo”. (Dominguez, L, 2012, p. 13).

Como bien se aprecia en dichas características, los sistemas nacen para cumplir un objetivo con base en la relación de los elementos que lo componen, los cuales a través de una serie de cambios se enfocan en alcanzar esa finalidad para la que fueron creados.

2.2.2 Sistemas de Información

Con la definición de sistema clara, es necesario profundizar en los sistemas de información, abarcando generalidades, componentes, desarrollo, entre otros conceptos que se describen a continuación.

2.2.2.1 Generalidades de un Sistema de Información

Existen múltiples definiciones sobre sistemas de información, sin embargo, para el desarrollo del presente proyecto, se seleccionaron las siguientes:

“Un sistema de información está integrado de una gran variedad de elementos que se interrelacionan entre sí con el fin de apoyar las actividades de una empresa o negocio”. (Dominguez, L, 2012, p. 34).

Otra definición es la propuesta por Laudon, K. y Laudon, J. (2016), la cual se indica a continuación:

“Conjunto de componentes interrelacionados que recolectan (o recuperan), procesan, almacenan y distribuyen información para apoyar los procesos de toma de decisiones y de control en una organización. Además de apoyar la toma de decisiones, la coordinación y el control, los sistemas de información también pueden ayudar a los gerentes y trabajadores

del conocimiento a analizar problemas, visualizar temas complejos y crear nuevos productos”. (p.16).

Ambas definiciones coinciden en que el objetivo de los Sistemas de Información es apoyar la toma de decisiones a nivel de una organización, y es en parte lo que se vive en la CGR, la que, para cumplir su rol como órgano de control, cuenta con una serie de sistemas de información que coadyuvan en potenciar sus funciones. Las necesidades de esos sistemas son cada vez mayores dado que se ha acelerado la digitalización de procesos y cada vez aumenta la cantidad de información.

Para lograr desarrollar un sistema de información exitoso y útil para las organizaciones, es imprescindible que estos estén alineados con la estrategia de la organización, buscando que la inversión de recursos que se realiza devuelva réditos positivos. Asimismo, es necesario considerar una serie de factores, los cuales, de acuerdo con Laudon (2016) son los siguientes:

“... los factores organizacionales que se deben considerar al planear un nuevo sistema son:

- El entorno en que debe funcionar la organización
- La estructura de la organización: jerarquía, especialización, rutinas y procesos de negocios
- La cultura y las políticas de la organización
- El tipo de organización y su estilo de liderazgo
- Los principales grupos de interés afectados por el sistema y las posturas de los trabajadores que utilizarán el sistema
- Los tipos de tareas, decisiones y procesos de negocios en los que el sistema de información está diseñado para ayudar”. (p.94).

2.2.2.2 Clasificación de los Sistemas de Información

Tal y como se ha mencionado en las diversas definiciones, los sistemas de información son un apoyo en distintos procesos de las organizaciones, tales como la producción, control, toma de decisiones, entre otros. De acuerdo con la finalidad para la que son implementados los sistemas, pueden clasificarse en distintos tipos.

Según Domínguez (2016), “los sistemas de información se clasifican en:

- Sistemas transaccionales
- Sistemas para la gestión de información
- Sistemas de información ejecutiva
- Sistemas de apoyo a las decisiones
- Sistemas expertos”. (p.34).

2.2.2.3 Componentes de un Sistema de Información

De acuerdo con Stair (2016), los componentes de un sistema de información son las entradas, el procesamiento y las salidas tal y como se muestra en la siguiente figura:

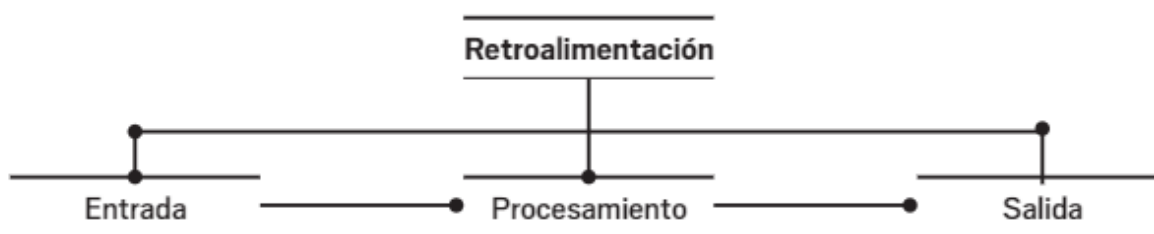


Figura 5: Componentes de un sistema de información según Stair

Fuente: Principios de sistemas de información (2016)

La entrada se refiere a todo aquel dato que se recopila o captura. Esas entradas son convertidas en salidas útiles por el procesamiento. La característica de la salida es que es información útil. Mientras que la retroalimentación es aquella información que se utiliza para realizar cambios en la entrada o en actividades de procesamiento.

Hay una máxima utilizada conocida en el mundo de los sistemas de información que indica que, si las entradas del sistema son basura, las salidas también serán basura. De ahí la importancia de poner las validaciones necesarias con el fin de que la información que ingresa al sistema sea correcta. De igual manera es importante realizar suficientes pruebas para verificar que el procesamiento se esté efectuando de manera adecuada. Con respecto a las salidas, deben verificarse siempre con los usuarios para determinar si es información de utilidad, pero sobre todo

si es lo que realmente necesitaban del sistema. Para considerar la información útil debe cumplir una serie de características, las cuales de acuerdo con Sousa, K y Oz, E, (2017) son las siguientes:

- “Relevante: La información debe relacionarse con el problema en cuestión. Debe presentarse de modo que ayude a una persona a comprenderla en un contexto específico.
- Completa: La información parcial suele ser peor que ninguna información.
- Precisa: La información errónea puede conducir a decisiones desastrosas.
- Actual: Las decisiones suelen basarse en la información disponible más reciente, pero un hecho de ayer tal vez ya no sirva hoy.
- Económica: En un ambiente de negocios, el costo de obtener información debe considerarse como un elemento del costo relacionado con cualquier decisión”. (p.10).

2.2.2.4 Desarrollo de un Sistema de Información

La metodología que se abarca con este proyecto está destinada a brindar un marco orientador en el proceso de desarrollo de sistemas en la Contraloría. Como consecuencia de lo anterior debe comprenderse que se entiende por desarrollo de sistemas, lo que Stair, R. y Reynolds, G, (2017) definen de la siguiente manera:

“El desarrollo de sistemas se define como la actividad que consiste en crear o modificar los sistemas de información. El desarrollo de sistemas se inició por muchas razones: reducir los costos y esfuerzos asociados con la operación de un sistema ya existente; satisfacer una nueva necesidad de negocio causada por cambios en la organización; cumplir con un nuevo requisito gubernamental; proporcionar un servicio nuevo o mejorado al cliente, o aprovechar el desarrollo de nuevas tecnologías”. (p.35).

Existen diversas visiones sobre el ciclo de vida del desarrollo de sistemas, sin embargo, la mayoría coincide en que existen etapas de Análisis, Diseño y Desarrollo, así como las Pruebas, la Implementación y el Mantenimiento.

En la siguiente figura, se expone una de las teorías del ciclo de vida del desarrollo de sistemas, según la visión de Stair:

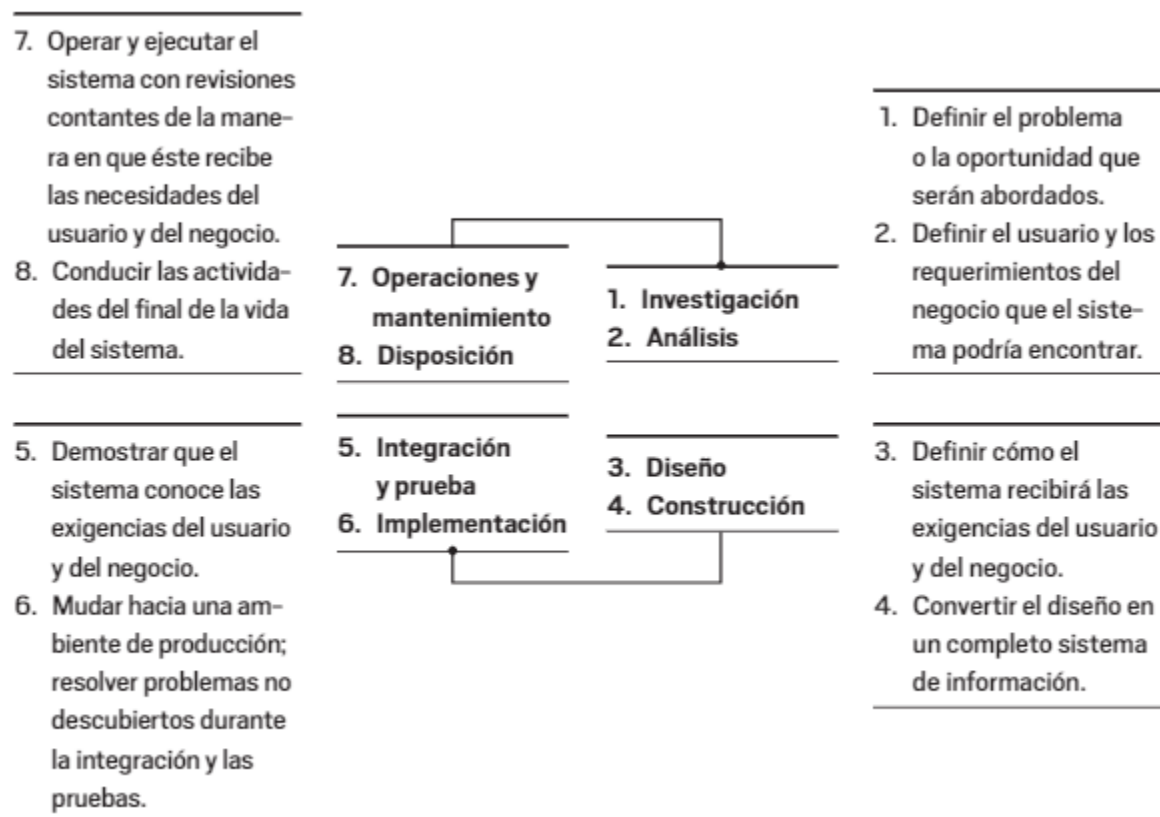


Figura 6: Ciclo de vida del desarrollo de sistemas de Stair

Fuente: Principios de sistemas de información (2016)

Reforzando lo anterior, Sommerville (2011) indica que pese a que existen muchos procesos para la creación de software, todos deben incluir cuatro actividades fundamentales que son las siguientes:

1. “Especificación del software Tienen que definirse tanto la funcionalidad del software como las restricciones de su operación.
2. Diseño e implementación del software Debe desarrollarse el software para cumplir con las especificaciones.
3. Validación del software Hay que validar el software para asegurarse de que cumple lo que el cliente quiere.

4. Evolución del software El software tiene que evolucionar para satisfacer las necesidades cambiantes del cliente”. (p.28).

Como complemento a las etapas descritas anteriormente, cabe agregar las descritas por Pressman (2010):

“La estructura del proceso establece el fundamento para el proceso completo de la ingeniería de software por medio de la identificación de un número pequeño de actividades estructurales que sean aplicables a todos los proyectos de software, sin importar su tamaño o complejidad.

Comunicación. Antes de que comience cualquier trabajo técnico, tiene importancia crítica comunicarse y colaborar con el cliente (y con otros participantes).¹¹ Se busca entender los objetivos de los participantes respecto del proyecto, y reunir los requerimientos que ayuden a definir las características y funciones del software. **Planeación.** Cualquier viaje complicado se simplifica si existe un mapa. Un proyecto de software es un viaje difícil, y la actividad de planeación crea un “mapa” que guía al equipo mientras viaja. El mapa —llamado plan del proyecto de software— define el trabajo de ingeniería de software al describir las tareas técnicas por realizar, los riesgos probables, los recursos que se requieren, los productos del trabajo que se obtendrán y una programación de las actividades.

Modelado. Ya sea usted diseñador de paisaje, constructor de puentes, ingeniero aeronáutico, carpintero o arquitecto, a diario trabaja con modelos. Crea un “bosquejo” del objeto por hacer a fin de entender el panorama general —cómo se verá arquitectónicamente, cómo ajustan entre sí las partes constituyentes y muchas características más—. Si se requiere, refina el bosquejo con más y más detalles en un esfuerzo por comprender mejor el problema y cómo resolverlo. Un ingeniero de software hace lo mismo al crear modelos a fin de entender mejor los requerimientos del software y el diseño que los satisfará.

Construcción. Esta actividad combina la generación de código (ya sea manual o automatizada) y las pruebas que se requieren para descubrir errores en éste.

Despliegue. El software (como entidad completa o como un incremento parcialmente terminado) se entrega al consumidor que lo evalúa y que le da retroalimentación, misma que se basa en dicha evaluación”. (p.13).

Pese a que pueden encontrarse diversos modelos de ingeniería de software, la mayoría siempre va a coincidir con que debe resolverse un problema para la organización, el cual debe tenerse claramente definido; que debe diseñarse o modelarse cómo se resolverá ese problema a través de un sistema; que debe construirse la solución por medio del desarrollo del sistema; que debe validarse que el sistema satisfaga las necesidades planteadas en el sistema y si es así implementarse; que se le brinde mantenimiento al sistema a lo largo del tiempo para que siga siendo funcional.

2.2.2.5 Evaluación del desarrollo de Sistemas de Información

Tal y como se abarcó anteriormente, el desarrollo de un sistema de información se ejecuta con el fin de satisfacer una necesidad de la organización. Con el fin de lograr obtener éxito en el proceso es recomendable que se evalúe el proceso de desarrollo.

Para ese fin, uno de los marcos referencias más utilizados es COBIT, el cual ofrece una serie de buenas prácticas para la gestión y el gobierno de tecnologías de información en las organizaciones.

COBIT 5, que es la versión más reciente de este marco de referencia, incluye el proceso BAI03 Gestionar la Identificación y Construcción de Soluciones, dentro del cual ofrece una serie de métricas relacionadas que es de utilidad para evaluar el proceso de desarrollo. Este proceso, perteneciente al dominio de Construir, Adquirir e Implementar, se define de la siguiente manera:

“Establecer y mantener soluciones identificadas en línea con los requerimientos de la empresa que abarcan el diseño, desarrollo, compras/contratación y asociación con proveedores/fabricantes. Gestionar la configuración, preparación de pruebas, realización de pruebas, gestión de requerimientos y mantenimiento de procesos de negocio, aplicaciones, datos/información, infraestructura y servicios”. (ISACA, 2012, p. 133).

En la tabla que se muestra a continuación se exponen las métricas expuestas por COBIT 5, las cuales permiten realizar una evaluación del desarrollo de sistemas:

Tabla 1. Métricas relacionadas al Proceso: Gestionar la Identificación y Construcción de Soluciones de COBIT 5

Meta	Métricas Relacionadas
Entrega de servicios de TI de acuerdo con los requisitos del negocio.	<ul style="list-style-type: none"> • Número de interrupciones del negocio debidas a incidentes en el servicio de TI. • Porcentaje de partes interesadas satisfechas con el cumplimiento del servicio de TI entregado respecto a los niveles de servicio acordados. • Porcentaje de usuarios satisfechos con la calidad de los servicios de TI entregados.
El diseño de la solución, incluyendo los componentes relevantes, debe cumplir con las necesidades de la empresa, alineándose con estándares y tratando todos los riesgos identificados.	<ul style="list-style-type: none"> • Número de rediseños realizados debido a discordancias con los requerimientos. • Tiempo para aprobar que el entregable de diseño ha cumplido los requerimientos.
La solución conforme con el diseño, es acorde con las normas organizativas y cuenta con controles, seguridad y ‘auditabilidad’ apropiadas.	<ul style="list-style-type: none"> • Número de excepciones al diseño observadas durante la fase de revisión.
La solución es de una calidad aceptable y ha sido probada convenientemente.	<ul style="list-style-type: none"> • Número de errores encontrados durante las pruebas. • Tiempo y esfuerzo para completar las pruebas.

Los cambios aprobados de los requerimientos están correctamente incorporadas a la solución. • Número de cambios aprobados y registrados que generan nuevos errores.

Las actividades de mantenimiento cumplen satisfactoriamente con las necesidades tecnológicas y de negocio. • Número de solicitudes de mantenimiento no atendidas.

Fuente: (ISACA, 2012)

Pese a que COBIT ofrece las métricas descritas anteriormente, es necesario que la organización defina dentro de sus procesos la evaluación de los sistemas y que implemente una serie de controles con el fin de poder medir el cumplimiento de las metas propuestas para el desarrollo del sistema.

2.3 METODOLOGÍAS DE DESARROLLO DE SISTEMAS DE INFORMACIÓN

Para el desarrollo de software existen diversas metodologías, las cuales deberían adaptarse de acuerdo con las necesidades y capacidades de las empresas.

En esta sección se expondrán cuáles son las principales metodologías de desarrollo de sistemas, pero antes, es necesario aclarar el concepto de metodología.

2.3.1 Concepto de Metodología

Según el diccionario de la Real Academia Española (2014), la metodología proviene de la palabra método, que se define de la siguiente manera: “Modo de obrar o proceder, hábito o costumbre que cada uno tiene y observa”.

Otra definición relacionada es la siguiente:

“La expresión metodología hace referencia al conjunto de métodos y procedimientos racionales articulados, utilizados para alcanzar objetivos que rigen a una investigación científica, o a la implantación de proyectos.

Alternativamente puede definirse la metodología como el estudio o elección de una secuencia de métodos pertinentes para un determinado objetivo.

Método es una palabra que proviene del término griego *methodos* (camino o vía) y que se refiere al medio utilizado para llegar a un fin. Su significado original señala el camino que conduce a un lugar”. (Castro, C, 2013, p. 10).

Con base en las definiciones anteriores, se entiende por metodología, una serie de procesos estructurados que se adoptan para conseguir un objetivo. Para el presente proyecto, se estudian metodologías desde dos vertientes, la primera para el desarrollo de software desde la perspectiva técnica, y la segunda enfocada desde la gestión de proyectos.

2.3.2 Metodologías tradicionales de desarrollo de sistemas

Para la presente investigación se entiende como Metodología Tradicional, a aquellas metodologías de desarrollo de software que no son ágiles, en los cuales se da prioridad al control de los procesos por medio de la definición de roles, actividades, tareas, con una documentación detallada.

Según Chiquito (2017):

“Las metodologías clásicas o tradicionales se enfocan en la planificación total del trabajo a realizar, una vez que se completa la descripción de los detalles, se empieza con el ciclo del desarrollo del producto de software.

Entre sus características principales se encuentran:

Permanencia en el entorno: El ambiente en el que se desarrollan los proyectos ya tiene definidas sus características, siguen un patrón que son repetidos constantemente.

Predecibles: Esta metodología está basada en el uso de herramientas de planificación y control, siendo (sic) así todos los esfuerzos apuntan a cumplir con los tiempos, costos y recursos”. (p. 24).

Como se aprecia en las descripciones anteriores, las metodologías tradicionales llevan un control de todas las etapas de desarrollo, lo que las hace concebirse como burocráticas, rígidas y con altos costos. Sin embargo, también ofrecen ventajas, ya que permiten contar con documentación que brinda información importante en cada etapa del proyecto, dándole formalidad, seguridad y fiabilidad a los involucrados.

En los siguientes párrafos se expondrán las principales metodologías tradicionales.

2.3.2.1 Modelo de cascada

El modelo “cascada” se denomina de esa manera debido a que se va dando un paso fase por fase, por lo que se hace la metáfora con una cascada. En esta metodología, no se puede iniciar con una nueva fase hasta que se haya finalizado la previa.

Se caracteriza ya que sus fases y tareas son dirigidas por un plan, el cual se elabora antes de iniciar el proceso de desarrollo.

Pressman (2010) define el proceso de cascada de la siguiente manera:

“El modelo de la cascada, a veces llamado ciclo de vida clásico, sugiere un enfoque sistemático y secuencial para el desarrollo del software, que comienza con la especificación de los requerimientos por parte del cliente y avanza a través de planeación, modelado, construcción y despliegue, para concluir con el apoyo del software terminado”. (p.34).

El modelo cascada cuenta con una serie de fases, las cuales Pressman resume en la figura 7, que se muestra a continuación:

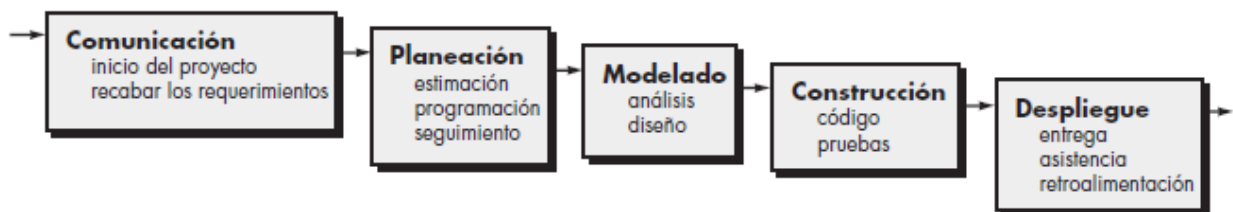


Figura 7: Modelo de la cascada

Fuente: Ingeniería de Software: Un enfoque práctico (2010)

Por su parte, Sommerville (2011) define las principales etapas del modelo en cascada de la siguiente manera:

- “1. Análisis y definición de requerimientos Los servicios, las restricciones y las metas del sistema se establecen mediante consulta a los usuarios del sistema. Luego, se definen con detalle y sirven como una especificación del sistema.
2. Diseño del sistema y del software El proceso de diseño de sistemas asigna los requerimientos, para sistemas de hardware o de software, al establecer una arquitectura de sistema global. El diseño del software implica identificar y describir las abstracciones fundamentales del sistema de software y sus relaciones.
3. Implementación y prueba de unidad Durante esta etapa, el diseño de software se realiza como un conjunto de programas o unidades del programa. La prueba de unidad consiste en verificar que cada unidad cumpla con su especificación.
4. Integración y prueba de sistema Las unidades del programa o los programas individuales se integran y prueban como un sistema completo para asegurarse de que se cumplan los requerimientos de software. Después de probarlo, se libera el sistema de software al cliente.
5. Operación y mantenimiento Por lo general (aunque no necesariamente), ésta es la fase más larga del ciclo de vida, donde el sistema se instala y se pone en práctica. El mantenimiento incluye corregir los errores que no se detectaron en etapas anteriores del ciclo de vida, mejorar la implementación de las unidades del sistema e incrementar los servicios del sistema conforme se descubren nuevos requerimientos”. (p.31).

Como es característico en este modelo, por cada una de las fases descritas se crea uno o más documentos formales. Por esta característica, se ha criticado este modelo debido a que, si se produce un cambio posterior, es necesario ir a modificar toda la documentación, lo que lo hace poco eficiente.

Pressman (2010) identificó los siguientes problemas del modelo cascada:

“1. Es raro que los proyectos reales sigan el flujo secuencial propuesto por el modelo. Aunque el modelo lineal acepta repeticiones, lo hace en forma indirecta. Como resultado, los cambios generan confusión conforme el equipo del proyecto avanza.

2. A menudo, es difícil para el cliente enunciar en forma explícita todos los requerimientos. El modelo de la cascada necesita que se haga y tiene dificultades para aceptar la incertidumbre natural que existe al principio de muchos proyectos.

3. El cliente debe tener paciencia. No se dispondrá de una versión funcional del (de los) programa(s) hasta que el proyecto esté muy avanzado. Un error grande sería desastroso si (sic) se detectara hasta revisar el programa en funcionamiento”. (p.34).

Con base en la información anterior, se desprende que el modelo de cascada resultaría útil en procesos en los que debe prevalecer la formalidad y la documentación ante otros factores como eficiencia, eficacia y economía. Asimismo, resulta útil para procesos que cuentan con los requerimientos bien definidos con bajo riesgo de cambio, en los que se avanza de forma lineal de inicio a fin. Sin embargo, por la dinámica que tiene el desarrollo de software actualmente, en los que constantemente se presentan cambios y debe llevarse un ritmo acelerado, el modelo cascada no se adapta a estas características por lo que su aplicación podría resultar inapropiada.

2.3.2.2 Modelo incremental

Tal y como su nombre lo indica, el modelo incremental se basa en una entrega inicial del producto con funciones básicas, a las que con el tiempo se le irán sumando nuevas características.

Sommerville (2011) describe el modelo de desarrollo incremental de la siguiente manera:

“El desarrollo incremental se basa en la idea de diseñar una implementación inicial, exponer ésta al comentario del usuario, y luego desarrollarla en sus diversas versiones hasta producir un sistema adecuado. Las actividades de especificación, desarrollo y validación están entrelazadas en vez de separadas, con rápida retroalimentación a través de las actividades”. (p.33).

Un elemento clave, tal y como se aprecia en la definición anterior, es que las entregas separadas permiten una retroalimentación del producto, lo que permite evaluar si el proceso está encaminado a satisfacer las demandas del usuario.

Por su parte, Pressman (2010) indica lo siguiente sobre el modelo incremental:

Cuando se utiliza un modelo incremental, es frecuente que el primer incremento sea el producto fundamental. Es decir, se abordan los requerimientos básicos, pero no se proporcionan muchas características suplementarias (algunas conocidas y otras no). El cliente usa el producto fundamental (o lo somete a una evaluación detallada). Como resultado del uso y/o evaluación, se desarrolla un plan para el incremento que sigue. El plan incluye la modificación del producto fundamental para cumplir mejor las necesidades del cliente, así como la entrega de características adicionales y más funcionalidad. Este proceso se repite después de entregar cada incremento, hasta terminar el producto final.

El modelo de proceso incremental se centra en que en cada incremento se entrega un producto que ya opera. Los primeros incrementos son versiones desnudas del producto final, pero proporcionan capacidad que sirve al usuario y también le dan una plataforma de evaluación. (p.36)

De acuerdo con lo descrito por Pressman, el modelo incremental también obedece a un plan, el cual se va repitiendo conforme se va desarrollando cada uno de los entregables, como se muestra en la siguiente figura:

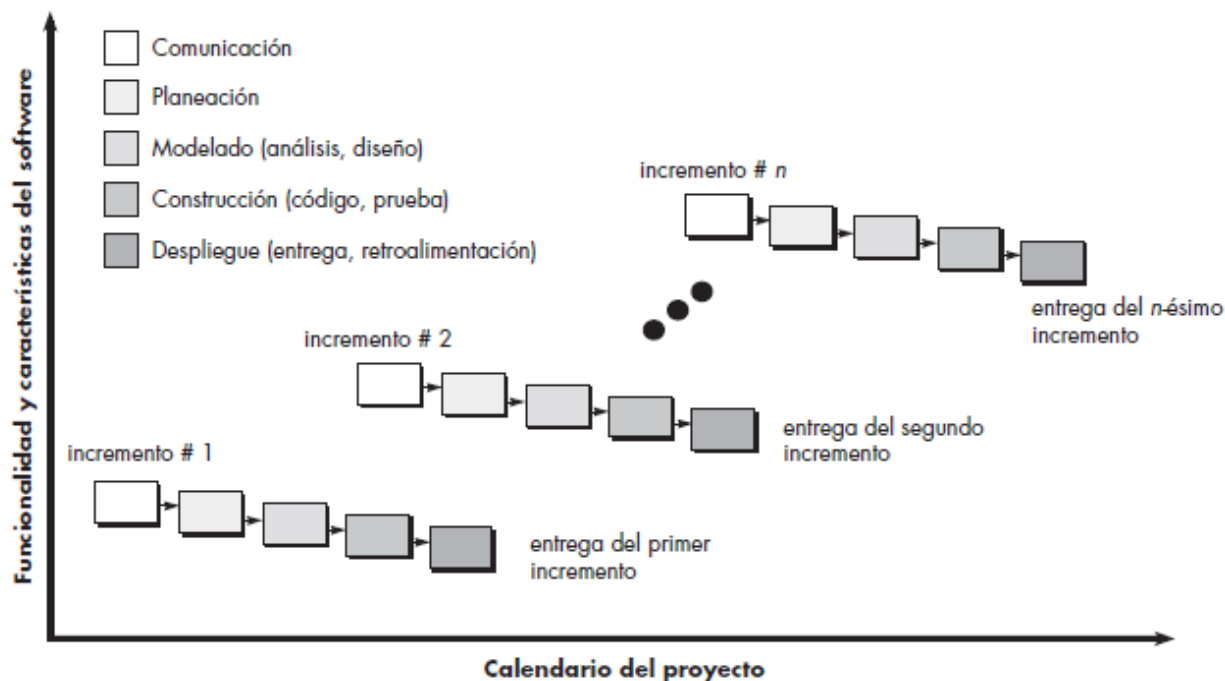


Figura 8: Modelo Incremental

Fuente: Ingeniería de Software: Un enfoque práctico (2010)

Dentro de las ventajas identificadas en este modelo, y como se mencionó previamente, está que el cliente tiene la posibilidad de evaluar el primer incremento, lo que permitiría redefinir requerimientos para futuros incrementos. Asimismo, con los primeros incrementos ya tendrían un software con los elementos básicos, que haría que las funciones primordiales del sistema, al ser entregadas de primero, reciban mayor cantidad de pruebas.

Pese a lo anterior, el modelo presenta algunos problemas, los cuales Pressman (2010) define de la siguiente manera:

- “1. La mayoría de los sistemas requieren de una serie de recursos que se utilizan para diferentes partes del sistema. Dado que los requerimientos no están definidos con detalle sino hasta que se implementa un incremento, resulta difícil identificar recursos comunes que necesiten todos los incrementos.
2. Asimismo, el desarrollo iterativo resulta complicado cuando se diseña un sistema de reemplazo. Los usuarios requieren de toda la funcionalidad del sistema antiguo, ya que es

común que no deseen experimentar con un nuevo sistema incompleto. Por lo tanto, es difícil conseguir retroalimentación útil del cliente.

3. La esencia de los procesos iterativos es que la especificación se desarrolla en conjunto con el software. Sin embargo, esto se puede contradecir con el modelo de adquisiciones de muchas organizaciones, donde la especificación completa del sistema es parte del contrato de desarrollo del sistema. En el enfoque incremental, no hay especificación completa del sistema, sino hasta que se define el incremento final. Esto requiere una nueva forma de contrato que los grandes clientes, como las agencias gubernamentales, encontrarían difícil de adoptar”. (p.48).

Deben analizarse las características de la empresa, así como las del proyecto para determinar si se adaptan a un modelo incremental, especialmente en la gestión de recursos, la gestión de adquisiciones, y las características del software que se está desarrollando, para determinar si es posible realizar pruebas parciales o se necesita probar el mismo con todas sus funcionalidades.

2.3.2.3 Proceso Unificado Racional (RUP)

Este modelo es una combinación de varios modelos tradicionales, incluyendo el modelo cascada e incremental que se describieron en las secciones anteriores, por lo tanto, se caracteriza por ser un modelo de proceso híbrido.

RUP tiene su origen en el lenguaje de modelado unificado, conocido como UML por sus siglas en inglés. Fue desarrollado por James Rumbaugh, Grady Booch e Ivar Jacobson a inicios de la década de 1990.

El proceso unificado racional, también es conocido como Proceso Unificado (PU). Este modelo, cuenta con cinco fases, que según Pressman (2010) son las siguientes:

“La fase de concepción del PU agrupa actividades tanto de comunicación con el cliente como de planeación. Al colaborar con los participantes, se identifican los requerimientos del negocio, se propone una arquitectura aproximada para el sistema y se desarrolla un plan para la naturaleza iterativa e incremental del proyecto en cuestión. Los requerimientos

fundamentales del negocio se describen por medio de un conjunto de casos de uso preliminares que detallan las características y funciones que desea cada clase principal de usuarios. En este punto, la arquitectura no es más que un lineamiento tentativo de subsistemas principales y la función y rasgos que tienen. La arquitectura se mejorará después y se expandirá en un conjunto de modelos que representarán distintos puntos de vista del sistema. La planeación identifica los recursos, evalúa los riesgos principales, define un programa de actividades y establece una base para las fases que se van a aplicar a medida que avanza el incremento del software.

La fase de elaboración incluye las actividades de comunicación y modelado del modelo general del proceso. La elaboración mejora y amplía los casos de uso preliminares desarrollados como parte de la fase de concepción y aumenta la representación de la arquitectura para incluir cinco puntos de vista distintos del software: los modelos del caso de uso, de requerimientos, del diseño, de la implementación y del despliegue. En ciertos casos, la elaboración crea una “línea de base de la arquitectura ejecutable” que representa un sistema ejecutable de “primer corte”. La línea de base de la arquitectura demuestra la viabilidad de ésta, pero no proporciona todas las características y funciones que se requieren para usar el sistema. Además, al terminar la fase de elaboración se revisa con cuidado el plan, a fin de asegurar que el alcance, riesgos y fechas de entrega siguen siendo razonables. Es frecuente que en este momento se hagan modificaciones al plan.

La fase de construcción del PU es idéntica a la actividad de construcción definida para el proceso general del software. Con el uso del modelo de arquitectura como entrada, la fase de construcción desarrolla o adquiere los componentes del software que harán que cada caso de uso sea operativo para los usuarios finales. Para lograrlo, se completan los modelos de requerimientos y diseño que se comenzaron durante la fase de elaboración, a fin de que reflejen la versión final del incremento de software. Después se implementan en código fuente todas las características y funciones necesarias para el incremento de software (por ejemplo, el lanzamiento). A medida que se implementan los componentes, se diseñan y efectúan pruebas unitarias para cada uno. Además, se realizan actividades de integración (ensamble de componentes y pruebas de integración). Se emplean casos de uso para obtener un grupo de pruebas de aceptación que se ejecutan antes de comenzar la siguiente fase del PU.

La fase de transición del PU incluye las últimas etapas de la actividad general de construcción y la primera parte de la actividad de despliegue general (entrega y retroalimentación). Se da el software a los usuarios finales para las pruebas beta, quienes reportan tanto los defectos como los cambios necesarios. Además, el equipo de software genera la información de apoyo necesaria (por ejemplo, manuales de usuario, guías de solución de problemas, procedimientos de instalación, etc.) que se requiere para el lanzamiento. Al finalizar la fase de transición, el software incrementado se convierte en un producto utilizable que se lanza.

La fase de producción del PU coincide con la actividad de despliegue del proceso general. Durante esta fase, se vigila el uso que se da al software, se brinda apoyo para el ambiente de operación (infraestructura) y se reportan defectos y solicitudes de cambio para su evaluación”. (p.47).

En la siguiente figura, Pressman hace una comparación entre las fases genéricas de desarrollo de sistemas, con las del Proceso Unificado, mostrando que están en la misma línea:

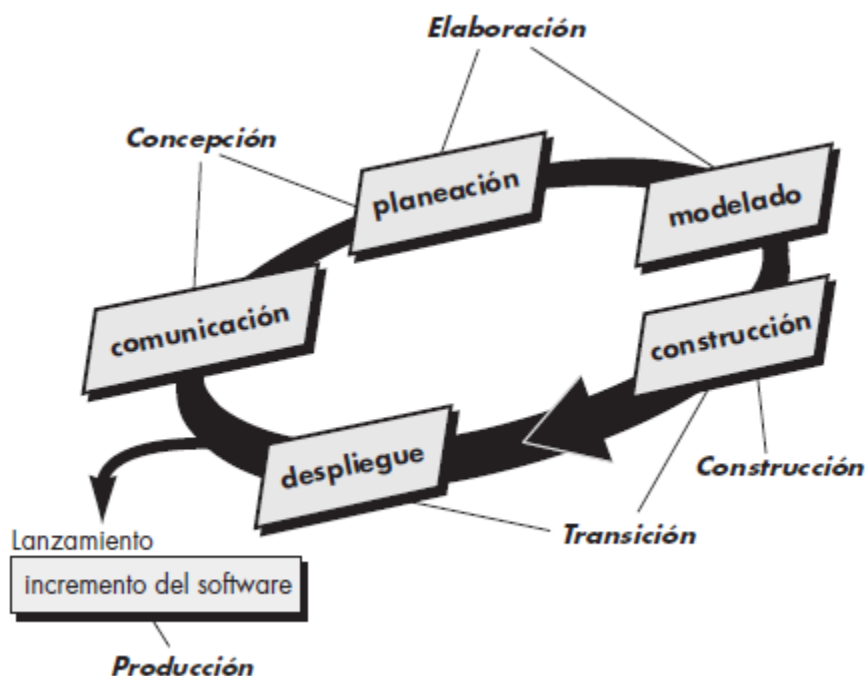


Figura 9: Proceso Unificado

Fuente: Ingeniería de Software: Un enfoque práctico (2010)

El modelo RUP también contempla una serie de buenas prácticas de ingeniería de software que se recomiendan para desarrollar sistemas. Según Sommerville (2011) son las siguientes:

- “1. Desarrollo de software de manera iterativa: Incrementar el plan del sistema con base en las prioridades del cliente, y desarrollar oportunamente las características del sistema de mayor prioridad en el proceso de desarrollo.
2. Gestión de requerimientos: Documentar de manera explícita los requerimientos del cliente y seguir la huella de los cambios a dichos requerimientos. Analizar el efecto de los cambios sobre el sistema antes de aceptarlos.
3. Usar arquitecturas basadas en componentes: Estructurar la arquitectura del sistema en componentes, como se estudió anteriormente en este capítulo.
4. Software modelado visualmente: Usar modelos UML gráficos para elaborar representaciones de software estáticas y dinámicas.
5. Verificar la calidad del software: Garantizar que el software cumpla con los estándares de calidad de la organización.
6. Controlar los cambios al software: Gestionar los cambios al software con un sistema de administración del cambio, así como con procedimientos y herramientas de administración de la configuración”. (p.52).

Al igual que los métodos anteriores, utilizar RUP no es conveniente para todos los tipos de desarrollo ni para todas las entidades, por ejemplo, no es funcional para el desarrollo de sistemas embebidos. De igual forma, dependiendo de la empresa y tipo de desarrollo, no necesariamente se ejecutarán todas las tareas del flujo de trabajo de esta metodología.

2.3.3 Metodologías Ágiles de desarrollo de sistemas

La economía mundial en los últimos años se caracteriza por sus cambios constantes, lo que ha llevado a que las empresas aprendan a responder rápidamente a esos cambios en el mercado y a buscar ventajas competitivas para mantenerse a flote.

Los sistemas de información se han convertido en aliados de las entidades para responder a esos cambios, y es por eso, que también han surgido nuevas metodologías de desarrollo que puedan responder con mayor velocidad a esas necesidades.

Algunas metodologías tradicionales generaban problemas a esta dinámica de cambio, debido al tiempo que se tardaba en especificar los requerimientos iniciales y sobre todo para gestionar y desarrollar los cambios que estos podían sufrir en el proceso de creación del sistema. Asimismo, el riguroso control por las tareas establecidas en el plan y los productos que se debían producir, muchas veces generaban trabas en los proyectos.

Debido a lo anterior, en la década de 1990, algunos desarrolladores de sistemas iniciaron con propuestas revolucionarias donde se daba más importancia al software en lugar del proceso, diseño y documentación.

En el año 2001, se creó la Alianza Ágil, en la que 17 expertos en desarrollo de software se reunieron para definir los valores y principios claves para generar software de una manera ágil, respondiendo a la dinámica de cambio que exigía el mercado. De ahí surgió el Manifiesto Ágil, que es un documento que resume la filosofía de las metodologías ágiles, de acuerdo con lo definido por los expertos en aquel momento.

Dentro del manifiesto se define lo siguiente:

“Al individuo y las interacciones del equipo de desarrollo sobre el proceso y las herramientas: La gente es el principal factor de éxito de un proyecto software. Es más importante construir un buen equipo que construir el entorno. Muchas veces se comete el error de construir primero el entorno y esperar que el equipo se adapte automáticamente. Es mejor crear el equipo y que éste configure su propio entorno de desarrollo en base a sus necesidades.

Desarrollar software que funciona más que conseguir una buena documentación: La regla a seguir es no producir documentos a menos que sean necesarios de forma inmediata para tomar una decisión importante”. Estos documentos deben ser cortos y centrarse en lo fundamental.

La colaboración con el cliente más que la negociación de un contrato: Se propone que exista una interacción constante entre el cliente y el equipo de desarrollo. Esta colaboración entre ambos será la que marque la marcha del proyecto y asegure su éxito.

Responder a los cambios más que seguir estrictamente un plan: La habilidad de responder a los cambios que puedan surgir a lo largo del proyecto (cambios en los requisitos, en la tecnología, en el equipo, etc.) determina también el éxito o fracaso de este. Por lo tanto, la planificación no debe ser estricta sino flexible y abierta”. (Canós, J, Letelier, P, & Penadés, M, 2003, p. 2).

Como se denota en estos valores del Manifiesto Ágil, algunos factores cambian su importancia con respecto a las metodologías tradicionales, por ejemplo, se le da más importancia a la construcción y funcionamiento del equipo, que al proceso y herramientas, situación similar sucede con la documentación, en la que pasa a segundo plano, y toma más importancia que se logre el desarrollo del software.

Producto de los valores supra citados, surgen también 12 principios definidos en dicho manifiesto:

1. La prioridad más alta es satisfacer al cliente por medio de la entrega pronta y continua de software valioso.
2. Son bienvenidos los requerimientos cambiantes, aun en una etapa avanzada del desarrollo. Los procesos ágiles dominan el cambio para provecho de la ventaja competitiva del cliente.
3. Entregar con frecuencia software que funcione, de dos semanas a un par de meses, de preferencia lo más pronto que se pueda.
4. Las personas de negocios y los desarrolladores deben trabajar juntos, a diario y durante todo el proyecto.
5. Hay que desarrollar los proyectos con individuos motivados. Debe darse a éstos el ambiente y el apoyo que necesiten, y confiar en que harán el trabajo.
6. El método más eficiente y eficaz para transmitir información a los integrantes de un equipo de desarrollo, y entre éstos, es la conversación cara a cara.
7. La medida principal de avance es el software que funciona.

8. Los procesos ágiles promueven el desarrollo sostenible. Los patrocinadores, desarrolladores y usuarios deben poder mantener un ritmo constante en forma indefinida.
9. La atención continua a la excelencia técnica y el buen diseño mejora la agilidad.
10. Es esencial la simplicidad: el arte de maximizar la cantidad de trabajo no realizado.
11. Las mejores arquitecturas, requerimientos y diseños surgen de los equipos con organización propia.
12. El equipo reflexiona a intervalos regulares sobre cómo ser más eficaz, para después afinar y ajustar su comportamiento en consecuencia. («12 Principles Behind the Agile Manifesto», 2015)

Estos principios diferencian la filosofía ágil de desarrollo de sistemas, de las metodologías tradicionales. Pese a lo anterior, no todas las metodologías ágiles aplican estrictamente estos 12 principios, ya que cada una de ellas, pone énfasis en diferentes aspectos, tal y como se describirá más adelante.

En la siguiente tabla, se hace un resumen de las diferencias entre las Metodologías Ágiles y las Metodologías Tradicionales:

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 provenientes 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.
El cliente es parte del equipo de desarrollo.	El cliente interactúa con el equipo de desarrollo mediante reuniones.

Grupos pequeños (<10 integrantes) y trabajando en 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.

Fuente: (Canós, J et al., 2003)

Pese a que el uso de metodologías ágiles ha venido en aumento en los últimos años, se han identificado una serie de dificultades para su aplicación en algunos proyectos. Sommerville (2011) expone algunas de esas dificultades:

- “1. Aunque es atractiva la idea del involucramiento del cliente en el proceso de desarrollo, su éxito radica en tener un cliente que desee y pueda pasar tiempo con el equipo de desarrollo, y éste represente a todos los participantes del sistema. Los representantes del cliente están comúnmente sujetos a otras presiones, así que no intervienen por completo en el desarrollo del software.
2. Quizás algunos miembros del equipo no cuenten con la personalidad adecuada para la participación intensa característica de los métodos ágiles y, en consecuencia, no podrán interactuar adecuadamente con los otros integrantes del equipo.
3. Priorizar los cambios sería extremadamente difícil, sobre todo en sistemas donde existen muchos participantes. Cada uno por lo general ofrece diversas prioridades a diferentes cambios.
4. Mantener la simplicidad requiere trabajo adicional. Bajo la presión de fechas de entrega, es posible que los miembros del equipo carezcan de tiempo para realizar las simplificaciones deseables al sistema.
5. Muchas organizaciones, especialmente las grandes compañías, pasan años cambiando su cultura, de tal modo que los procesos se definan y continúen. Para ellas, resulta difícil moverse hacia un modelo de trabajo donde los procesos sean informales y estén definidos por equipos de desarrollo”. (p.61).

De acuerdo con lo anterior, las dificultades provienen de distintas vertientes. En algunos casos desde el factor humano, como la personalidad de los miembros del equipo. Por otra parte, la cultura organizacional con sus procesos y flexibilidad, y otro factor es el cliente y su disposición para integrarse al equipo. Asimismo, también se señala la dificultad a la hora de establecer los contratos, dado que, por la dinámica cambiante de las metodologías ágiles, al formalizar un proyecto se dificulta conocer la totalidad de requerimientos formales que se abarcarán.

Actualmente existe un debate acerca de cuándo utilizar una metodología ágil y cuándo una tradicional. Como se ha expuesto hasta el momento, en ambos grupos existen pros y contras que deben ser analizados por cada ente, para determinar cuál es la que más se adapta para el cumplimiento de sus objetivos.

En las secciones siguientes, se realizará una descripción de las principales metodologías ágiles en la actualidad.

2.3.3.1 Programación Extrema (XP)

Dentro de las metodologías ágiles, la Programación Extrema es una de las más reconocidas y utilizadas. Está basada en metodologías tradicionales, sin embargo, lleva al extremo los principios y prácticas de dichas metodologías, de ahí proviene su nombre.

“XP es una metodología ágil centrada en potenciar las relaciones interpersonales como clave para el éxito en desarrollo de software, promoviendo el trabajo en equipo, preocupándose por el aprendizaje de los desarrolladores, y propiciando un buen clima de trabajo. XP se basa en realimentación continua entre el cliente y el equipo de desarrollo, comunicación fluida entre todos los participantes, simplicidad en las soluciones implementadas y coraje para enfrentar los cambios. XP se define como especialmente adecuada para proyectos con requisitos imprecisos y muy cambiantes, y donde existe un alto riesgo técnico”. (Canós, J et al., 2003, p. 3).

Como se aprecia en la cita anterior, XP se basa mucho en las relaciones interpersonales tanto entre el equipo de trabajo como en el cliente, por lo que, para garantizar el éxito en el uso de

esta metodología es importante desarrollar aparte del conocimiento técnico, las habilidades blandas de los involucrados.

Dentro de esta metodología hay cinco valores que se convierten en pilares para todo proyecto de desarrollo de sistemas. Estos valores son: comunicación, simplicidad, retroalimentación, valentía y respeto.

Pressman (2010) explica los principales valores de la siguiente manera:

“A fin de lograr la comunicación eficaz entre los ingenieros de software y otros participantes (por ejemplo, para establecer las características y funciones requeridas para el software), XP pone el énfasis en la colaboración estrecha pero informal (verbal) entre los clientes y los desarrolladores, en el establecimiento de metáforas para comunicar conceptos importantes, en la retroalimentación continua y en evitar la documentación voluminosa como medio de comunicación.

Para alcanzar la simplicidad, XP restringe a los desarrolladores para que diseñen sólo para las necesidades inmediatas, en lugar de considerar las del futuro. El objetivo es crear un diseño sencillo que se implemente con facilidad en forma de código. Si hay que mejorar el diseño, se rediseñará en un momento posterior.

La retroalimentación se obtiene de tres fuentes: el software implementado, el cliente y otros miembros del equipo de software. Al diseñar e implementar una estrategia de pruebas eficaz, el software (por medio de los resultados de las pruebas) da retroalimentación al equipo ágil. XP usa la prueba unitaria como su táctica principal de pruebas. A medida que se desarrolla cada clase, el equipo implementa una prueba unitaria para ejecutar cada operación de acuerdo con su funcionalidad especificada. Cuando se entrega un incremento a un cliente, las historias del usuario o casos de uso que se implementan con el incremento se utilizan como base para las pruebas de aceptación. El grado en el que el software implementa la salida, función y comportamiento del caso de uso es una forma de retroalimentación. Por último, conforme se obtienen nuevos requerimientos como parte de la planeación iterativa, el equipo da al cliente una retroalimentación rápida con miras al costo y al efecto en la programación de actividades”. (p.61).

La metodología XP, por medio de esos valores, comulga con los principios del desarrollo ágil establecidos en el manifiesto, y los pone en práctica a través de su proceso de desarrollo. En ese proceso entran en juego los siguientes roles: programador, cliente, encargado de pruebas (tester), encargado de seguimiento (tracker), entrenador (coach), consultor y gestor (big boss). Las funciones de cada rol se describen a continuación:

- “- Programador. El programador escribe las pruebas unitarias y produce el código del sistema.
- Cliente. Escribe las historias de usuario y las pruebas funcionales para validar su implementación. Además, asigna la prioridad a las historias de usuario y decide cuáles se implementan en cada iteración centrándose en aportar mayor valor al negocio.
- Encargado de pruebas (Tester). Ayuda al cliente a escribir las pruebas funcionales. Ejecuta las pruebas regularmente, difunde los resultados en el equipo y es responsable de las herramientas de soporte para pruebas.
- Encargado de seguimiento (Tracker). Proporciona realimentación al equipo. Verifica el grado de acierto entre las estimaciones realizadas y el tiempo real dedicado, para mejorar futuras estimaciones. Realiza el seguimiento del progreso de cada iteración.
- Entrenador (Coach). Es responsable del proceso global. Debe proveer guías al equipo de forma que se apliquen las prácticas XP y se siga el proceso correctamente.
- Consultor. Es un miembro externo del equipo con un conocimiento específico en algún tema necesario para el proyecto, en el que puedan surgir problemas.
- Gestor (Big boss). Es el vínculo entre clientes y programadores, ayuda a que el equipo trabaje efectivamente creando las condiciones adecuadas. Su labor esencial es de coordinación”. (Canós, J et al., 2003, p.5).

El proceso de programación extrema se compone de un ciclo iterativo, que comprende las etapas ilustradas en la siguiente figura:

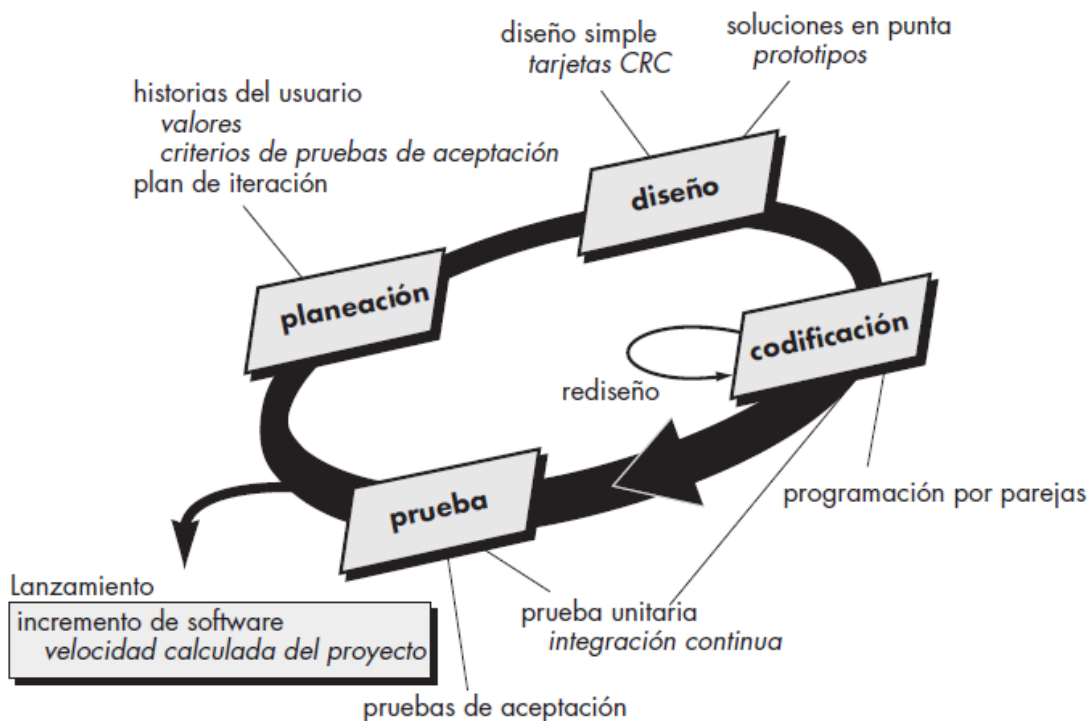


Figura 10: Proceso de programación extrema

Fuente: Ingeniería de Software: Un enfoque práctico (2010)

Según se aprecia en la figura anterior, cabe destacar que los requerimientos se realizan por medio de la técnica de historias de usuario, para lo que, en tarjetas de papel el cliente describe características que desea en el sistema. Estas historias son dinámicas y flexibles, y generalmente se desarrollan en ciclos cortos, generalmente entre una y tres semanas. Otro aspecto relevante en XP, es que se da programación por parejas, en la cual dos personas trabajan juntas creando el código para una historia con el fin de solucionar problemas en tiempo real y asegurar la calidad. Adicional a lo anterior, por la dinámica de la programación XP, las pruebas unitarias resultan fundamentales, para que, conforme se presenten incrementos, pueda probarse que todo marche conforme con lo planeado.

En la Tabla 3, se muestra una descripción de los principios y prácticas más importantes.

Tabla 3. Prácticas de programación extrema

Principio o práctica	Descripción
Planeación incremental	Los requerimientos se registran en tarjetas de historia (story cards) y las historias que se van a incluir en una liberación se determinan por el tiempo disponible y la prioridad relativa. Los desarrolladores desglosan dichas historias en “tareas” de desarrollo.
Liberaciones pequeñas	Al principio se desarrolla el conjunto mínimo de funcionalidad útil, que ofrece valor para el negocio. Las liberaciones del sistema son frecuentes y agregan incrementalmente funcionalidad a la primera liberación.
Diseño simple	Se realiza un diseño suficiente para cubrir sólo aquellos requerimientos actuales.
Desarrollo de la primera prueba	Se usa un marco de referencia de prueba de unidad automatizada al escribir las pruebas para una nueva pieza de funcionalidad, antes de que esta última se implemente.
Refactorización	Se espera que todos los desarrolladores refactoricen de manera continua el código y, tan pronto como sea posible, se encuentren mejoras de éste. Lo anterior conserva el código simple y mantenible.
Programación en pares	Los desarrolladores trabajan en pares, y cada uno comprueba el trabajo del otro; además, ofrecen apoyo para que se realice siempre un buen trabajo.
Propiedad colectiva	Los desarrolladores en pares laboran en todas las áreas del sistema, de manera que no se desarrollan islas de experiencia, ya que todos los desarrolladores se responsabilizan por todo el código. Cualquiera puede cambiar cualquier función.
Integración continua	Tan pronto como esté completa una tarea, se integra en todo el sistema. Después de tal integración, deben aprobarse todas las pruebas de unidad en el sistema.

Ritmo sustentable	Grandes cantidades de tiempo extra no se consideran aceptables, pues el efecto neto de este tiempo libre con frecuencia es reducir la calidad del código y la productividad de término medio.
Cliente en sitio	Un representante del usuario final del sistema (el cliente) tiene que disponer de tiempo completo para formar parte del equipo XP. En un proceso de programación extrema, el cliente es miembro del equipo de desarrollo y responsable de llevar los requerimientos del sistema al grupo para su implementación.

Fuente: (Sommerville, I, 2011)

De acuerdo con las descripciones de la tabla anterior, se aprecia que muchas de las prácticas y principios de la metodología XP no son nuevas en la industria del desarrollo de sistemas, sin embargo, si es relevante que estén unidas en una sola metodología por lo que su aplicación es importante para conseguir los resultados esperados.

Pese a que la programación extrema es una de las metodologías ágiles con mayor aceptación, es importante considerar que también presenta debilidades, por ejemplo, la falta de formalización en los requerimientos, falta de documentación de estructuras, dificultades para mantener una relación constructiva con los clientes, entre otras. Es por eso que para su implantación, deben considerarse estos aspectos y adaptarla de la mejor forma según las características propias de cada entidad.

2.3.3.1 Scrum

Scrum es un método de desarrollo ágil cuyo nombre proviene de una jugada del rugby, cuya característica es que los jugadores trabajan en conjunto alrededor del balón. Fue creado a principios de la década de los años 1990 por Ken Schwaber y Jeff Sutherland.

Según (Pressman, R, 2010):

“Los principios Scrum son congruentes con el manifiesto ágil y se utilizan para guiar actividades de desarrollo dentro de un proceso de análisis que incorpora las siguientes actividades estructurales: requerimientos, análisis, diseño, evolución y entrega. Dentro de

cada actividad estructural, las tareas del trabajo ocurren con un patrón del proceso) llamado sprint. El trabajo realizado dentro de un sprint (el número de éstos que requiere cada actividad estructural variará en función de la complejidad y tamaño del producto) se adapta al problema en cuestión y se define —y con frecuencia se modifica— en tiempo real por parte del equipo Scrum”. (p.69).

Como se aprecia en el párrafo anterior, los principios de Scrum están alineados a lo establecido en el manifiesto ágil, por lo que su uso para proyectos de desarrollo de sistemas mediante el uso de metodologías ágiles se ha afianzado en los últimos años.

Los creadores de Scrum lo describen de la siguiente manera:

“La esencia de Scrum es un pequeño equipo de personas. El equipo individual es altamente flexible y adaptativo. Estas fortalezas continúan operando en un equipo, en varios, en muchos y en redes de equipos que desarrollan, liberan, operan y mantienen el trabajo y los productos de trabajo de miles de personas. Ellos colaboran e interoperan a través de arquitecturas de desarrollo sofisticadas y ambientes finales de liberación”. (Schwaber, K & Sutherland, J, 2017, p. 4).

La implementación de Scrum tiene como pilares la transparencia, la inspección y adaptación. Sus creadores describen estos aspectos de la siguiente manera:

“Transparencia: Los aspectos significativos del proceso deben ser visibles para aquellos que son responsables del resultado. La transparencia requiere que dichos aspectos sean definidos por un estándar común, de tal modo que los observadores compartan un entendimiento común de lo que se están viendo.

Inspección: Los usuarios de Scrum deben inspeccionar frecuentemente los artefactos de Scrum y el progreso hacia un objetivo para detectar variaciones indeseadas. Su inspección no debe ser tan frecuente como para que interfiera en el trabajo. Las inspecciones son más beneficiosas cuando se realizan de forma diligente por inspectores expertos en el mismo lugar de trabajo.

Adaptación: Si un inspector determina que uno o más aspectos de un proceso se desvían de límites aceptables y que el producto resultante será inaceptable, el proceso o el material

que está siendo procesado deben ajustarse. Dicho ajuste debe realizarse cuanto antes para minimizar desviaciones mayores”.

Aparte de estos pilares, Scrum tiene cinco valores que son, el compromiso, coraje, foco, apertura y respeto. La aplicación de estos valores en los equipos es fundamental para alcanzar el éxito en el proyecto.

Los roles definidos en los equipos de Scrum son los siguientes:

“-Dueño del Producto (Product Owner): es el responsable de maximizar el valor del producto resultante del trabajo del Equipo de Desarrollo. El cómo se lleva a cabo esto podría variar ampliamente entre distintas organizaciones, equipos Scrum e individuos. Es la única persona responsable de gestionar la Lista del Producto (Product Backlog). Para que el Dueño de Producto pueda hacer bien su trabajo, toda la organización debe respetar sus decisiones. Las decisiones del Dueño de Producto se reflejan en el contenido y en la priorización de la Lista del Producto. Nadie puede forzar al Equipo de Desarrollo a que trabaje con base en un conjunto diferente de requisitos.

-El equipo de Desarrollo (Development Team): son los profesionales que realizan el trabajo de entregar un Incremento de producto “Terminado” que potencialmente se pueda poner en producción al final de cada Sprint. Un Incremento “Terminado” es obligatorio en la Revisión del Sprint. Solo los miembros del Equipo de Desarrollo participan en la creación del Incremento. Tienen las siguientes características: Son autoorganizados, multifuncionales, no se reconocen títulos dentro del equipo, no se reconocen sub-equipos, las responsabilidades recaen en el equipo como un todo y no en miembros individuales. El tamaño óptimo del Equipo de Desarrollo es lo suficientemente pequeño como para permanecer ágil y lo suficientemente grande como para completar una cantidad de trabajo significativa. Tener menos de tres miembros en el Equipo de Desarrollo reduce la interacción y resulta en ganancias de productividad más pequeñas. Generalmente los equipos están conformados entre tres y nueve personas.

-El Scrum Master: El Scrum Master es responsable de promover y apoyar Scrum como se define en la Guía de Scrum. Los Scrum Masters hacen esto ayudando a todos a entender la teoría, prácticas, reglas y valores de Scrum. Es un líder que está al servicio del Equipo

Scrum, ayuda a las personas externas al Equipo Scrum a entender qué interacciones con el Equipo Scrum pueden ser útiles y cuáles no. El Scrum Master ayuda a todos a modificar estas interacciones para maximizar el valor creado por el Equipo Scrum”. (Schwaber, K & Sutherland, J, 2017, p. 8).

Dentro de los roles que se acaban de describir, es necesario resaltar el papel del Scrum Master, ya que es el encargado de coordinar con distintos actores dentro del proceso. A continuación, se describen dichas relaciones:

Servicio de Scrum Master con dueño del Producto:

“-Asegurar que los objetivos, el alcance y el dominio del producto sean entendidos por todos en el equipo Scrum de la mejor manera posible;
 -Encontrar técnicas para gestionar la Lista de Producto de manera efectiva;
 -Ayudar al Equipo Scrum a entender la necesidad de contar con elementos de Lista de Producto claros y concisos;
 -Entender la planificación del producto en un entorno empírico;
 -Asegurar que el Dueño de Producto conozca cómo ordenar la Lista de Producto para maximizar el valor;
 -Entender y practicar la agilidad;
 -Facilitar los eventos de Scrum según se requiera o necesite”. (Schwaber, K & Sutherland, J, 2017, p. 8).

Según la descripción anterior, el servicio que brinda el Scrum Master al dueño del Producto es trascendental para que el Equipo de Scrum logre comprender la verdadera necesidad que tiene el dueño del producto, y pueda alcanzarse el objetivo.

Servicio del Scrum Master al Equipo de Desarrollo:

“-Guiar al Equipo de Desarrollo en ser auto organizado y multifuncional;
 -Ayudar al Equipo de Desarrollo a crear productos de alto valor;
 -Eliminar impedimentos para el progreso del Equipo de Desarrollo;
 -Facilitar los eventos de Scrum según se requiera o necesite;

-Guiar al Equipo de Desarrollo en entornos organizacionales”. (Schwaber, K & Sutherland, J, 2017, p. 8).

Pese a que en Scrum los equipos son auto dirigidos el Scrum Master debe ser una motivación para que los equipos alcancen sus objetivos, así como ser una figura que ayude a solventar cualquier inconveniente que se presente, teniendo siempre presente los valores y principios de este marco de trabajo.

Servicio del Scrum Master a la Organización:

“-Liderar y guiar a la organización en la adopción de Scrum;
 -Planificar las implementaciones de Scrum en la organización;
 -Ayudar a los empleados e interesados a entender y llevar a cabo Scrum y el desarrollo empírico de producto;
 -Motivar cambios que incrementen la productividad del Equipo Scrum;
 -Trabajar con otros Scrum Masters para incrementar la efectividad de la aplicación de Scrum en la organización”. (Schwaber, K & Sutherland, J, 2017, p. 9).

En un inicio, el cambio de la organización va a requerir del apoyo de los Scrum Master que estén involucrados en el proyecto, los cuales, conociendo con profundidad la metodología Scrum, puedan guiar a la organización a alcanzar los resultados esperados y a adaptarse de la mejor forma a esa nueva forma de trabajo.

Como todas las metodologías, Scrum tiene una serie de etapas, denominadas en este caso eventos, los cuales son: los Sprint, la Planificación de Sprint, el Objetivo del Sprint, el Scrum Diario, la Revisión de Sprint y la Retrospectiva de Sprint. Estos se describen con mayor detalle a continuación:

El Sprint es el punto medular de Scrum. Se define de la siguiente manera:

“Es un bloque de tiempo (time-box) de un mes o menos durante el cual se crea un incremento de producto “Terminado” utilizable y potencialmente desplegable. Es más

conveniente si la duración de los Sprints es consistente a lo largo del esfuerzo de desarrollo. Cada nuevo Sprint comienza inmediatamente después de la finalización del Sprint anterior. Los Sprints contienen y consisten en la Planificación del Sprint (Sprint Planning), los Scrums Diarios (Daily Scrums), el trabajo de desarrollo, la Revisión del Sprint (Sprint Review), y la Retrospectiva del Sprint (Sprint Retrospective).

Durante el Sprint:

- No se realizan cambios que puedan afectar al Objetivo del Sprint (Sprint Goal);
- Los objetivos de calidad no disminuyen;
- El alcance puede clarificarse y renegociarse entre el Dueño de Producto y el Equipo de Desarrollo a medida que se va aprendiendo más”. (Schwaber, K & Sutherland, J, 2017, p. 10).

La Planeación de Sprint se describe de la siguiente manera:

“El trabajo a (sic) realizar durante el Sprint se planifica en la Planificación de Sprint. Este plan se crea mediante el trabajo colaborativo del Equipo Scrum completo.

La Planificación de Sprint tiene un máximo de duración de ocho horas para un Sprint de un mes. Para Sprints más cortos el evento es usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña al Equipo Scrum a mantenerse dentro del bloque de tiempo.

La Planificación de Sprint responde a las siguientes preguntas:

- ¿Qué puede entregarse en el Incremento resultante del Sprint que comienza?
- ¿Cómo se conseguirá hacer el trabajo necesario para entregar el Incremento?”. (Schwaber, K & Sutherland, J, 2017, p. 10).

Con respecto al objetivo del Sprint, cabe destacar lo siguiente:

“El Objetivo del Sprint es una meta establecida para el Sprint (sic) que puede lograrse mediante la implementación de la Lista de Producto. Proporciona una guía al Equipo de Desarrollo acerca de por qué está construyendo el incremento. Se crea durante la Planificación del Sprint. El objetivo del Sprint brinda (sic) al equipo de desarrollo cierta flexibilidad con respecto a la funcionalidad implementada en el Sprint. Los elementos de

la Lista del Producto seleccionados ofrecen una función coherente que puede ser el objetivo del Sprint. El objetivo del Sprint puede representar otro nexo de unión que haga que el Equipo de Desarrollo trabaje en conjunto y no en iniciativas separadas”. (Schwaber, K & Sutherland, J, 2017, p. 11).

Dentro de los Sprints, diariamente debe hacerse una reunión. Estos son los Scrum Diarios que consisten en lo siguiente:

“El Scrum Diario es una reunión con un bloque de tiempo de 15 minutos para el Equipo de Desarrollo. El Scrum Diario se lleva a cabo cada día del sprint. En él, el Equipo de Desarrollo planea el trabajo para las siguientes 24 horas. Esto optimiza la colaboración y el desempeño del equipo inspeccionando el trabajo avanzado desde el último Scrum Diario y haciendo una proyección del trabajo del Sprint a realizar a continuación. El Scrum Diario se realiza a la misma hora y en el mismo lugar todos los días para reducir la complejidad. El Equipo de Desarrollo usa el Scrum Diario para evaluar el progreso hacia el Objetivo del Sprint y para evaluar qué tendencia sigue este progreso hacia la finalización del trabajo contenido en la Lista de Pendientes del Sprint. El Scrum Diario optimiza las posibilidades de que el Equipo de Desarrollo cumpla el Objetivo del Sprint. Cada día, el Equipo de Desarrollo debería entender cómo intenta trabajar en conjunto como un equipo autoorganizado para lograr el Objetivo del Sprint y crear el Incremento esperado hacia el final del Sprint”. (Schwaber, K & Sutherland, J, 2017, p. 12).

Al finalizar cada Sprint, debe realizarse una revisión en la que se abarquen una serie de aspectos. Estas son las Revisiones de Sprint que se describen de la siguiente manera:

“La Revisión de Sprint se ejecuta para inspeccionar el Incremento y adaptar la Lista de Producto si fuese necesario. Durante la Revisión de Sprint, el Equipo Scrum y los interesados colaboran acerca de lo que se hizo durante el Sprint. Basándose en esto y en cualquier cambio a la Lista de Producto durante el Sprint, los asistentes colaboran para determinar las siguientes cosas que podrían hacerse para optimizar el valor. Se trata de una reunión informal, no una reunión de seguimiento, y la presentación del Incremento tiene como objetivo facilitar la retroalimentación de información y fomentar la colaboración.

Se trata de una reunión de, a lo sumo, cuatro horas para Sprints de un mes. Para Sprints más cortos, el evento usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado.

La Revisión de Sprint incluye los siguientes elementos:

- Los asistentes son el Equipo Scrum y los interesados clave invitados por el Dueño de Producto;
- El Dueño de Producto explica qué elementos de la Lista de Producto se han “Terminado” y cuales no se han “Terminado”;
- El Equipo de Desarrollo habla acerca de qué estuvo bien durante el Sprint, qué problemas aparecieron y cómo fueron resueltos esos problemas;
- El Equipo de Desarrollo hace una demostración del trabajo que ha “Terminado” y responde preguntas acerca del Incremento;
- El Dueño de Producto habla acerca de la Lista de Producto en su estado actual. Proyecta objetivos probables y fechas de entrega en el tiempo basándose en el progreso obtenido hasta la fecha (si fuera necesario);
- El grupo completo colabora acerca de qué hacer a continuación, de modo que la Revisión del Sprint proporcione información de entrada valiosa para Reuniones de Planificación de Sprints subsiguientes.
- Revisión de cómo el mercado o el uso potencial del producto podría haber cambiado lo que es de más valor para hacer a continuación; y,
- Revisión de la línea de tiempo, presupuesto, capacidades potenciales y mercado para las próximas entregas de funcionalidad o capacidad prevista del producto.

El resultado de la Revisión de Sprint es una Lista de Producto revisada que define los elementos de la Lista de Producto posibles para el siguiente Sprint. Es posible además que la Lista de Producto reciba un ajuste general para enfocarse en nuevas oportunidades”. (Schwaber, K & Sutherland, J, 2017, p. 14).

El último de los elementos de Scrum consiste en la Retrospectiva de Sprint, que consiste en una oportunidad para identificar e implementar mejoras para el siguiente Sprint. Los creadores de Scrum lo definen de la siguiente manera:

La Retrospectiva de Sprint es una oportunidad para el Equipo Scrum de inspeccionarse a sí mismo y de crear un plan de mejoras que sean abordadas durante el siguiente Sprint. Tiene lugar después de la Revisión de Sprint y antes de la siguiente Planificación de Sprint. Se trata de una reunión de, a lo sumo, tres horas para Sprints de un mes. Para Sprints más cortos el evento es usualmente más corto. El Scrum Master se asegura de que el evento se lleve a cabo y que los asistentes entiendan su propósito.

El Scrum Master se asegura de que la reunión sea positiva y productiva. El Scrum Master enseña a todos a mantener el evento dentro del bloque de tiempo fijado. El Scrum Master participa en la reunión como un miembro del equipo ya que la responsabilidad del proceso Scrum recae sobre él.

El propósito de la Retrospectiva de Sprint es:

- Inspeccionar cómo fue el último Sprint en cuanto a personas, relaciones, procesos y herramientas;
- Identificar y ordenar los elementos más importantes que salieron bien y las posibles mejoras;
- Crear un plan para implementar las mejoras a la forma en la que el Equipo Scrum desempeña su trabajo.

El Scrum Master alienta al equipo para que mejore, dentro del marco de proceso Scrum, su proceso de desarrollo y sus prácticas para hacerlos más efectivos y amenos para el siguiente Sprint. (Schwaber, K & Sutherland, J, 2017, p. 14)

En la figura que se muestra a continuación, se ve en síntesis el diagrama del proceso Scrum que se acaba de describir:

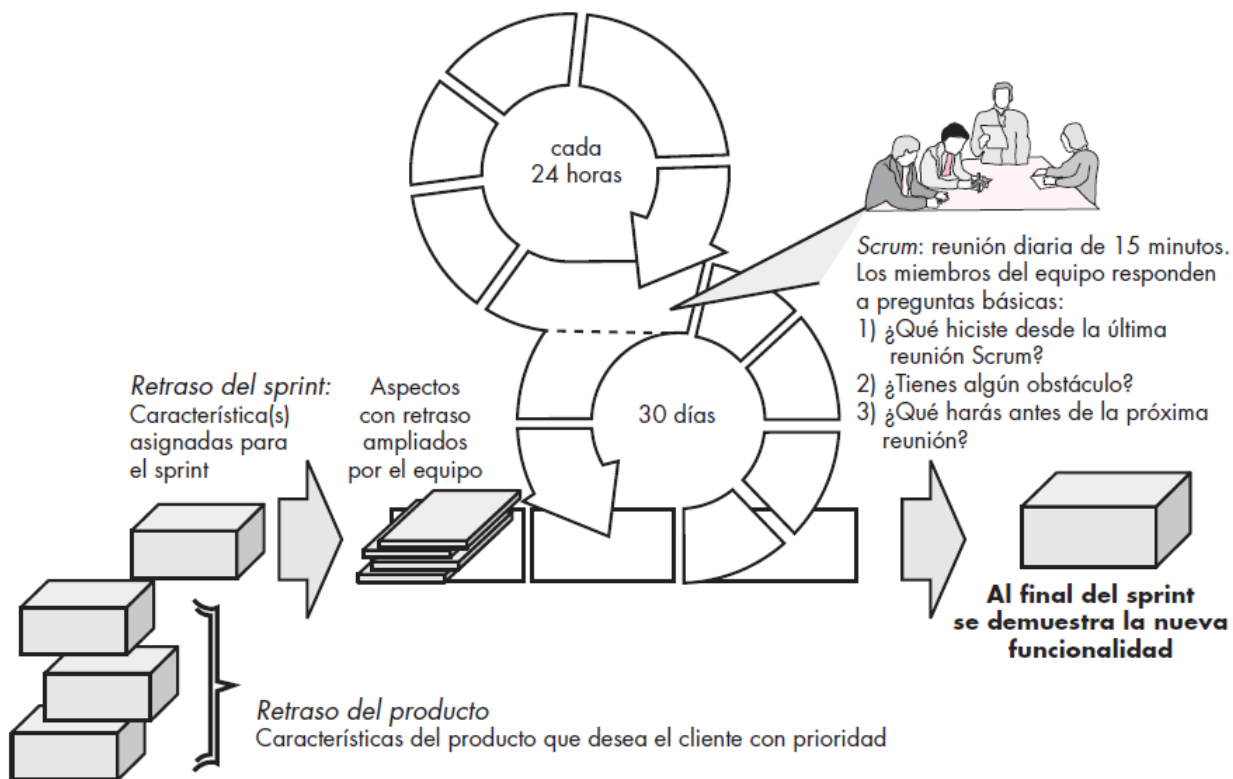


Figura 11: Flujo del proceso Scrum

Fuente: Ingeniería de Software: Un enfoque práctico (2010)

En los párrafos anteriores, se ha descrito qué es Scrum, sus etapas, y su funcionamiento. Además, es relevante conocer los dos artefactos de Scrum. El primer artefacto es la Lista de Producto (Product Backlog) que se define de la siguiente manera:

“La Lista de Producto es una lista ordenada de todo lo que se conoce que es necesario en el producto. Es la única fuente de requisitos para cualquier cambio a realizarse en el producto. El Dueño de Producto (Product Owner) es el responsable de la Lista de Producto, incluyendo su contenido, disponibilidad y ordenación.

Una Lista de Producto nunca está completa. El desarrollo más temprano de la misma solo refleja los requisitos conocidos y mejor entendidos al principio. La Lista de Producto evoluciona a medida que el producto y el entorno en el que se usará también lo hacen. La Lista de Producto es dinámica; cambia constantemente para identificar lo que el producto necesita para ser adecuado, competitivo y útil. Si un producto existe, su Lista de Producto también existe.

La Lista de Producto enumera todas las características, funcionalidades, requisitos, mejoras y correcciones que constituyen cambios a realizarse sobre el producto para entregas futuras. Los elementos de la Lista de Producto tienen como atributos la descripción, el orden, la estimación y el valor. Los elementos de La Lista de Producto muchas veces incluyen descripciones de las pruebas que demostrarán la completitud de tales elementos cuando estén “Terminados”.

A medida que un producto es utilizado y se incrementa su valor y el mercado proporciona retroalimentación, la Lista de Producto se convierte en una lista más larga y exhaustiva. Los requisitos nunca dejan de cambiar así que la Lista de Producto es un artefacto vivo”. (Schwaber, K & Sutherland, J, 2017, p. 15).

Tal y como se indica en la cita anterior, en la lista de producto están definidos los requerimientos que tiene el producto, con tal de satisfacer las necesidades del cliente. Es importante destacar que esa lista es viva, es decir, puede sufrir cambios, lo que resalta una de las características de las metodologías ágiles.

Otro de los artefactos es la Lista de Pendientes del Sprint (Sprint Backlog), la cual es similar a la Lista del Producto, sólo que seleccionando algunos elementos que se desarrollarán en un determinado Sprint. A continuación, se detalla este artefacto:

“La Lista de Pendientes del Sprint es el conjunto de elementos de la Lista de Producto seleccionados para el Sprint, más un plan para entregar el Incremento de producto y conseguir el Objetivo del Sprint. La Lista de Pendientes del Sprint es una predicción hecha por el Equipo de Desarrollo acerca de qué funcionalidad formará parte del próximo Incremento y del trabajo necesario para entregar esa funcionalidad en un Incremento “Terminado”.

La Lista de Pendientes del Sprint hace visible todo el trabajo que el Equipo de Desarrollo identifica como necesario para alcanzar el Objetivo del Sprint. Para asegurar el mejoramiento continuo, la Lista de Pendientes del Sprint incluye al menos una mejora de procesos de alta prioridad identificada en la Retrospectiva inmediatamente anterior.

La Lista de Pendientes del Sprint es un plan con un nivel de detalle suficiente como para que los cambios en el progreso se puedan entender en el Scrum Diario. El Equipo de

Desarrollo modifica la Lista de Pendientes del Sprint durante el Sprint y esta Lista de Pendientes del Sprint emerge a lo largo del Sprint. Esto ocurre a medida que el Equipo de Desarrollo trabaja en lo planeado y aprende más acerca del trabajo necesario para conseguir el Objetivo del Sprint”. (Schwaber, K & Sutherland, J, 2017, p. 17).

El último de los artefactos de Scrum es el Incremento. Este se define de la siguiente manera:

El Incremento es la suma de todos los elementos de la Lista de Producto completados durante un Sprint y el valor de los incrementos de todos los Sprints anteriores. Al final de un Sprint el nuevo Incremento debe estar “Terminado”, lo cual significa que está en condiciones de ser utilizado y que cumple la Definición de “Terminado” del Equipo Scrum. Un incremento es un cuerpo de trabajo inspeccionable y terminado que respalda el empirismo al final del Sprint. El incremento es un paso hacia una visión o meta. El incremento debe estar en condiciones de utilizarse sin importar si el Dueño de Producto decide liberarlo o no. (Schwaber, K & Sutherland, J, 2017, p. 17)

Se ha identificado una serie de ventajas de utilizar Scrum para el desarrollo de software. Sommerville, I, (2011) hace referencia a esos beneficios de la siguiente forma:

1. El producto se desglosa en un conjunto de piezas manejables y comprensibles.
2. Los requerimientos inestables no retrasan el progreso.
3. Todo el equipo tiene conocimiento de todo y, en consecuencia, se mejora la comunicación entre el equipo.
4. Los clientes observan la entrega a tiempo de los incrementos y obtienen retroalimentación sobre cómo funciona el producto.
5. Se establece la confianza entre clientes y desarrolladores, a la vez que se crea una cultura positiva donde todos esperan el triunfo del proyecto. (p.74)

Scrum puede mezclarse con otras metodologías, por ejemplo, al no definir elementos como análisis y documentación de requerimientos, diseño de pruebas, desarrollo, entre otras, podrían utilizarse alternativas como las sugeridas en otras metodologías ágiles como XP. Pese a lo anterior,

siempre deben respetarse todos sus eventos, roles, reglas y artefactos, ya que, según sus creadores, aunque se aplique la mayoría, no sería Scrum.

2.4 Gestión de proyectos

La gestión de proyectos ha cobrado relevancia en las cúpulas de las organizaciones como una herramienta para mejorar la obtención de resultados en las organizaciones. Pese a que desde hace siglos se ha visto cómo se desarrollan grandes proyectos, es hasta a mediados del siglo XX cuando se empieza a formalizar la dirección de proyectos como una profesión.

Dentro de los proyectos clave en las principales organizaciones, se encuentran los sistemas de información, por lo que es necesario conocer las generalidades de los proyectos, así como el detalle de las principales metodologías que se utilizan para gestionarlos, por lo que esta es la temática que se abarca en esta sección.

2.4.1 Generalidades de los proyectos

La Real Academia Española (2014) define un proyecto como “Conjunto de escritos, cálculos y dibujos que se hacen para dar idea de cómo ha de ser y lo que ha de costar una obra de arquitectura o de ingeniería”. Por su parte, el Project Management Institute, (2017), propone la siguiente: “Un proyecto es un esfuerzo temporal que se lleva a cabo para crear un producto, servicio o resultado único”. Pese a que es una definición sencilla encierra un significado más amplio que se desarrolla en los siguientes párrafos.

Castro, C, (2013) explica el concepto de proyecto de la siguiente forma:

En una definición general podemos decir que es la búsqueda de una solución inteligente a un problema tendiente a resolver, fundamentalmente, necesidades humanas.

Más específicamente, es la combinación de todos los recursos necesarios, reunidos en una organización temporal, para la transformación de una idea en una realidad tangible y operante.

¿Y qué es una idea? Es la representación mental que se genera a partir del razonamiento, de la imaginación o inspiración de una o más personas. Las necesitamos para resolver problemas o generar nuevas oportunidades y que, de probarse su viabilidad, podrían disparar proyectos de cambio.

Ahora bien, cuando hablamos de recursos necesarios ¿A qué nos referimos? Nos referimos al personal, a los recursos tecnológicos, financieros y otros insumos, que con su correcta utilización pueden producir un bien o servicio útil al ser humano y a la sociedad en general.

Con base en las definiciones anteriores, está claro que el proyecto busca satisfacer una necesidad previamente definida a través de la creación de un producto o servicio, por medio del uso de recursos en un periodo determinado de tiempo.

Tal y como se abarcó previamente, en la Guía del PMBOK, y con base en la definición que se utiliza para proyectos, estos deben cumplir cuatro características clave, las cuales se especifican a continuación.

En primer lugar, se abarca la característica de que un proyecto genera un producto, servicio o resultado único:

“Los proyectos se llevan a cabo para cumplir objetivos mediante la producción de entregables. El cumplimiento de los objetivos del proyecto puede producir uno o más de los siguientes entregables:

-Un producto único, que puede ser un componente de otro elemento, una mejora o corrección de un elemento o un nuevo elemento final en sí mismo (p.ej., la corrección de un defecto en un elemento final);

-Un servicio único o la capacidad de realizar un servicio (p.ej., una función de negocio que brinda apoyo a la producción o distribución);

-Un resultado único, tal como una conclusión o un documento (p.ej., un proyecto de investigación que desarrolla conocimientos que se pueden emplear para determinar si existe una tendencia o si un nuevo proceso beneficiará a la sociedad);

-Una combinación única de uno o más productos, servicios o resultados (p.ej., una aplicación de software, su documentación asociada y servicios de asistencia al usuario)”.
(Project Management Institute, 2017, p. 4).

La segunda característica de un proyecto es que requiere un esfuerzo temporal:

“La naturaleza temporal de los proyectos implica que un proyecto tiene un principio y un final definidos. Que sea temporal no significa necesariamente que un proyecto sea de corta duración. El final del proyecto se alcanza cuando se cumplen una o más de las siguientes situaciones:

- Los objetivos del proyecto se han logrado;
- Los objetivos no se cumplirán o no pueden cumplirse;
- El financiamiento del proyecto se ha agotado o ya no está disponible;
- La necesidad del proyecto ya no existe (p.ej., el cliente ya no desea terminar el proyecto, un cambio de estrategia o prioridad pone fin al proyecto, la dirección de la organización decide finalizar el proyecto);
- Los recursos humanos o físicos ya no están disponibles; o
- El proyecto se da por terminado por conveniencia o causa legal”. (Project Management Institute, 2017, p. 5).

En tercer lugar, se tiene que los proyectos son impulsores del cambio, la cual se explica de la siguiente manera:

“Los proyectos impulsan el cambio en las organizaciones. Desde una perspectiva de negocio, un proyecto está destinado a mover una organización de un estado a otro estado a fin de lograr un objetivo específico. Antes de que comience el proyecto, normalmente se dice que la organización está en el estado actual. El resultado deseado del cambio impulsado por el proyecto se describe como el estado futuro.

Para algunos proyectos esto puede implicar la creación de un estado de transición, donde se llevan a cabo múltiples pasos a lo largo de un continuo para alcanzar el estado futuro. La conclusión exitosa de un proyecto conduce a que la organización pase al estado futuro y alcance el objetivo específico”. (Project Management Institute, 2017, p. 6).

Por último, la cuarta característica define que los proyectos hacen posible la creación de valor de negocio, y se explica de la siguiente manera:

“El PMI define el valor del negocio como el beneficio cuantificable neto que se deriva de una iniciativa de negocio. El beneficio puede ser tangible, intangible o ambos. En análisis de negocios, el valor del negocio es considerado el retorno en forma de elementos como tiempo, dinero, bienes o intangibles, a cambio de algo intercambiado

El valor del negocio en los proyectos se refiere al beneficio que los resultados de un proyecto específico proporcionan a sus interesados. El beneficio de los proyectos puede ser tangible, intangible o ambos”. (Project Management Institute, 2017, p. 7).

Como complemento a lo anterior, Castro (2013) indica que cualquier proyecto, independientemente del tipo, debe cumplir con estas características:

“-Orientado a objetivos claros y accesibles

-Temporal: debe tener inicio y fin definidos con anterioridad

-Elaboración progresiva: por medio de actividades interrelacionadas, coordinadas entre sí y que, en general, no son repetitivas

-Correcta asignación de recursos, ya que son limitados

-Costos y tiempos predefinidos

-Diferentes entre sí. Si bien pueden existir similitudes entre distintos proyectos, siempre obtienen resultados únicos”. (p.18).

Por otra parte, con respecto al ciclo de vida de un proyecto, pueden presentarse etapas que, dependiendo de la metodología, tienen nombres distintos. Dentro de las más conocidas está la propuesta en la Guía de PMBOK, la cual propone la siguiente estructura:

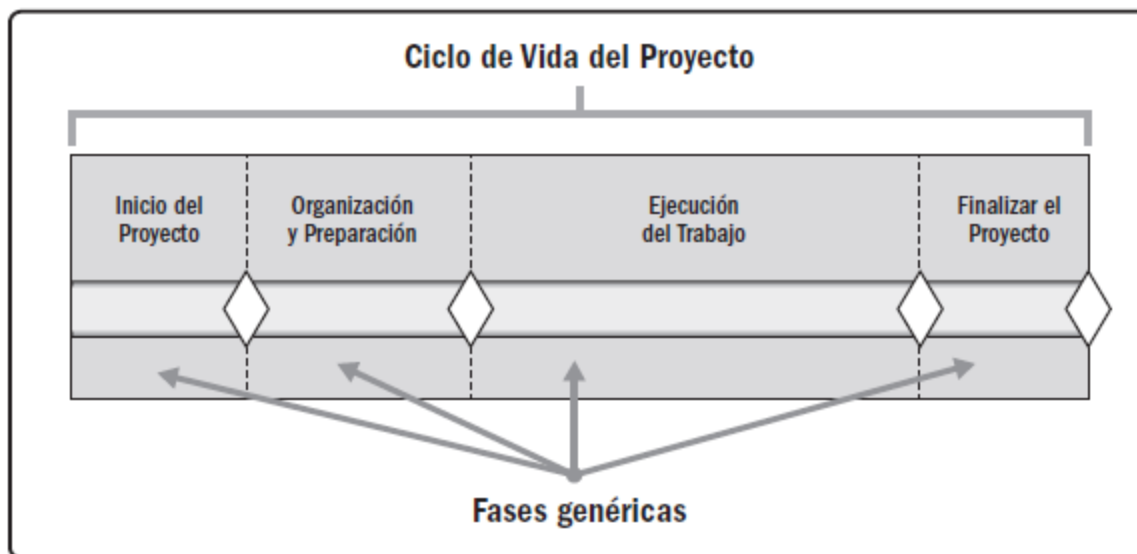


Figura 12: Representación Genérica del Ciclo de Vida de un Proyecto

Fuente: Guía del PMBOK (2018)

En la figura anterior se puede apreciar que en PMBOK se establecen cuatro fases genéricas: Inicio del proyecto, Organización y preparación, Ejecución del trabajo y Finalizar el proyecto. Asimismo, divide los procesos de proyectos en cinco grupos que son: de Inicio, de Planificación, de Ejecución, de Monitoreo y Control y de Cierre.

Por su parte, Castro, C (2013) en su metodología de administración de proyectos de tecnologías de información propone las siguientes fases:

“-Fase 1: Estudio, formulación y planificación: fase que estudia, formula, planifica, presupuesta y evalúa, la o las ideas y, de demostrar ser viable la solución planteada con sus resultados esperados

-Fase 2: Supervisión y Control: fase que estará presente durante todo el desarrollo del proyecto. El control, sobre todo en los puntos clave o hitos del proyecto, deberá estar presente, generando informes de diferentes tipos hasta el momento de la aprobación y cierre final.

-Fase 3: Ejecución: Se pone en marcha cuando, aprobado el proyecto como consecuencia de haberse demostrado su factibilidad y conveniencia de implantación, se decida invertir, adquirir, construir, asignar personal, capacitar y aplicar todos los recursos que requiera la

implementación del proyecto de acuerdo con la planificación y los presupuestos realizados en la fase 1.

-Fase 4: Puesta en marcha y operación en régimen: Estamos en presencia de la puesta en marcha o muy cerca de ella cuando se esté en condiciones de comenzar a generar los productos, servicios y funcionalidades que el proyecto prometió. Este es un momento crítico que requiere una dedicación extraordinaria y donde aparecerán todas las resistencias o críticas que hasta el momento no han sido explicitadas.

-Fase 5: Evaluación: la fase que se activa cuando la solución se encuentra funcionando en régimen dentro de la estructura organizacional con sus operadores funcionales, sin necesidad de asistencia externa del equipo de implantación del proyecto. Allí se evaluará si la solución cumplió con las expectativas, en los tiempos previstos, con los resultados esperados, con la planificación y los costos presupuestados y con un aceptable manejo de las variables sociales incluyendo a los conflictos, que sin duda se hayan dado durante su ejecución y puesta en régimen de la solución.

-Fase 6: Finalización y Documentación: Última fase que, por ser así, muchas veces es denominada la fase olvidada. Aquí deben realizarse todas las formalizaciones que se requieran para transferir la responsabilidad de la gestión de la solución a las áreas internas: se liberan responsabilidades y se asignan nuevas para que la solución quede implantada definitivamente en la organización”. (p.26).

Cabe destacar que estas seis fases no son secuenciales, si no que en algunos momentos pueden ser transversales, tal y como sucede también en PMBOK con sus grupos de procesos.

Teniendo en claro el concepto de proyecto, sus características y fases, es importante resaltar que la institución que adopte y aplique una metodología de administración de proyectos, podría obtener una serie de beneficios, como una mejor organización y aplicación de recursos, poseer información clara de en qué se está trabajando y el estado de cada uno de esos trabajos, reducir riesgos con respecto a cuándo se trabaja con lo imprevisto y realizar una mejor gestión de los cambios que puedan surgir como lo demanda el entorno.

2.4.2 Metodologías de gestión de proyectos

Pese a que en secciones anteriores se abarcó el concepto de metodología, para los conceptos que se describen a continuación, es importante repasar brevemente el concepto:

“Una metodología es un sistema de prácticas, técnicas, procedimientos y normas utilizado por quien trabajan en una disciplina”. (Rivera, F, 2015, p. 7).

Ahora bien, dado que en esta sección se abarcan metodologías de gestión de proyectos, es importante clarificar qué es la gestión de proyectos y su importancia para las organizaciones. La Guía de PMBOK define estos aspectos de la siguiente manera:

“La dirección de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para cumplir con los requisitos del mismo. Se logra mediante la aplicación e integración adecuadas de los procesos de dirección de proyectos identificados para el proyecto. La dirección de proyectos permite a las organizaciones ejecutar proyectos de manera eficaz y eficiente”. (Project Management Institute, 2017, p. 10).

Dentro de los beneficios de una buena administración de proyectos, destacan que se logran cumplir los objetivos de negocio satisfaciendo las necesidades de los interesados, poder medir plazos y lograr oportunidad en la entrega de los productos, optimizar el uso de los recursos, gestionar restricciones y el cambio de mejor manera.

En la actualidad existe un amplio número de estándares para la gestión de proyectos, como lo son IPMA Competence Baseline (ICB) creado por IPMA (International Project Management Association), la ISO 21500:2012 Guidance on project management desarrollada por el comité ISO, Projects in Controlled Environments (PRINCE2) creado por el gobierno del Reino Unido y Project Management Body of Knowledge (PMBOK) creado por el Project Management Institute (PMI).

Para la presente investigación se profundiza en PMBOK y en PRINCE2, que son las metodologías de administración de proyectos más conocidas en la región.

2.4.2.1 Project Management Body of Knowledge (PMBOK)

Tal y como se mencionó anteriormente, el estándar PMBOK fue creado por el Project Management Institute (PMI). En él se proporcionan detalles sobre conceptos clave, tendencias emergentes, consideraciones para adaptar los procesos de la dirección de proyectos e información sobre cómo aplicar herramientas y técnicas a los proyectos.

“Es desarrollado por el Project Management Institute (PMI). PMI es una asociación de miembros sin fines de lucro para la profesión de dirección de proyectos. Es fundada en 1969 para identificar las prácticas de gestión común en los proyectos de todas las industrias. Posteriormente en 1981 se crea el grupo Ethics, Standards and Accreditation (ESA) para desarrollar los procedimientos y conceptos de la dirección de proyectos. En 1983 se publican los resultados de este grupo y en 1984 se realizan las primeras certificaciones de Project Management Professional (PMP). Entre 1986 y 1987 se realiza la elaboración y publicación de los Fundamentos para la dirección de proyectos. A partir de 1991 se publica la primera edición del PMBOK”. (Fernández, K., Garrido, A, Ramírez, Y., & Perdomo, I., 2015, p. 4).

Actualmente, y desde el año 2017, está vigente la sexta edición que “incluye diez áreas de conocimiento, 49 procesos e información de cómo aplicar la metodología en ambientes ágiles, tendencias y prácticas emergentes, énfasis en el conocimiento estratégico del negocio y una sección especial para el rol del gerente de proyecto”. (Vélez, S., Zapata, A., & Henao, A., 2018, p. 70).

PMBOK describe procesos de dirección de proyectos y la relación que tienen entre sí. De acuerdo con esta guía, un proceso se entiende como: “Serie sistemática de actividades dirigidas a producir un resultado final de forma tal que se actuará sobre una o más entradas para crear una o más salidas”. (Project Management Institute, 2017, p. 18). Además, como se mencionó anteriormente, PMBOK divide los procesos en cinco grupos, los cuales son independientes a las fases del proyecto. Estos grupos se describen con mayor detalle a continuación:

“- Grupo de procesos de Inicio: Procesos concebidos para definir un nuevo proyecto o una nueva fase de un proyecto existente al obtenerse la autorización para iniciar el proyecto o fase.

- Grupo de procesos de Planificación: Procesos requeridos para establecer el alcance del proyecto, refinar los objetivos y definir el curso de acción requerido para alcanzar los objetivos propuestos del proyecto.
- Grupo de procesos de Ejecución: Procesos realizados para completar el trabajo definido en el plan para la dirección del proyecto a fin de satisfacer las especificaciones del mismo.
- Grupo de procesos de Monitoreo y control: Procesos requeridos para rastrear, revisar y regular el progreso y el desempeño del proyecto, para identificar áreas en las que el plan requiera cambio y para iniciar los cambios correspondientes.
- Grupo de procesos de Cierre: Procesos realizados para finalizar todas las actividades a través de todos los grupos de procesos, a fin de cerrar formalmente el proyecto o una fase del mismo”. (Fernández, K. et al., 2015, p. 5)

A su vez, los procesos descritos en la guía, se agrupan en áreas de conocimiento. Estas son definidas de la siguiente manera: “Área identificada de la dirección de proyectos definida por sus requisitos de conocimientos y que se describe en términos de sus procesos, prácticas, datos iniciales, resultados, herramientas y técnicas que los componen”. (Project Management Institute, 2017, p. 18).

Las áreas de conocimiento son diez, gestión de la integración, gestión del alcance, del tiempo, de los costos, de la calidad, de los recursos humanos, de las comunicaciones, de los riesgos, de las adquisiciones y gestión de los interesados y se describen con mayor detalle a continuación:

- “-Gestión de la Integración del Proyecto. Incluye los procesos y actividades para identificar, definir, combinar, unificar y coordinar los diversos procesos y actividades de dirección del proyecto dentro de los Grupos de Procesos de la Dirección de Proyectos.
- Gestión del Alcance del Proyecto. Incluye los procesos requeridos para garantizar que el proyecto incluye todo el trabajo requerido y únicamente el trabajo requerido para completarlo con éxito.
- Gestión del Cronograma del Proyecto. Incluye los procesos requeridos para administrar la finalización del proyecto a tiempo.

- Gestión de los Costos del Proyecto. Incluye los procesos involucrados en planificar, estimar, presupuestar, financiar, obtener financiamiento, gestionar y controlar los costos de modo que se complete el proyecto dentro del presupuesto aprobado.
- Gestión de la Calidad del Proyecto. Incluye los procesos para incorporar la política de calidad de la organización en cuanto a la planificación, gestión y control de los requisitos de calidad del proyecto y el producto, a fin de satisfacer las expectativas de los interesados.
- Gestión de los Recursos del Proyecto. Incluye los procesos para identificar, adquirir y gestionar los recursos necesarios para la conclusión exitosa del proyecto.
- Gestión de las Comunicaciones del Proyecto. Incluye los procesos requeridos para garantizar que la planificación, recopilación, creación, distribución, almacenamiento, recuperación, gestión, control, monitoreo y disposición final de la información del proyecto sean oportunos y adecuados.
- Gestión de los Riesgos del Proyecto. Incluye los procesos para llevar a cabo la planificación de la gestión, identificación, análisis, planificación de respuesta, implementación de respuesta y monitoreo de los riesgos de un proyecto.
- Gestión de las Adquisiciones del Proyecto. Incluye los procesos necesarios para la compra o adquisición de los productos, servicios o resultados requeridos por fuera del equipo del proyecto.
- Gestión de los Interesados del Proyecto. Incluye los procesos requeridos para identificar a las personas, grupos u organizaciones que pueden afectar o ser afectados por el proyecto, para analizar las expectativas de los interesados y su impacto en el proyecto, y para desarrollar estrategias de gestión adecuadas a fin de lograr la participación eficaz de los interesados en las decisiones y en la ejecución del proyecto”. (Project Management Institute, 2017, p. 24).

En la figura que se muestra a continuación, se muestra la interrelación entre las áreas de conocimiento, grupos de procesos y ciclo de vida del proyecto:

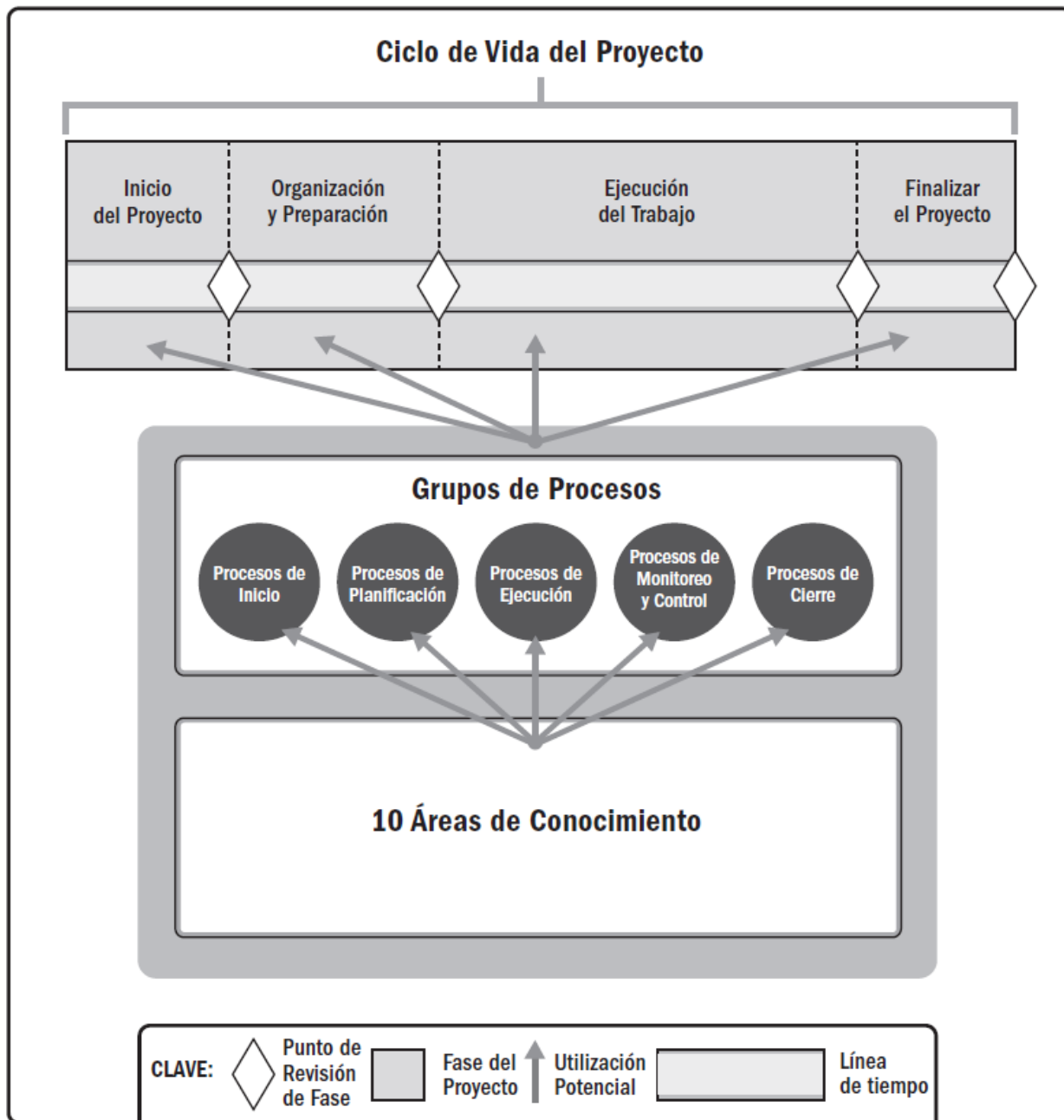


Figura 13: Interrelación entre los Componentes Clave de los Proyectos de la Guía del PMBOK

Fuente: Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK) (2017)

Se ha identificado una serie de beneficios y de limitaciones del uso de PMBOK en las organizaciones. Fernández, K. et al., (2015) en un estudio sobre diversas metodologías, definen los siguientes beneficios:

- Proporciona toda la orientación sobre los conocimientos para la gestión de proyectos.
- Proporciona procesos completos paso a paso sobre la gestión de un proyecto.
- Tiene una estructura organizada por grupos de procesos y áreas de conocimiento.
- Muestra una lista de las herramientas y técnicas que se pueden utilizar en un proceso específico.
- Incorpora el conocimiento sobre los aspectos específicos para la gestión de las adquisiciones (contratación).
- Proporciona orientación sobre liderazgo y otras habilidades interpersonales.
- Proporciona orientación sobre los informes de costos del proyecto para la presentación de informes eficaces.
- Proporciona orientación sobre el código de conducta profesional de un gerente de proyecto. (p.18)

Asimismo, las limitaciones encontradas fueron las siguientes:

- “- Define las salidas para los procesos descritos, pero poca orientación sobre el contenido de estas.
- No existe orientación sobre las responsabilidades del equipo de gestión de proyectos.
 - Poca orientación sobre cómo asimilar o implementar exactamente la guía, o una variante paso a paso.
 - No existe el conocimiento de la actividad post proyecto que asegure el éxito del mismo a partir del punto de vista de los beneficios.
 - No existen detalles del contenido de los planes que describan cómo deben ser implementados en cada uno de los niveles y por quién”. (p.18).

Para concluir lo concerniente a la Guía de PMBOK, resulta importante destacar que en su sexta edición se realizaron cambios importantes para incorporar metodologías ágiles en los fundamentos para la dirección de proyectos definidos por PMI.

Entre estos cambios destaca que, en la última edición, cada una de las áreas de conocimiento posee una sección en la que se explica la aplicación de metodologías ágiles en su

contexto. Asimismo, incluye un apéndice donde explican la gestión de proyectos en un contexto ágil entre otras prácticas iterativas.

Este apéndice es un esfuerzo realizado entre Agile Alliance y PMI, y se titula Guía práctica Agile. Estos son los puntos principales que se abarcan en dicho documento:

“-La mentalidad Agile: Expone el manifiesto ágil y explica los valores y principios de las metodologías ágiles, Kanban y de gestión esbelta (Lean) aplicados a la gestión de proyectos.

-Selección del ciclo de vida: Ayuda al Gerente de proyecto a escoger y adaptar el enfoque ágil más apropiado y a construir un híbrido si es necesario.

-Creación de un entorno ágil: Aborda los factores críticos de éxito para adoptar la gestión de proyectos con metodologías ágiles, como por ejemplo la construcción de equipos por medio de liderazgo de servicio en lugar de liderazgo de control o dictatorial.

-Entrega de un entorno ágil: Cubre la creación de valor para el área de negocio y la entrega de productos, incluyendo métricas y reportes.

-Consideraciones organizacionales: Lleva al Gerente de proyecto entre los distintos factores críticos de éxito tales como la cultura, gestión de cambios y el rol de la Oficina de gestión de proyectos (PMO).

-Mapa de la Guía del PMBOK: Une los puntos entre la gestión de proyectos predictiva y la gestión de proyectos con metodologías ágiles, aclarando similitudes y diferencias.

-Mapa del manifiesto ágil: Une los puntos entre los 12 principios del manifiesto ágil y la guía de prácticas Agile.

-Visión general de marcos de trabajo: Información sobre diversos marcos de trabajo ágiles tales como Scrum, programación extrema (XP), Kanban, Scrumban, Feature Driven Development (FDD), método de desarrollo de sistemas dinámico (DSDM) proceso unificado Agile (AUF) y marco de trabajo Agile por escala (Safe).

-Atributos que afectan la adaptación (Apéndice): Guía de alto nivel sobre cómo adaptar los marcos de trabajo ágiles a la situación particular.

-Herramientas para determinar la adecuación de metodologías ágiles: Más ayuda al Gerente de proyectos para seleccionar el enfoque apropiado y elaborar híbridos cuando sea necesario”. (pmoinformatica.com, 2017).

Se puede esperar en futuras ediciones del PMBOK, que se continúe con la línea de aumentar los elementos de metodologías ágiles presentes en su guía, dado el auge que estas han tenido en la ejecución de proyectos.

2.4.2.2 Projects in Controlled Environments (PRINCE2)

La metodología de Proyectos en entornos controlados, conocida por sus abreviaturas en inglés como PRINCE2, fue concebida para implementar una serie de mejoras en el desarrollo de los proyectos y así aumentar las probabilidades de éxito. Es uno de los métodos que originalmente surgió para la gestión de proyectos en la industria informática, sin embargo, con el paso de los años se fue aplicando a todo tipo de proyectos.

“Estuvo inicialmente basado en PROMPTII, método creado por Simfact Systems en 1975, que es adoptado por la CCTA como el estándar a utilizar en todos los proyectos del gobierno británico. La evolución de este modelo tomó en 1989 el nombre de PRINCE y en la actualidad es desarrollado por la organización británica en la que se integró la CCTA: OGC (Office Government Commerce). Posteriormente con la experiencia y contribución de expertos se enriquece el método, obteniéndose en 1996 la primera edición de PRINCE 2, método genérico, válido para todos los entornos, sin ser exclusivo para un tipo de proyecto en específico”. (Fernández, K. et al., 2015).

PRINCE2 tiene la particularidad de que es un método gubernamental, ya que debe ser aplicado de forma oficial en todos los proyectos que ejecuta el Gobierno de Reino Unido. Pese a lo anterior, también es muy común que en la Unión Europea sea adaptado por otros gobiernos y por la empresa privada.

La estructura de esta metodología se resume en la siguiente figura:

Temas	Procesos
Proceso de Negocio (<i>Business Case</i>).	[SU] Comienzo de un Proyecto (<i>Starting Up a Project</i>).
Organización (<i>Organization</i>).	[IP] Inicio de un Proyecto (<i>Initiating a Project</i>).
Calidad (<i>Quality</i>).	[DP] Dirigir un Proyecto (<i>Directing a Project</i>).
Planes (<i>Plans</i>).	[CS] Controlar una Fase (<i>Controlling a Stage</i>).
Riesgo (<i>Management of Risk</i>).	[MP] Gestión del Suministro de Productos (<i>Managing Product Delivery</i>).
Control del Cambio (<i>Change Control</i>).	[SB] Gestión del Límite de las Fases (<i>Managing Stage Boundaries</i>).
Progreso (<i>Progress</i>).	[CP] Cerrar un Proyecto (<i>Closing a Project</i>).
Técnicas	Roles
Planificación con Base en el Producto (<i>Product-basedplanning</i>).	Consejo/Junta Directiva (<i>Project Board</i>). Usuario Representativo (<i>Senior User</i>). Director Ejecutivo (<i>Executive</i>). Suministrador/Proveedor Representativo (<i>Senior Supplier</i>)
Revisión de la Calidad (<i>Qualityreview</i>).	Jefe de Proyecto (<i>Project Manager</i>). Jefe de Equipo (<i>Team Manager</i>). Responsable de Garantía (<i>Project Assurance</i>). Responsable de Soporte (<i>Project Support</i>).

Figura 14: Estructura PRINCE2

Fuente: Calvo, E. & Ramírez, M., 2016, p. 22

Como se aprecia, PRINCE2 está dividido en siete temas y siete procesos, los cuales se detallan en los siguientes párrafos.

Los temas que se abarcan son: Proceso de Negocio, Organización, Calidad, Planes, Administración del Riesgo, Control del Cambio y Progreso. A continuación, se describe el detalle de cada uno de ellos:

- “1. Caso de negocio: describe cómo es desarrollada la idea en una propuesta de inversión viable para la organización y cómo la gerencia de proyecto se centra en los objetivos de esta a lo largo del proyecto.
2. Organización: describe los roles y responsabilidades en la organización temporal del proyecto que son requeridos para una gestión eficiente.
3. Calidad: describe cómo se desarrolla el esquema original dentro de los criterios de calidad y cómo la gestión de proyectos asegura que estos criterios se alcancen posteriormente.
4. Planes: especifica qué planes pueden y deben estar presentes en el proyecto, quién es responsable de crearlos, cómo se va a llevar a cabo y qué elementos se deben incluir. Sugiere la técnica de planificación basada en el producto.
5. Riesgos: describe como la gestión de proyectos gestiona las incertidumbres en los planes y en un entorno más amplio del proyecto.

6. Cambios: describe cómo evalúa y actúa sobre problemas que tienen un impacto potencial en aspectos de la línea base del proyecto. Estos problemas pueden ser preocupaciones o problemas imprevistos, solicitudes de cambios o instancias de calidad fallida.

7. Progreso: se refiere a la viabilidad de los planes. Este explica el proceso de toma de decisiones para la aprobación de los planes, el monitoreo del rendimiento actual, las acciones correctivas a ser tomadas y el proceso de escalabilidad si se prevé que el rendimiento exceda la tolerancia acordadas”. (Fernández, K. et al., 2015, p. 13).

De igual forma, esta metodología cuenta con siete procesos, los cuales se describen a continuación:

“1. SU - Puesta en marcha de un proyecto (Starting Up a Project): Este es el primer proceso de PRINCE 2. Comienza con el nombramiento de las personas clave para el proyecto y la producción de un Mandato de Proyecto que perfila la necesidad de llevar a cabo un proyecto. Tiene como objetivo proporcionar un inicio controlado del proyecto, asegurar que esté disponible la información requerida por el resumen del proyecto, diseñar y nombrar el equipo de gestión del proyecto, genera el expediente del proyecto que incluye la descripción del caso de negocio preliminar, así como crear el plan de la fase de inicio.

2. IP - Inicio de Proyecto (Initiating a Project): Este proceso empieza cuando se ha producido la Autorización del Plan de la Fase de Inicio, junto al Enfoque del Proyecto y el Resumen de Proyecto, y el Comité de Proyecto aprueba el comienzo del mismo. Como objetivo incluye proponer los planes del proyecto, planificar la calidad de los productos a entregar, refinar el caso de negocio, definir cómo se identificarán y controlarán los riesgos y cambios del proyecto. De igual forma se prepara la estrategia de comunicación y se ordena el control del proyecto. Se crea el Documento de inicio del proyecto (PID).

3. DP - Dirección del Proyecto (Directing a Project): La dirección del proyecto actúa durante todo el ciclo de vida del proyecto, desde el arranque hasta el cierre. Este proceso proporciona la autorización del inicio del proyecto, la gestión de dirección y control, la comunicación con la Dirección Corporativa o de Programa y la confirmación del Cierre de Proyecto.

4. CS – Control de una Fase (Controlling a Stage): El Control de Fase se realiza una vez que los recursos han sido comprometidos y ha sido aprobado un Plan de Fase. El proceso

mantiene el centro de atención del Equipo de Gestión del Proyecto en la entrega de los productos dentro de las tolerancias previamente aceptadas. Este proceso es imperativo para el éxito del proyecto y éste se logra mediante el control día a día del trabajo que está realizándose.

5. MP – Gestión de la Entrega de Productos (Managing Product Delivery): La Gestión de Entrega de Productos permite al Team Manager-Responsable de Equipo: acordar con el Project Manager-Responsable de Proyecto el trabajo a llevar a cabo, asegurar que el paquete de trabajo esté completo y entregar el Paquete de Trabajo completado al Project Manager-Responsable de Proyecto.

6. SB – Gestión de los Límites de Fase (Managing Stage Boundaries): Este proceso asegura que los productos de la fase actual hayan sido completados como se definieron para que el Comité de Proyecto valore y determine si el proyecto continúa siendo viable. Se registran las lecciones aprendidas de la fase actual, se planifica y se obtiene la autorización para la siguiente fase.

7. CP – Cierre del Proyecto (Closing a Project): Este proceso tiene como objetivo que los objetivos que se encontraban en el Documento de Inicio del Proyecto (PID – Project Initiation Document) han sido alcanzados, y confirma la satisfacción de los Clientes y la aceptación del producto. También recomienda acciones de seguimiento, documentos y/o lecciones aprendidas en el proyecto y se crea un Informe de Fin de Proyecto”. (Fernández, K. et al., 2015, p. 12).

Al igual que la Guía de PMBOK, PRINCE2 presenta una serie de beneficios, así como algunas limitaciones. Con respecto a los beneficios cabe señalar los siguientes:

- “- Ofrece las prácticas de gestión de proyectos con mayor profundidad (How to).
- El conocimiento es organizado en torno a principios, procesos y componentes.
- Proporciona la descripción y plantillas de los productos de gestión de proyectos.
- Define roles y responsabilidades, los cuales son consensuados sobre una estructura organizativa que se apoya en los intereses del negocio, los usuarios y proveedores.
- Énfasis en el caso de negocio y en el logro de los beneficios.
- Se puede adaptar para corresponder al entorno del proyecto, en cuanto al tamaño, entorno, complejidad, importancia, capacidad y riesgo del proyecto.

- Centrada en el producto, está orientado a la definición y entrega de productos, en particular de sus requisitos de calidad.
- Gestión por Excepción permitiendo un mayor enfoque en las áreas que requieren atención.
- Gestión por etapas, solamente una etapa es comprometida a la vez”. (Fernández, K. et al., 2015, p. 18).

Con respecto a las limitaciones de esta metodología, se identificaron las siguientes:

- No se garantizan técnicas detalladas para planificar y llevar a cabo las actividades del proyecto.
- Las habilidades de liderazgo y gestión de personas no están cubiertas en el método.
- Poca flexibilidad ofrecida en la adaptación de las mejores prácticas, ya que es más un método.
- Poca énfasis en la orientación o el desarrollo de gestores de proyectos en torno al conjunto de habilidades. (Fernández, K. et al., 2015)

2.4.2.3 Comparación entre PMBOK y PRINCE2

Pese a que existen diferencias marcadas entre PMBOK y PRINCE2, también existen similitudes en la forma propuesta para la gestión de proyectos. Es decisión de cada organización decidir cuál metodología es más conveniente adoptar, analizando el contexto en que se desarrolla, así como el tipo de proyectos.

Para resumir esas diferencias, en las siguientes tablas se muestra una comparación de estas dos metodologías en función de sus características generales, estructura y procesos que abarcan.

En la siguiente tabla, se muestran las diferencias según sus características generales:

Tabla 4: Análisis de las características generales de PMBOK y PRINCE2

PMBOK	PRINCE2
Colección de buenas prácticas para la gestión de proyectos.	Método de gestión de proyectos.
Descriptivo.	Prescriptivo

Impulsado por los requisitos del cliente.	Impulsado por un caso de negocio.
Cada tema se puede consultar aisladamente del resto.	Un conjunto integrado de procesos y componentes (no son elementos aislados que se pueden aplicar de forma independiente).
Orientado a los gerentes de proyectos.	Cubre los roles de la gestión de proyectos, definiendo los roles y sus responsabilidades.
Cubre las competencias interpersonales.	No cubre las competencias interpersonales.
Tiene dos niveles de certificación CAMP, PMP.	Tiene tres niveles de certificación Foundation, Practitioner, Professional.
Incluye la gestión por fases.	Se divide el proyecto en una serie de fases que facilita su planificación, asignación, tareas, supervisión y control.

Fuente: (Fernández, K. et al., 2015)

En referencia a la diferencia entre la composición de los procesos, en la siguiente tabla se muestra una equivalencia entre ambas metodologías:

Tabla 5: Relación entre los procesos de PMBOK y los procesos de PRINCE2

PMBOK	PRINCE2
Iniciación	Puesta en marcha de un proyecto. Dirección de un proyecto
Planificación	Inicio del proyecto. Gestión de los límites de la fase. Gestión de la entrega de productos.
Ejecución	Control de la fase. Gestión de la entrega de productos.
Seguimiento y control	Dirección del proyecto. Control de la fase. Gestión de los límites de la fase.
Cierre	Gestión de los límites de la fase.

 Cierre del proyecto

Fuente: (Fernández, K. et al., 2015)

Tal y como se aprecia, varios de los procesos de PRINCE2 se traslapan en los procesos de PMBOK, dado que este último los agrupa de forma más general.

Por último, es posible realizar una relación entre las áreas de conocimiento del PMBOK y las temáticas, procesos, principios y temas de PRINCE2, los cuales se aprecian en la siguiente tabla:

Tabla 6: Relación entre las áreas de conocimientos definidas en PMBOK y los procesos y componentes de PRINCE2

PMBOK	PRINCE2
Integración	-Justificación continúa del negocio. -Gestión por excepción.
Tiempo	-Plan de proyecto, plan de fase, plan de equipo, plan de revisión de beneficios.
Alcance	-Planes, Caso de negocio, Progreso. -Técnica de planificación basada en el producto, donde se realiza la estructura de descomposición del producto (EDT), se escriben las descripciones detalladas y se realiza el diagrama de flujo.
Costos	-Solamente se aborda de manera general algunos aspectos referentes al valor ganado.
Calidad	-Orientación al producto. -Lecciones aprendidas y mejora continua. -Gestión de la configuración.
Riesgos	-Similar a PMBOK.
Comunicaciones	-Progreso. -Estrategia de gestión de las comunicaciones.

Recursos humanos, involucrados	Se definen los roles y responsabilidades del equipo de gestión del proyecto.
Adquisiciones	No se cubre.
Interesados	Estrategia de gestión de comunicaciones.

Fuente: (Fernández, K. et al., 2015)

De acuerdo con la tabla anterior, de alguna u otra forma la mayoría de las áreas de conocimiento de PMBOK están cubiertas en PRINCE2. En caso de Recursos Humanos PRINCE2 define un mayor nivel de detalle, definiendo roles y responsabilidades del equipo, sin embargo, no cubre el área de adquisiciones y en materia de costos lo que se abarca es muy básico con respecto al PMBOK.

CAPÍTULO III: MARCO METODOLÓGICO

A lo largo de este capítulo, el lector puede conocer la metodología empleada en la investigación, con el fin de responder al problema planteado, así como para cumplir los objetivos definidos.

Para ello se define el tipo de investigación realizada, las fuentes de información utilizadas para dar sustento al proyecto, las técnicas y herramientas que se aplicaron para la recolección de datos, las variables definidas en función de los objetivos.

Adicionalmente se establece el diseño de la investigación en el que se detallan las distintas fases en las que se estructuró el proyecto. Resumiendo lo anterior, se presenta una matriz de coherencia que permite una mejor comprensión de todos los elementos antes descritos.

3.1 TIPO Y ENFOQUE DE LA INVESTIGACIÓN

3.1.1 Tipo de investigación

Para el presente proyecto se definió una investigación de tipo aplicada, dado que se ponen en práctica una serie de conocimientos en la búsqueda de resolver el problema identificado en las etapas iniciales.

Al respecto, Lozada (2014) propone lo siguiente:

“La investigación aplicada es un proceso que permite transformar el conocimiento teórico que proviene de la investigación básica en conceptos, prototipos y productos, sucesivamente. La elaboración de conceptos debe obligatoriamente contar con la participación de los usuarios finales y la industria para que responda a las necesidades reales de la sociedad. Bajo estas condiciones, una estrecha colaboración entre la academia y la industria puede generar un elevado valor agregado en la sociedad, por la creación de nuevos procesos o productos”. (p.38).

En este caso, mediante la metodología de desarrollo de sistemas de información que se propone como solución al problema identificado, se lleva a la práctica una serie de conocimientos y teorías que benefician a la Unidad de Tecnologías de Información de la Contraloría General de la República.

3.1.2 Enfoque de la investigación

Para la presente investigación se utiliza un enfoque cualitativo, con el cual se busca estudiar y analizar el proceso de desarrollo de sistemas de la UTI, con el fin de descubrir y construir una solución ante la problemática definida en el proyecto.

Como parte de una investigación con enfoque cualitativo se identifican las siguientes características:

“-Busca describir, comprender e interpretar los fenómenos, a través (sic) de las percepciones y significados producidos por las experiencias de los participantes.

-Se aplica la lógica inductiva. De lo particular a lo general (de los datos a las generalizaciones —no estadísticas— y la teoría).

- Diseño abierto, flexible, construido durante el trabajo de campo o realización del estudio.

-La recolección de los datos está orientada a proveer de un mayor entendimiento de los significados y experiencias de las personas. El investigador es el instrumento de recolección de los datos, se auxilia de diversas técnicas que se desarrollan durante el estudio. Es decir, no se inicia la recolección de los datos con instrumentos preestablecidos, sino que el investigador comienza a aprender por observación y descripciones de los participantes y concibe formas para registrar los datos que se van refinando conforme avanza la investigación.

-Por lo general, el análisis de los datos no se inicia con ideas preconcebidas sobre cómo se relacionan los conceptos o variables. Conforme se van reuniendo los datos verbales, en texto y/o audiovisuales, se integran en una base de datos, la cual se analiza para determinar significados y describir el fenómeno estudiado desde el punto de vista de sus actores. Se conjuntan descripciones de participantes con las del investigador”. (Hernández, R., Fernández, C., & Baptista, M., 2014, p. 12).

De acuerdo con lo anterior, se evaluará la percepción de las personas involucradas en el proceso de desarrollo de sistemas y se estudiará el proceso como tal. Una vez recopilada la información, se procederá a ejecutar un análisis para construir la solución propuesta.

3.2 FUENTES Y SUJETOS DE INFORMACIÓN

En este apartado se describen las fuentes y sujetos de información utilizados para sustentar la presente investigación. Con base en ellos, se extrajeron datos que se utilizaron como puntos de referencia para fundamentar tanto los aspectos teóricos como prácticos de la propuesta desarrollada.

3.2.1 Fuentes primarias

Para el presente proyecto se utilizaron las siguientes fuentes primarias:

- Encuestas al equipo de desarrollo de sistemas de la Unidad de Tecnologías de Información de la CGR.
- Entrevistas a funcionarios responsables del equipo de desarrollo de sistemas de la Unidad de Tecnologías de Información de la CGR.
- Guías de Administración de proyectos del PMI.
- Marco de trabajo Scrum.
- Marco de trabajo de Programación Extrema.

3.2.2 Fuentes secundarias

Las siguientes son las fuentes secundarias utilizadas en el desarrollo del proyecto:

- Trabajos finales de graduación.
- Revistas tecnológicas.
- Páginas Web con información de metodologías de desarrollo de software y gestión de proyectos.
- Informes de auditoría.

3.2.3 Sujetos de información

A continuación, se presenta una tabla con las personas contactadas con el fin de recopilar información para la elaboración del presente proyecto:

Tabla 7. Sujetos de Información

Descripción	Profesión	Experiencia	Relación con el tema
Jefe de Unidad	Ingeniería en computación	27 años	Jefe de la Unidad de Tecnologías de Información de la CGR.
Asistente Técnico (Supervisor)	Ingeniería en computación	25 años	Supervisor del equipo de desarrollo de aplicaciones de la UTI.
Desarrollador	Ingeniería en Sistemas	21 años	Desarrollador de aplicaciones de la UTI.
Administrador de Base de Datos	Ingeniería en Computación	24 años	Administrador de las Bases de Datos de la UTI.

Fuente: Elaboración propia

3.3 TÉCNICAS Y HERRAMIENTAS DE RECOLECCIÓN DE DATOS

En esta sección se explican las técnicas y herramientas utilizadas en el desarrollo del proyecto, por medio de las cuales se recopiló información para lograr el resultado planteado.

A continuación, se detallan dichas técnicas:

- **Revisión documental:** Se realizó un análisis de distintas fuentes de información como libros, revistas, páginas web, investigaciones, marcos de referencia, estándares, entre otro material bibliográfico. Esto con el fin de obtener suficientes insumos técnicos para sustentar la propuesta planteada en la investigación.

- Entrevista: De acuerdo con Hernández, R. et al., (2014), entrevista se define de la siguiente forma:

“Se define como una reunión para conversar e intercambiar información entre una persona (el entrevistador) y otra (el entrevistado) u otras (entrevistados). En el último caso podría ser tal vez una pareja o un grupo pequeño como una familia o un equipo de manufactura. En la entrevista, a través de las preguntas y respuestas se logra una comunicación y la construcción conjunta de significados respecto a un tema”. (p.403).

En el desarrollo del proyecto se entrevistó a personas clave del proceso de desarrollo de software de la UTI, con el fin de recabar información sobre las expectativas y puntos de mejora del proceso que se ejecuta actualmente.

- Encuestas: Según Hernández, R. et al., (2014), se definen las encuestas como:
“Son métodos de recolección de información que se usan para describir, comparar o explicar conocimientos, sentimientos, valores, preferencias y conductas. Es decir, en la literatura clásica sobre metodología de la investigación, la encuesta se considera como una técnica cuantitativa para recabar, mediante preguntas, datos de un grupo seleccionado de personas”. (p.159).

En la presente investigación se aplica una encuesta al equipo de desarrollo de aplicaciones de la UTI con el fin de conocer la percepción sobre el proceso actual, recopilar información sobre las oportunidades de mejora que podrían aplicarse y medir el conocimiento en temas de otras metodologías de desarrollo y la apertura a las mismas.

3.4 VARIABLES DE INVESTIGACIÓN

Las variables que fueron definidas en función de los objetivos definidos para el proyecto, se presentan en la tabla que se muestra a continuación:

Tabla 8: Variables de Investigación

Objetivos Específicos	Variables asociadas	Descripción
Diagnosticar la situación actual del proceso de desarrollo de sistemas de información, mediante la identificación de procedimientos que se ejecutan actualmente y la normativa aplicable, con el fin de identificar las necesidades de la Institución.	Procedimientos de desarrollo de sistemas. Normativa aplicable al desarrollo de sistemas en la institución.	Se elabora un diagnóstico del proceso de desarrollo de sistemas que posibilita la identificación de las necesidades actuales de la institución.
Ejecutar un análisis de la metodología de desarrollo de sistemas que se utiliza actualmente en la Institución mediante la aplicación de técnicas de recolección de información, para evaluar si responde a las necesidades institucionales.	Metodología de desarrollo de sistemas.	Se realiza un análisis de la metodología de desarrollo vigente en la Contraloría para determinar si responde a las necesidades identificadas en el diagnóstico.
Identificar las mejores prácticas, estándares y metodologías para el desarrollo de software que podrían implementarse para satisfacer las necesidades de la institución, mediante un análisis comparativo.	Mejores prácticas, estándares y metodologías para el desarrollo de software.	Se documenta un análisis comparativo entre las metodologías, mejores prácticas y estándares que más se utilizan en el desarrollo de software identificadas.

Desarrollar una propuesta de mejora a la metodología de desarrollo de sistemas de información que responda a las necesidades actuales de la Institución.	Metodología propuesta.	Se crea una propuesta de metodología de desarrollo de aplicaciones, ajustada a las necesidades identificadas de la institución.
--	------------------------	---

Fuente: Elaboración propia

3.5 DISEÑO DE LA INVESTIGACIÓN

Para la presente investigación se definieron cuatro etapas, las cuales se muestran gráficamente en el siguiente diagrama:

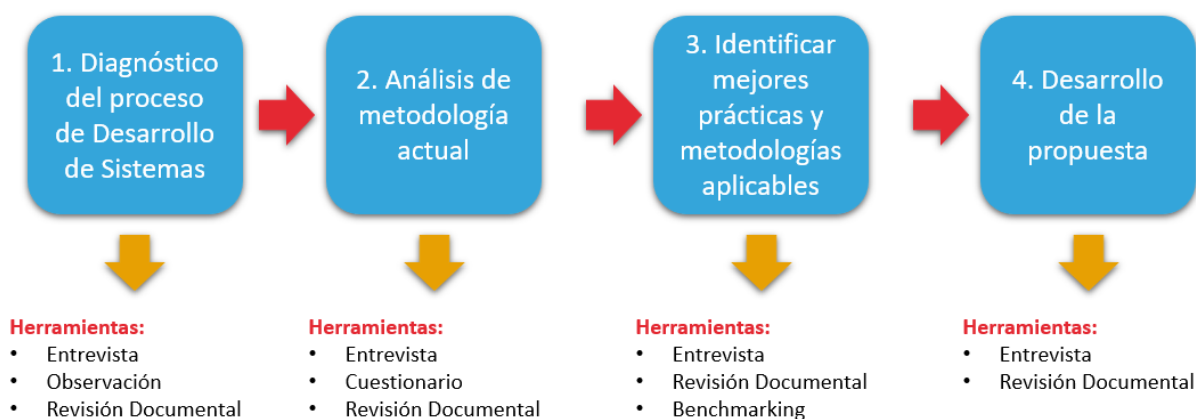


Figura 15: Etapas de la investigación

Fuente: Elaboración propia

A continuación, se realiza una descripción de esas cuatro etapas con sus respectivas herramientas:

- Diagnóstico del proceso de Desarrollo de Sistemas: En esta etapa se ejecutará un diagnóstico para conocer la situación actual del proceso de desarrollo de sistemas que se ejecuta en la UTI para identificar las necesidades institucionales en esta materia.

Asimismo, se hará una revisión del marco normativo aplicable, el cual debe ser tomado en cuenta a la hora de diseñar la propuesta en etapas posteriores.

Para alcanzar este fin se aplican entrevistas tanto al jefe de la unidad, como al asistente técnico que es el responsable del equipo. Asimismo, se realiza una observación del proceso en sus distintas fases de acuerdo con los proyectos que se estén desarrollando. En adición a lo anterior, se revisa la documentación de proyectos finalizados, así como de distintas métricas que se llevan en el proceso.

- **Análisis de la metodología actual:** Para esta fase se documenta el análisis de la metodología actual, con el fin de evaluar cómo se está respondiendo a las necesidades identificadas en la fase de diagnóstico. Para lograr el objetivo se aplica un cuestionario a los miembros del equipo de desarrollo y se realizarán entrevistas para ahondar en algunos temas de interés. Asimismo, para fundamentar el análisis, se realiza una revisión documental de la metodología actual y de los planes institucionales relacionados con el desarrollo de sistemas.
- **Identificar mejores prácticas y metodologías aplicables:** En esta fase se documenta un análisis comparativo entre las distintas metodologías de desarrollo de sistemas, mejores prácticas y estándares más utilizados en la industria de desarrollo de software. Para tales efectos se realizan entrevistas a desarrolladores para determinar mejores prácticas, se elabora un *benchmarking* y como resultado, una matriz con un análisis comparativo de los aspectos antes mencionados. Una vez realizado este análisis y tomando en cuenta las necesidades anteriores, se tiene claridad en cuáles metodologías, estándares y buenas prácticas se deben contemplar para el diseño de la propuesta de metodología.
- **Desarrollo de la propuesta:** Con base en la información recabada en las etapas anteriores, se desarrolla una propuesta de metodología de desarrollo, la cual responde a las necesidades institucionales detectadas. Para ello se emplean las mejores prácticas, metodologías y estándares estudiados en la etapa anterior. Para esta construcción, se realizan entrevistas con los encargados del proceso de desarrollo con el fin de validar que la propuesta es viable. Asimismo, se revisa la documentación obtenida en etapas anteriores

con el fin de aplicar los resultados obtenidos del diagnóstico y los análisis que se ejecutaron.

3.6 MATRIZ DE COHERENCIA

En la siguiente tabla se detalla la matriz de coherencia, con la cual se aprecia la relación entre los objetivos específicos, los entregables y la fase de la investigación en la que se desarrolla, técnicas y metodologías de recolección de información, instrumentos y temas relacionados para el marco teórico.

Tabla 9: Matriz de coherencia

Objetivo	Entregable	Etapas de la metodología del proyecto que posibilita la realización del entregable	Técnicas/métodos de recolección de la información	Instrumentos	Temas relacionados para marco teórico
Diagnosticar la situación actual del proceso de desarrollo de sistemas de información, mediante la identificación de procedimientos que se ejecutan actualmente y la normativa aplicable, con el fin de identificar las necesidades de la Institución.	Documento descriptivo de la situación actual del proceso de desarrollo de sistemas de información de la CGR y el marco normativo que le aplica.	Diagnóstico de la situación actual del proceso de desarrollo de sistemas y la normativa que le aplica.	Entrevista	Minutas de entrevistas con personal de la UTI. Documento de texto con el diagnóstico de la situación actual e identificación de necesidades. Diagrama del proceso realizado con la aplicación Bizagi	Concepto de diagnóstico de la situación actual. Planes de Tecnologías de Información

Ejecutar un análisis de la metodología de desarrollo de sistemas que se utiliza actualmente en la Institución mediante la aplicación de técnicas de recolección de información, para evaluar si responde a las necesidades institucionales.	Documento de análisis de la metodología de desarrollo.	Análisis de la metodología de desarrollo de software.	Entrevista Cuestionario	Minutas de entrevistas con actores involucrados en metodología de desarrollo de software. Encuesta para actores involucrados	Evaluación de sistemas de información.
Identificar las mejores prácticas, estándares y metodologías para el desarrollo de software que podrían implementarse para satisfacer las necesidades de la CGR, mediante un análisis comparativo.	Documento con análisis comparativo entre metodologías, mejores prácticas y estándares más utilizados en la industria de desarrollo de software.	Análisis de la metodología de desarrollo de software. Benchmarking.	Entrevista	Minutas de entrevistas con desarrolladores. Matriz de análisis comparativo.	Gestión de Proyectos Estándares de Administración de Proyectos. Gestión de proyectos de desarrollo de software Mejores prácticas en desarrollo de software. Estándares de desarrollo de software.
Desarrollar una propuesta de mejora a la metodología de desarrollo de sistemas de información que responda a las necesidades actuales de la CGR.	Documento con la propuesta de mejora a la metodología de desarrollo de sistemas de información	Diseño de Propuesta	Entrevista Análisis y aplicación de datos y resultados obtenidos	Minutas con entrevistas con personal de la UTI. Documento con propuesta de mejora.	Propuesta de Mejora. Desafíos en fiscalización.

Fuente: Elaboración propia

CAPÍTULO IV: DIAGNÓSTICO DE LA SITUACIÓN ACTUAL

Con los análisis desarrollados en el presente capítulo, se quiere reflejar la situación actual de la metodología de desarrollo de sistemas de información de la Contraloría General de la República.

Se realiza un estudio del proceso actual, en el diagnóstico administrativo. Asimismo, se toma en cuenta la percepción que tienen sus principales usuarios, así como la que están teniendo los clientes, producto de la aplicación de la metodología por medio de un diagnóstico de percepción. Al concluir el capítulo, se puede tener un panorama claro de las brechas que se pretenden cerrar con la propuesta de metodología del capítulo cinco.

4.1 Diagnóstico Administrativo

El desarrollo de sistemas de información en la CGR forma parte del Proceso de Gestión de Tecnologías de Información y Comunicación del Manual General de Fiscalización Integral, que es el instrumento normativo de mayor jerarquía en la regulación y descripción general de los procesos que deben aplicarse en la institución.

Como principal herramienta de ese proceso, se encuentra la Metodología de Desarrollo de Proyectos de Tecnologías de Información y Comunicación (en adelante “la Metodología”) que actualmente se utiliza para el desarrollo de sistemas en la UTI.

El objetivo de esta metodología es el siguiente:

“esbozar una serie de pasos comunes a (sic) seguir, con el fin de mejorar las probabilidades de éxito de los proyectos, teniendo siempre presente que en última instancia dicho éxito está en función del nivel de motivación y mística de que estén impregnados los integrantes del equipo de trabajo, de la disponibilidad de recursos y del nivel de apoyo que brinde oportunamente la alta gerencia.” (Unidad de Sistemas y Tecnologías de la Información - CGR, 2016, p. 1)

Esta metodología se elaboró en atención a lo dispuesto en las Normas Técnicas para la Gestión y el Control de las Tecnologías de Información (N-2-2007-CO-DFOE), publicadas en La Gaceta N° 119 del 21 de junio de 2007 donde se dispone lo siguiente:

“La organización debe asegurar el logro de los objetivos propuestos como parte de la gestión de TI, para lo cual debe establecer un marco de referencia y un proceso de seguimiento en los que defina el alcance, la metodología y los mecanismos para vigilar la gestión de TI. Asimismo, debe determinar las responsabilidades del personal a cargo de dicho proceso.” (Contraloría General de la República, 2007, p. 11)

Para el desarrollo de sistemas de información en la CGR, es obligatorio que se aplique la Metodología, según dictan las Directrices sobre Seguridad y Buen Uso de las Tecnologías de Información y Comunicaciones (R-DC-0004-2015). Asimismo, en el Marco General para la Gestión de la Calidad en Tecnologías de Información y Comunicaciones (R-DC-120-2012), se establece la obligatoriedad de que los proyectos tecnológicos sean diseñados con el apoyo de dicha metodología, con el fin de que los productos tengan la calidad esperada.

En la metodología se define una estructura organizativa para cada proyecto, en ella destacan los siguientes integrantes:

- Comité Gerencial de Tecnologías de Información y Comunicaciones (CGTIC): Por delegación del máximo jerarca (Contralor o Contralora General), es el máximo ente en lo referente a recomendar sobre las directrices y lineamientos por seguir en la planificación y dirección de los procesos de desarrollo tecnológico. Está conformado por representantes de alto nivel gerencial, por el Jefe de Auditoría Interna como asesor del comité y por el jefe la UTI. Este comité reporta al máximo jerarca, quien lo preside y define su integración.
- Unidad ejecutora: Es el ente coordinador de los aspectos relacionados con el desarrollo del proyecto de TIC. Está conformada por el patrocinador, coordinador y líder del proyecto, así como el líder tecnológico.
- Patrocinador del proyecto: Es el máximo jerarca (o en quien delegue), de la unidad organizacional para la cual se va a desarrollar el proyecto de TIC.
- Coordinador de proyectos: Este rol es asumido por la jefatura de la UTI o delegado en un funcionario de la misma unidad, para velar por la adecuada ejecución de los proyectos de TIC observando aspectos como integración, calidad, logro de objetivos y eficiencia en los diseños y desarrollo de las soluciones.

- Líder del proyecto: Es un funcionario con gran conocimiento de su área funcional, aspecto por el cual se le ha conferido la capacidad de tomar decisiones y la responsabilidad de liderar activamente el proyecto; Vela tanto por los mejores intereses de la Contraloría General de la República, como por los de su área en lo que al proyecto concierne. En casos justificados y por recomendación del CGTIC, la dirección puede ser asumida por un representante de la UTI.
- Líder tecnológico del proyecto: Este es un funcionario al cual, por su formación en el área de Informática, experiencia y capacidad, se la ha conferido la responsabilidad de administrar los aspectos tecnológicos de un proyecto de TIC.
- Experto tecnológico: Es el recurso profesional que por su conocimiento tecnológico en la materia propia del proyecto que se está desarrollando, trabaja bajo la coordinación y dirección del líder tecnológico, en el desarrollo de la solución.

La metodología comprende las etapas de: anteproyecto, iniciación, planeación, ejecución, control y conclusión, las cuales se ilustran en la siguiente figura:

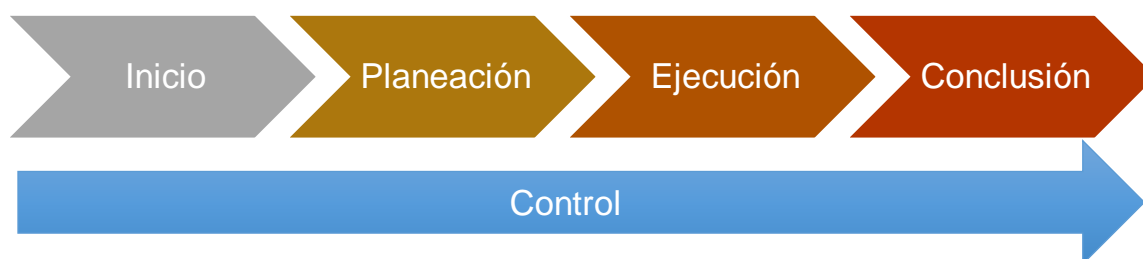


Figura 16: Fases de un proyecto

Fuente: Elaboración propia con base en información de la Metodología para el desarrollo de proyectos de TIC – CGR

Para el desarrollo de sistemas de información automatizado, en la metodología se detallan una serie de fases más específicas, las cuales se describen a continuación:

- I. Análisis integral de requerimientos: En esta fase se identifican, analizan y documentan los requerimientos funcionales y no funcionales que deben soportar el sistema o la solución propuesta.

Se prepara un documento de análisis el cual debe contener la siguiente información:

- a. Especificación de requerimientos: El líder tecnológico y el analista designado para el proyecto; trabajan con un grupo de usuarios para definir los requerimientos de

una manera comprensible. En la metodología se indica que puede utilizarse la descripción narrativa de requerimientos, el prototipo de interfaz de usuario o casos de uso.

- b. Beneficios esperados
- c. Recursos para desarrollo
- d. Definición de infraestructura tecnológica adicional
- e. Cronograma ajustado de las fases posteriores del proyecto
- f. Aprobación de los requerimientos

En la siguiente figura se ilustra lo contemplado en esa fase:

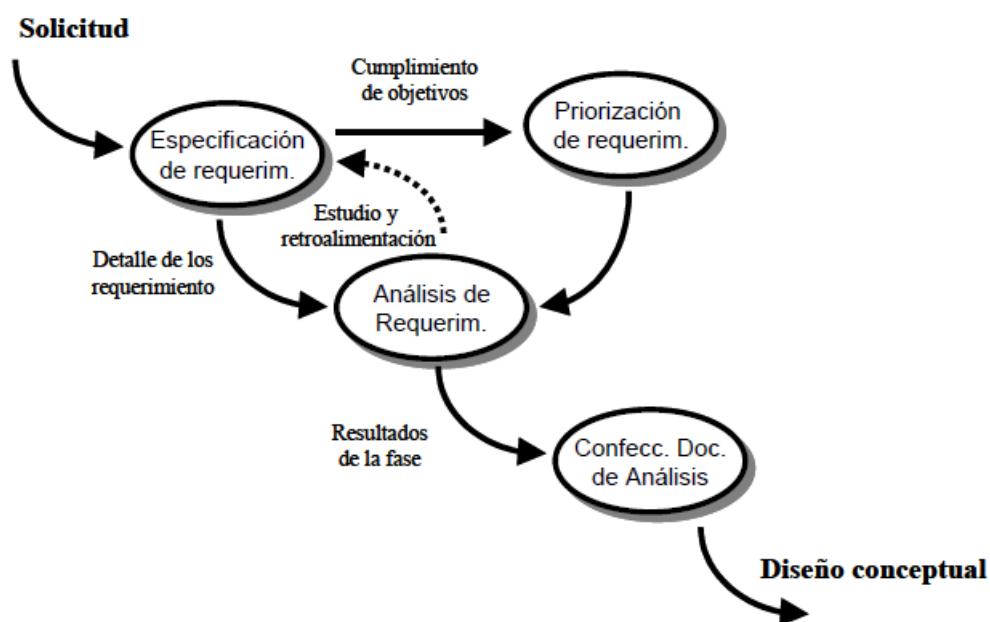


Figura 17: Proceso de análisis de requerimientos

Fuente: Metodología para el Desarrollo de Proyectos de TIC (2016)

- II. Diseño conceptual de la solución: En esta fase se identifican los primeros elementos de diseño del nuevo sistema.

Para esta etapa se debe documentar lo siguiente:

- a. Descripción de módulos y sub-módulos y su interacción
- b. Diseño de la interfaz de usuario

- c. Identificación de interrelaciones con otros sistemas o módulos
- d. Identificar roles de usuario

En la siguiente figura se diagrama lo contemplado en esta fase:

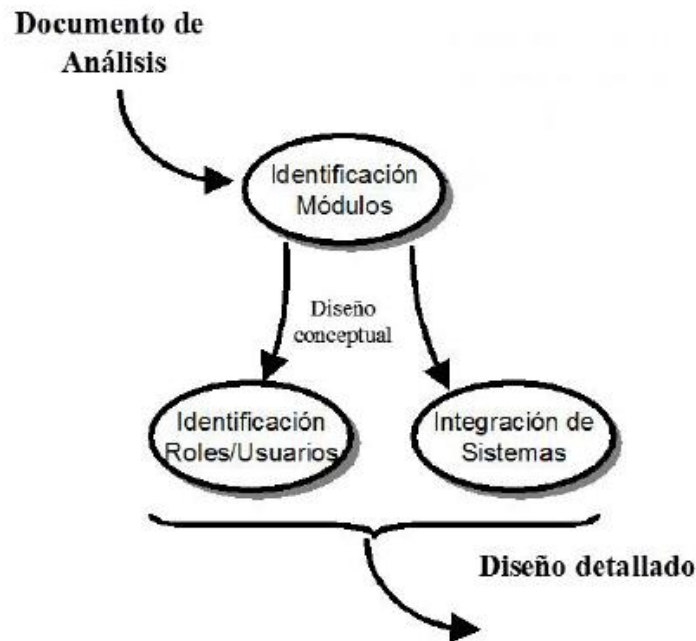


Figura 18: Proceso de diseño conceptual

Fuente: Metodología para el Desarrollo de Proyectos de TIC (2016)

III. Diseño detallado de la aplicación: En esta fase se establecen, con mayor detalle, las características que debe tener el nuevo sistema.

Se prepara un documento de diseño detallado y un documento con el diseño de pruebas.

El documento de diseño detallado debe contener la siguiente información:

- a. Descripción de procesos
- b. Diagrama lógico del modelo de datos
- c. Diseño físico de la base de datos
- d. Estimación del volumen de datos
- e. Definición de controles y seguridad por utilizar
- f. Organización para la operación del sistema

El documento de diseño de pruebas debe incluir lo siguiente:

- a. Especificación de tipos de pruebas

- b. Requerimientos para las pruebas
- c. Casos y datos de prueba
- d. Usuarios para las pruebas

En la figura que se presenta a continuación se ilustra este proceso:

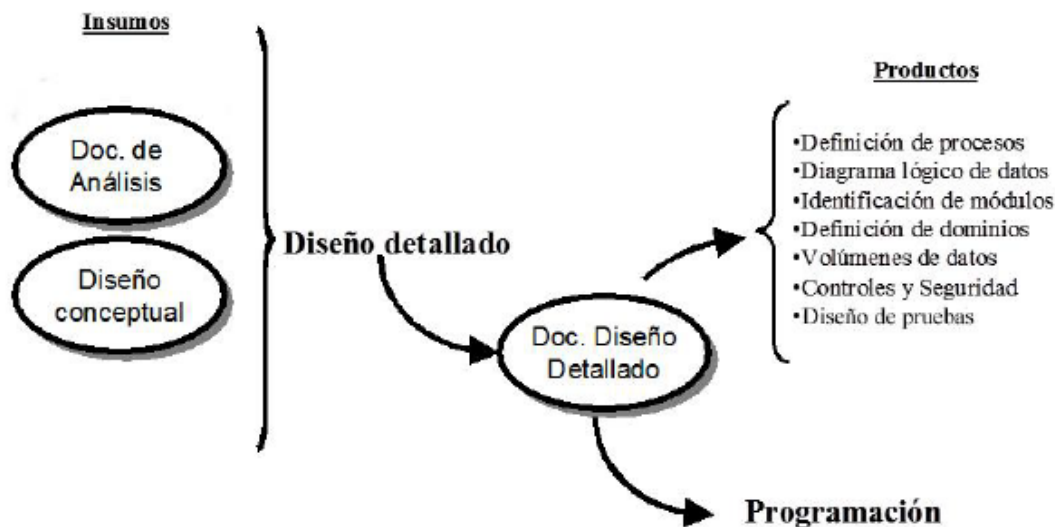


Figura 19: Proceso de diseño detallado

Fuente: Metodología para el Desarrollo de Proyectos de TIC (2016)

IV. Programación y pruebas: Esta fase comprende el desarrollo de los programas y la realización de pruebas a partir de los documentos de análisis y diseño.

Los entregables de esa etapa son los siguientes:

- a. *Scripts* de creación de objetos en la base de datos
- b. Componentes de programación (Elementos de programación del nuevo sistema)
- c. Documentación de pruebas (donde se indiquen los resultados de las pruebas realizadas)
- d. Aceptación del sistema

En la figura que se muestra a continuación se ilustra esta fase:

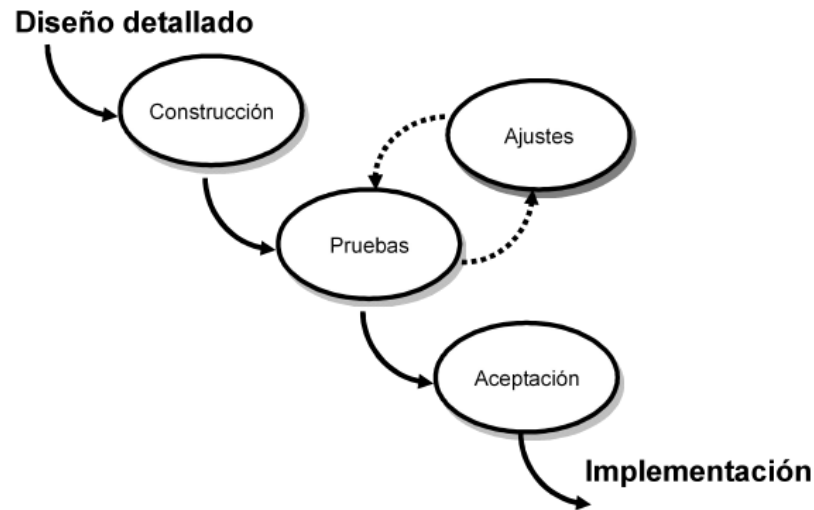


Figura 20: Construcción y pruebas

Fuente: Metodología para el Desarrollo de Proyectos de TIC (2016)

- V. Documentación: Se genera en todas las etapas del proyecto. Debe mantenerse actualizada y debidamente ordenada en el expediente del proyecto.
- VI. Capacitación: El patrocinador debe elaborar e impartir la capacitación a los usuarios del sistema.
- VII. Implementación: En esta etapa se define el método de implementación más adecuado para el sistema, considerando por ejemplo si es necesario un paralelo, el método de carga de datos y otras decisiones estratégicas con el fin de concluir el proceso de manera exitosa. Se debe preparar un plan de implementación donde se documente lo siguiente:
 - a. Instalación del sistema
 - b. Conversión y carga inicial de datos
 - c. Ejecución del paralelo (en caso de que se utilice)

En la siguiente figura se ilustra este proceso:

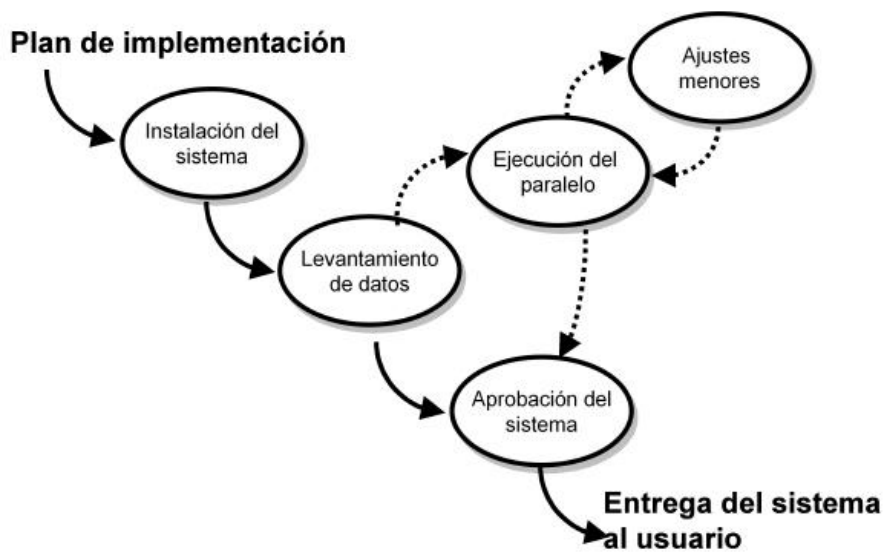
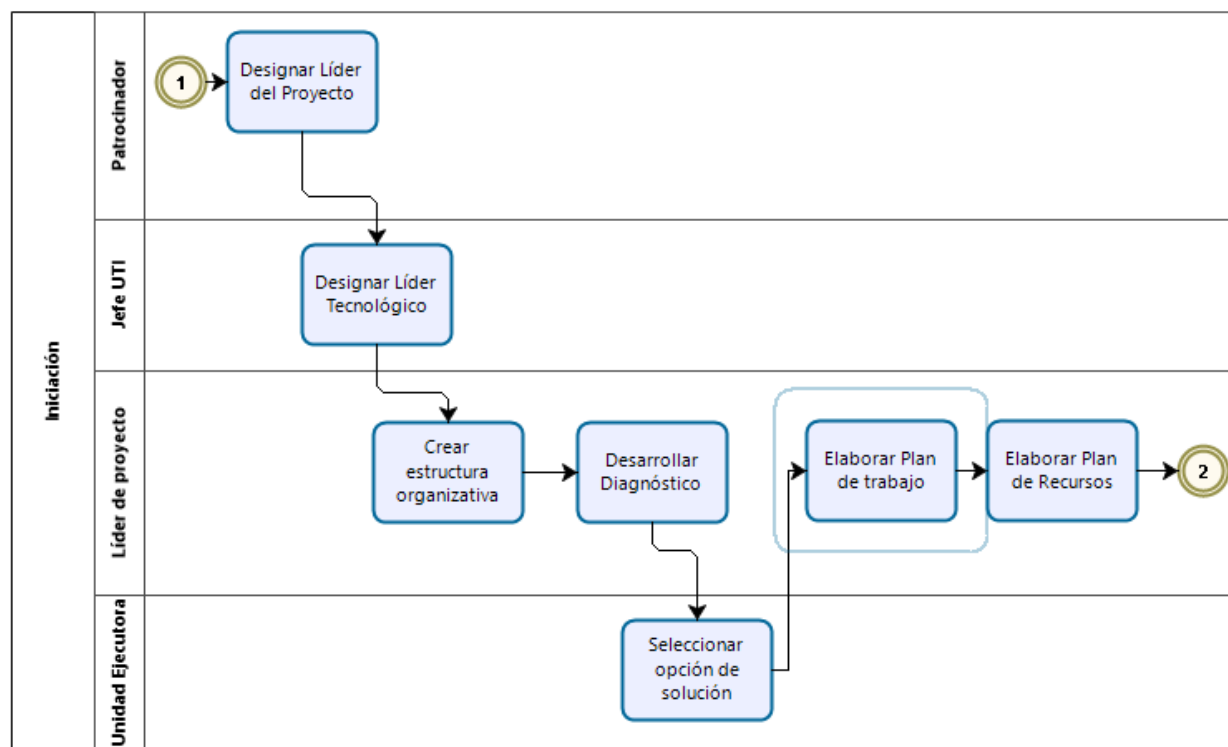
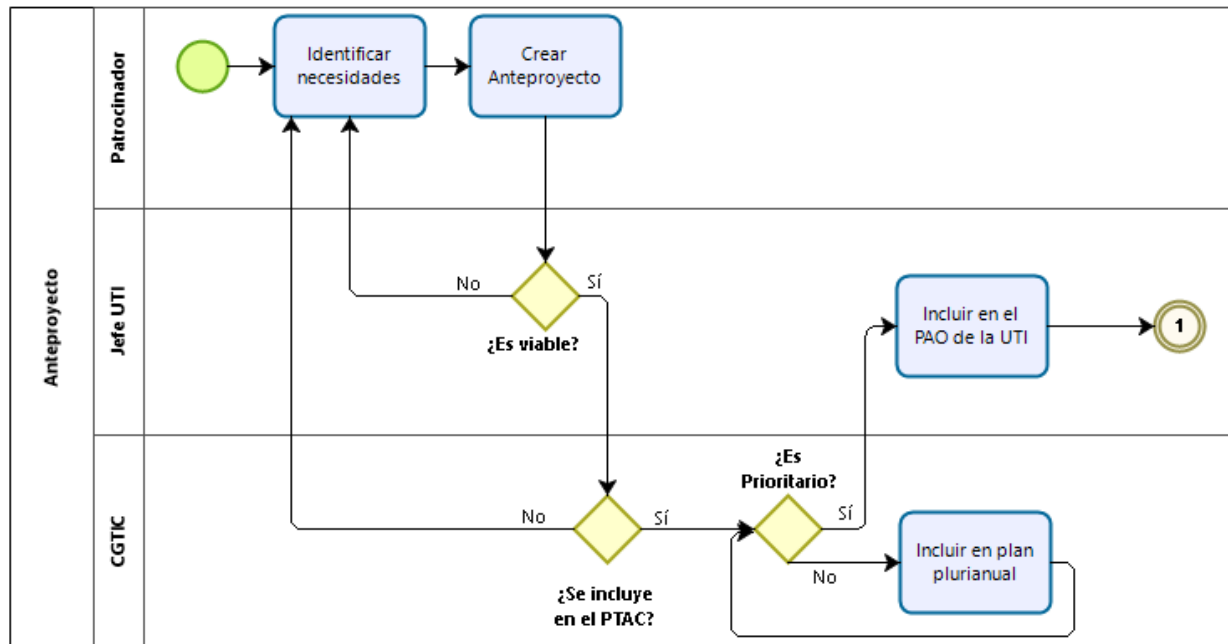


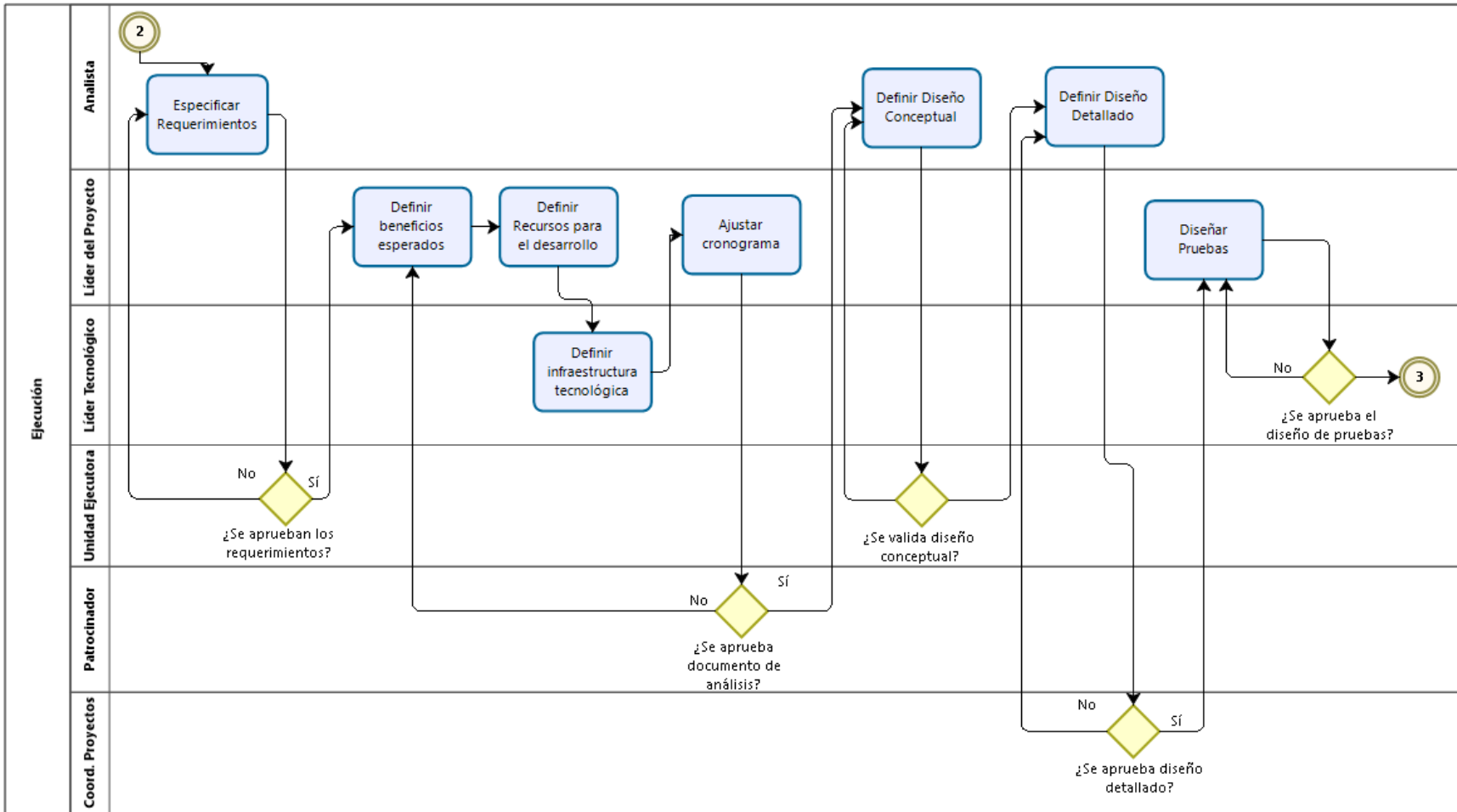
Figura 21: Proceso de implementación

Fuente: Metodología para el Desarrollo de Proyectos de TIC (2016)

- VIII. Evaluación post-implementación: Debe realizarse una evaluación del sistema con el fin de identificar si se cumple con los requerimientos previamente establecidos y en caso de ser necesario realizar los ajustes que se consideren pertinentes. Asimismo, deben discutirse las lecciones aprendidas en el proceso.

En los siguientes diagramas se ilustra el proceso de desarrollo de sistemas vigente con un mayor nivel de detalle.





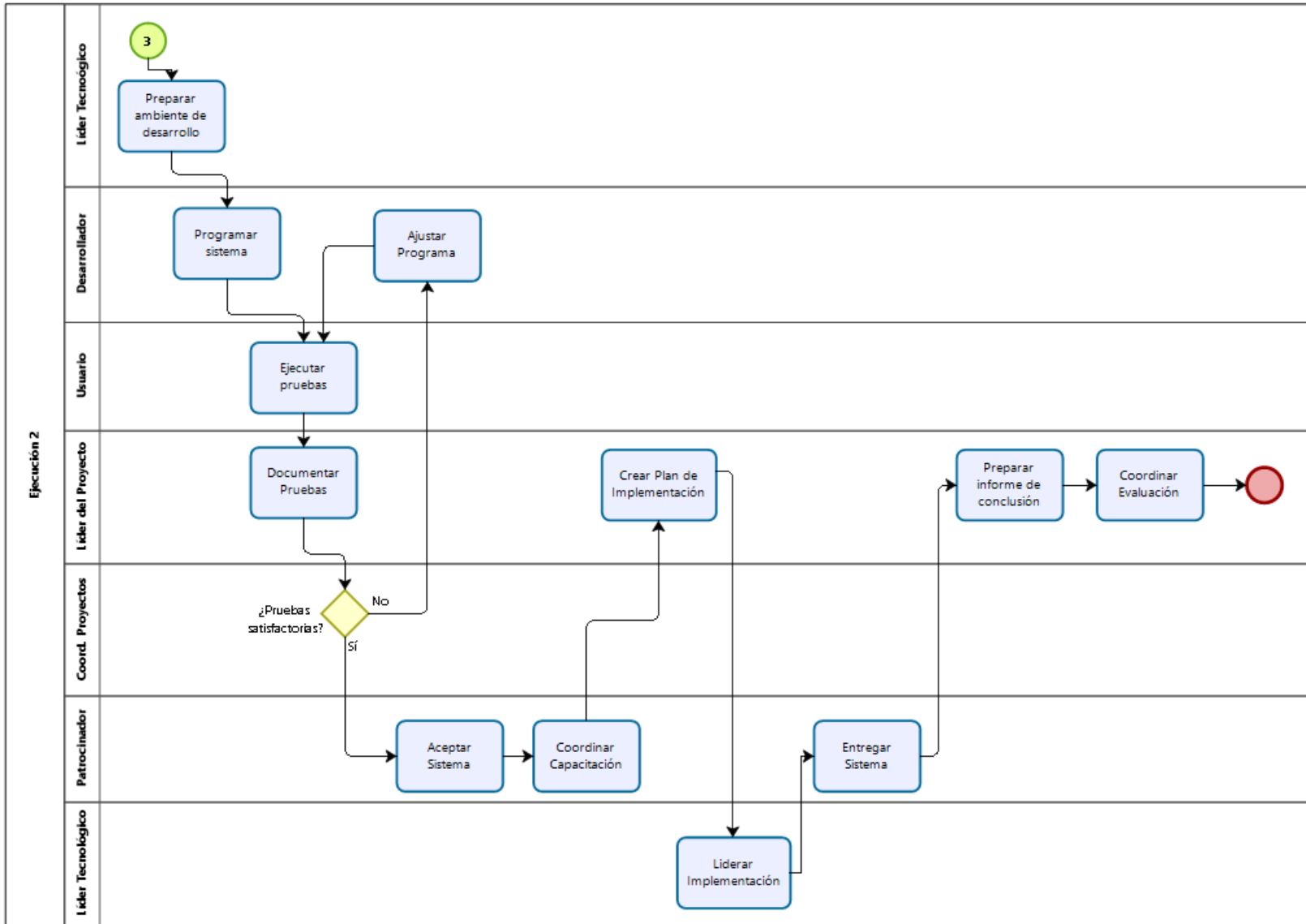


Figura 22: Diagrama del desarrollo de un nuevo sistema

Fuente: Elaboración propia con base en la Metodología para el Desarrollo de Proyectos de TIC (2016)

4.2 Diagnóstico de Percepción

A lo largo de esta sección, se tendrá la percepción de los usuarios de sistemas y del equipo de desarrollo de sistemas, las cuales fueron recabadas por medio de encuestas, entrevistas y revisión de otros estudios en la materia.

Para la percepción de los usuarios del sistema se toman en cuenta evaluaciones de los últimos dos sistemas que se pusieron en producción. El primero corresponde al Sistema de Fiscalización Posterior (Fipnet) y en el Módulo de Aprobación Presupuestaria del Sistema de Información sobre planes y Presupuestos (SIPP).

Con respecto al Fipnet, se tomó en cuenta la evaluación realizada por la Auditoría Interna de la Contraloría General de la República en el informe “I-AIG-09-2018 - Estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones” emitido el 18 de diciembre de 2018. Para la percepción sobre el Módulo de Aprobación Presupuestaria del SIPP, se consideró una encuesta aplicada del 6 al 12 de febrero de 2019.

Por otra parte, se aplicó una encuesta al equipo de desarrollo de sistemas de la UTI, así como una serie de entrevistas tanto al asistente técnico del equipo (coordinador), así como a un desarrollador.

A continuación, se detallan los resultados más relevantes obtenidos con la aplicación de los instrumentos.

4.2.1 Percepción de los usuarios de los sistemas de información

Como se mencionó anteriormente, se estudió la percepción de los usuarios con respecto a los dos últimos sistemas que se desarrollaron y que ya se encuentran en producción.

Para el sistema Fipnet se cuenta con la percepción de 82 usuarios, cuyos resultados más relevantes son los siguientes:

Tabla 10. Resultados más relevantes de la encuesta para medir la satisfacción de los usuarios del Sistema Fipnet

Temática	Valoraciones
Eficacia	Un 79% cree que nunca o pocas veces el sistema le permite alcanzar los objetivos de las labores asignadas.
Satisfacción	El 83.9% considera que nunca o pocas veces el desempeño del sistema es satisfactorio.
Utilidad	El 81.5% considera malo o regular el cumplimiento de los tiempos requeridos en las labores asignadas por medio del sistema.
Capacitación	El 85,2% de los usuarios recibieron capacitación del sistema.
Comodidad	El 83.9% de los usuarios están satisfechos con las herramientas Tecnológicas, siempre o la mayor parte de las veces.
Corrección funcional	El 53.1% cree que pocas veces el sistema proporciona resultados correctos y precisos.
Adecuación funcional	El 88.9% cree que nunca o pocas veces el sistema facilita las labores asignadas.
Comportamiento relativo al tiempo	El 65.4% cree que nunca o pocas veces el tiempo de respuesta y procesamiento del sistema es adecuado.
Facilidad de uso	El 69.1% considera que el sistema no es fácil de usar (facilidad de aprendizaje, operación, y detección de errores).
Facilidad de aprendizaje	El 79% indican que el sistema no tiene tutoriales o manuales que le permitan una mejor comprensión de la funcionalidad.
Estética de la interfaz del usuario	El 50.6% considera que los elementos de estética (colores, diseño) del sistema no son satisfactorios.
Accesibilidad	El 91.4% considera que nunca y pocas veces el sistema tiene elementos de accesibilidad (que facilitan el uso).
Conformidad	El 53.1% considera que la información del sistema no tiene formatos estandarizados para apoyar la calidad de los datos.

Fuente: Estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones (2018)

De los datos contemplados en la tabla anterior, se desprende que hay rubros importantes con más de un 80% de opiniones negativas, lo que denota un problema entre lo que el usuario espera y la solución que se le está ofreciendo.

Dentro de esos rubros negativos destaca la eficacia, dado que un 79% considera que el sistema no le permite alcanzar los objetivos y labores asignadas, asimismo un 84% indica que el sistema no se desempeña satisfactoriamente. Adicional a lo anterior un 89% cree que el sistema no facilita las labores asignadas, y más del 90% considera que el sistema es deficiente en los elementos que facilitan su uso.

En los gráficos que se muestran a continuación, se ilustran los resultados descritos en el párrafo anterior:

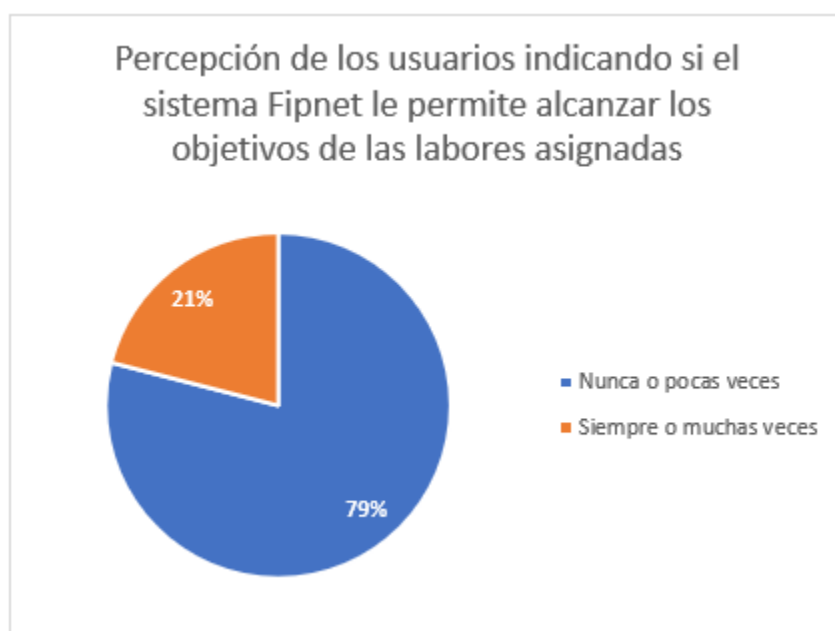


Figura 23: Percepción de los usuarios indicando si el sistema Fipnet le permite alcanzar los objetivos de las labores asignadas
Fuente: Elaboración propia con base en el estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones (2018)

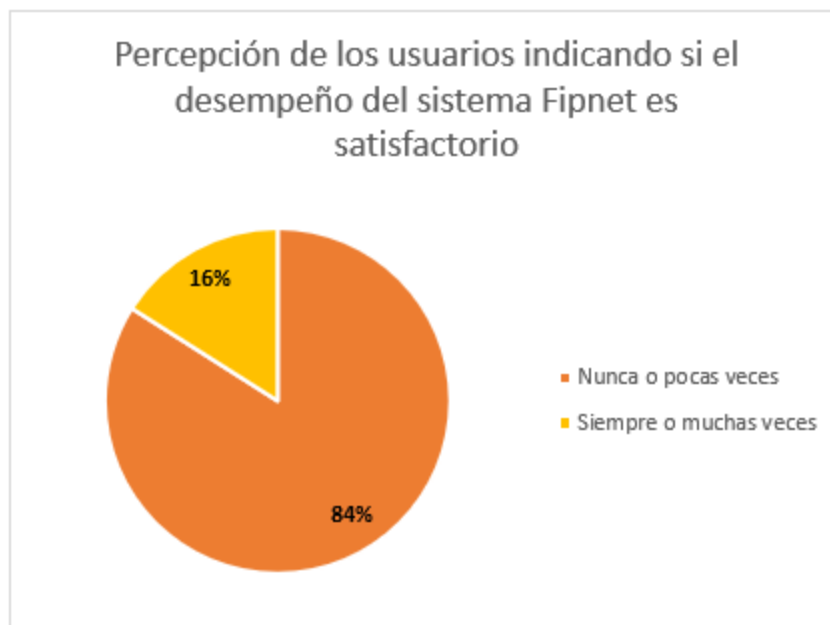


Figura 24: Percepción de los usuarios indicando si el desempeño del sistema Fipnet es satisfactorio

Fuente: Elaboración propia con base en el estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones (2018)

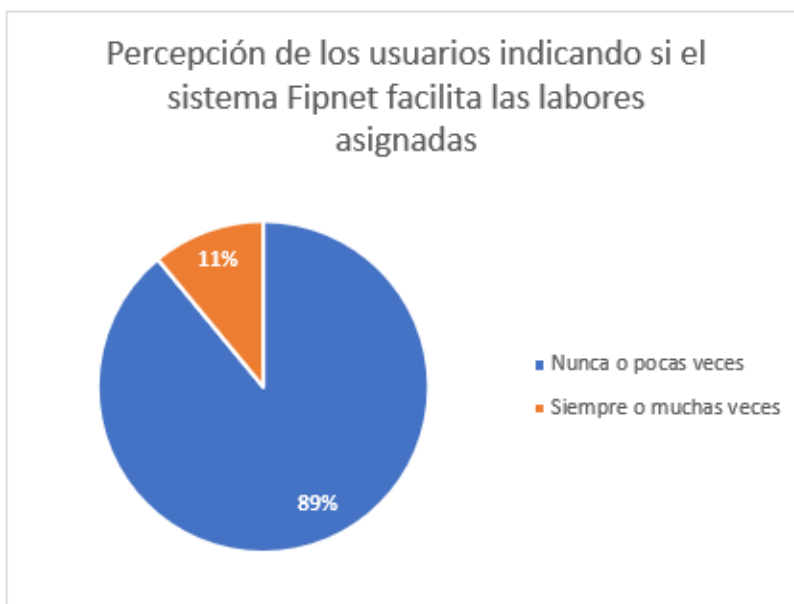


Figura 25: Percepción de los usuarios indicando si el sistema Fipnet facilita las labores asignadas

Fuente: Elaboración propia con base en el estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones (2018)

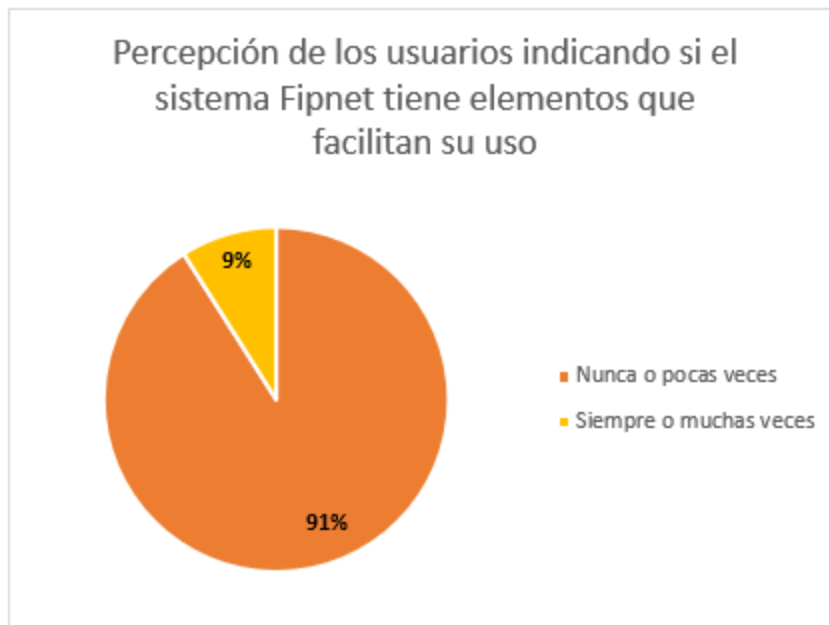


Figura 26: Percepción de los usuarios indicando si el sistema Fipnet tiene elementos que facilitan su uso

Fuente: Elaboración propia con base en el estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones (2018)

Tomando en cuenta estos resultados, se aprecia que no se está entregando el producto de acuerdo con los requisitos o necesidades del negocio, lo que evidencia una desvinculación entre lo que la unidad desarrolladora (UTI) y el representante del negocio (patrocinador) están desarrollando en el proyecto.

Con respecto a la creación de manuales o tutoriales que guíen al usuario en el uso del sistema, también se identifica una deficiencia, ya que un 79% de los usuarios expresaron que el sistema no tiene tutoriales o manuales que le faciliten el uso efectivo del sistema.

Para efectos de evaluar el SIPP, específicamente el módulo de aprobación presupuestaria, se aplicó una encuesta a 53 usuarios. Este nuevo módulo entró en producción en el mes de octubre del 2018.

Como parte de los resultados más relevantes, se encuentra que un 83% considera que la implementación del sistema contribuyó a agilizar el proceso de aprobación presupuestaria. Aparte de lo anterior, un 87% consideran que la navegación dentro del sistema es fácil y un 79% considera que las alertas y ayudas son claras, suficientes y útiles.

En los siguientes gráficos se ilustran los principales resultados sobre la evaluación del SIPP:

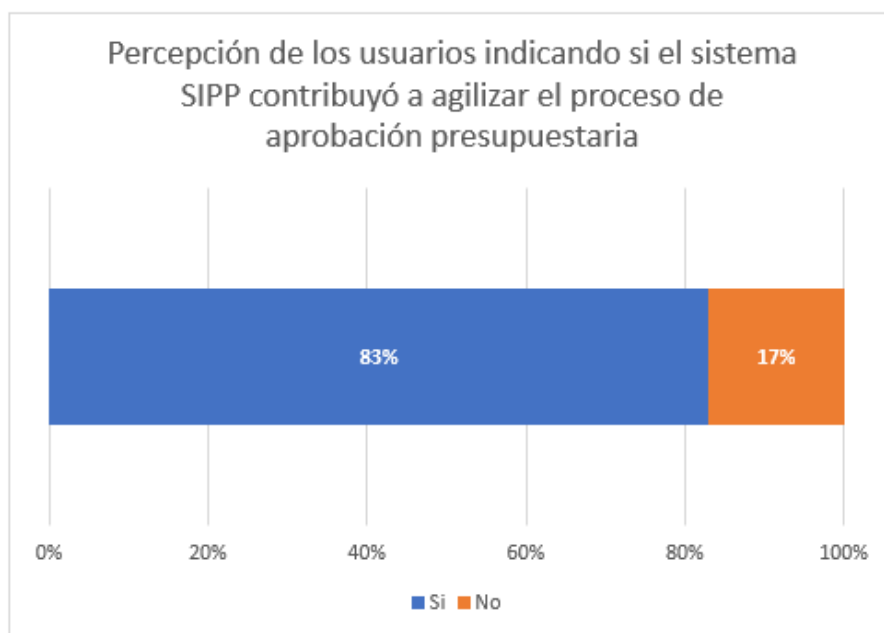


Figura 27: Percepción de los usuarios indicando si el sistema SIPP contribuyó a agilizar el proceso de aprobación presupuestaria
Fuente: Elaboración propia

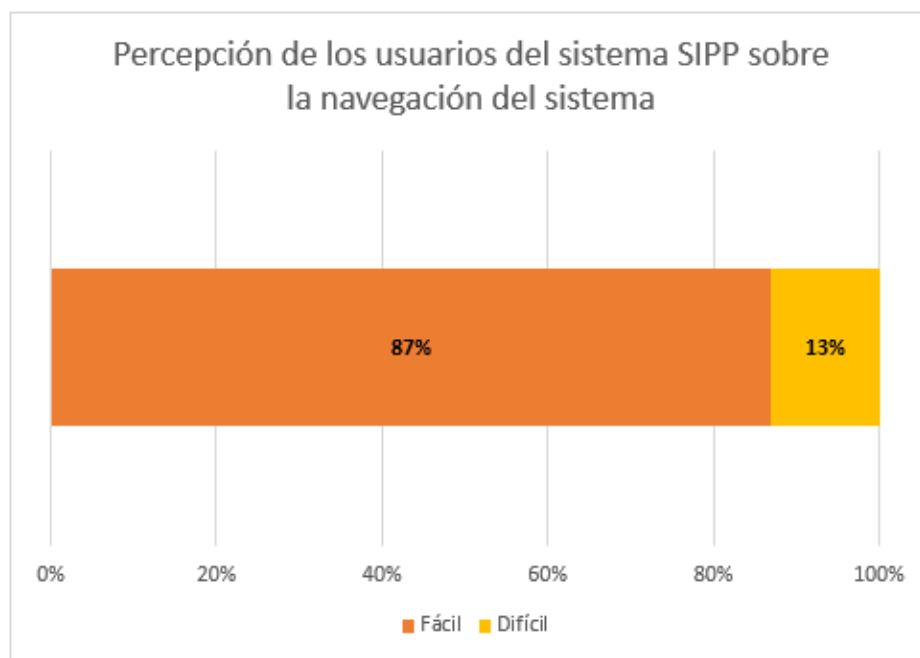


Figura 28: Percepción de los usuarios del sistema SIPP sobre la navegación del sistema
Fuente: Elaboración propia

Estos resultados contrastan con la percepción del FIPNET, lo que evidencia una disparidad en la calidad de los productos generados por la UTI.

Pese a que los resultados generales de percepción del SIPP son positivos, los usuarios señalaron los siguientes puntos de mejora en la encuesta aplicada:

- Existe poca flexibilidad y dificultad para ejecutar algunas revisiones y solicitudes de cambio, por lo que se recomienda rediseñar el proceso en el sistema con el fin de que no sea más difícil que el proceso que se hacía de forma manual.
- Se reciben excesivas alertas al correo electrónico, por lo que podría dosificarse según el tipo de usuario y el rol de responsabilidad que posea.
- Se recomienda que pueda realizarse todo el proceso de aprobación por medio del sistema, ya que actualmente, para firmar en forma digital, los productos finales dentro del sistema, hay que descargarlos y utilizar otra herramienta.
- Se destaca que el proceso de devolución de cuentas mediante el sistema es complejo, por lo que recomiendan facilitar ese proceso.
- Se indica que algunos campos en los que había que digitar, eran muy pequeños, lo que dificultaba la redacción.

Aparte de los puntos anteriores, es importante destacar que un 69% manifestó que los reportes fueron útiles para la aprobación, lo que representa también una oportunidad de mejora, ya que en algunos casos manifestaron que los reportes presentados no se ajustan a lo que realmente necesitan para ejecutar el proceso, y otros usuarios indicaron que, pese a los desarrollados en el nuevo sistema, utilizaron otra herramienta para conseguir los reportes más adaptados a sus necesidades.

Como se evidencia en los resultados de ambas encuestas, es evidente la necesidad de que el usuario final trabaje de la mano con los desarrolladores, desde la concepción de los requerimientos, las pruebas del desarrollo, y la evaluación de resultados finales. Esto coadyuvaría con aumentar la satisfacción de los usuarios con los productos entregados.

Aparte de lo anterior, es importante destacar que podría darse un reforzamiento en los controles, lo que permite evitar el descontento del cliente, la afectación negativa en el cumplimiento de objetivos del proyecto, así como la ineficiencia en los procesos.

4.2.2 Percepción del equipo de desarrollo de sistemas de la UTI

Para medir la percepción del equipo de desarrollo de sistemas de la UTI se les remitió un cuestionario, el cual fue contestado por ocho ingenieros. Se obtuvieron los siguientes resultados:

- Un 38% no conoce con profundidad la metodología de desarrollo vigente.
- Un 75% indica que la metodología vigente no se adapta a las necesidades institucionales y no está acorde con los recursos disponibles.
- Un 50% considera que no se realiza adecuadamente un estudio de factibilidad en el que se contemplen elementos básicos como, recurso humano, infraestructura, costos, tiempo y herramientas disponibles.
- Un 75% considera que no se definen adecuadamente por parte de los usuarios los requerimientos funcionales y no funcionales, mientras que un 88% considera que las técnicas utilizadas para la especificación de los requerimientos no son las adecuadas.
- En relación con el cronograma, un 63% considera que no se documentan todas las tareas por realizar, y un 75% considera que no se cumplen los plazos que se establecen en él.
- Con respecto a las pruebas, un 63% indica que no se especifican los tipos de pruebas por realizar, que no se definen los requerimientos para las pruebas ni los casos y datos necesarios para que el proceso sea exitoso. Asimismo, un 88% indica que los usuarios no cumplen con el tiempo definido en el cronograma para la ejecución de las pruebas.
- En referencia a la gestión de los cambios realizados al alcance del proyecto, un 75% considera que no se hace adecuadamente.
- La metodología vigente establece estándares de nomenclatura, sin embargo, un 50% indica que no los conoce y un 63% no los aplica.
- En un 50% de los casos se indica que no se realiza una sesión con el equipo del proyecto para evaluar la aplicación desarrollada y discutir lecciones aprendidas.

A continuación, se muestran gráficamente los principales resultados obtenidos.

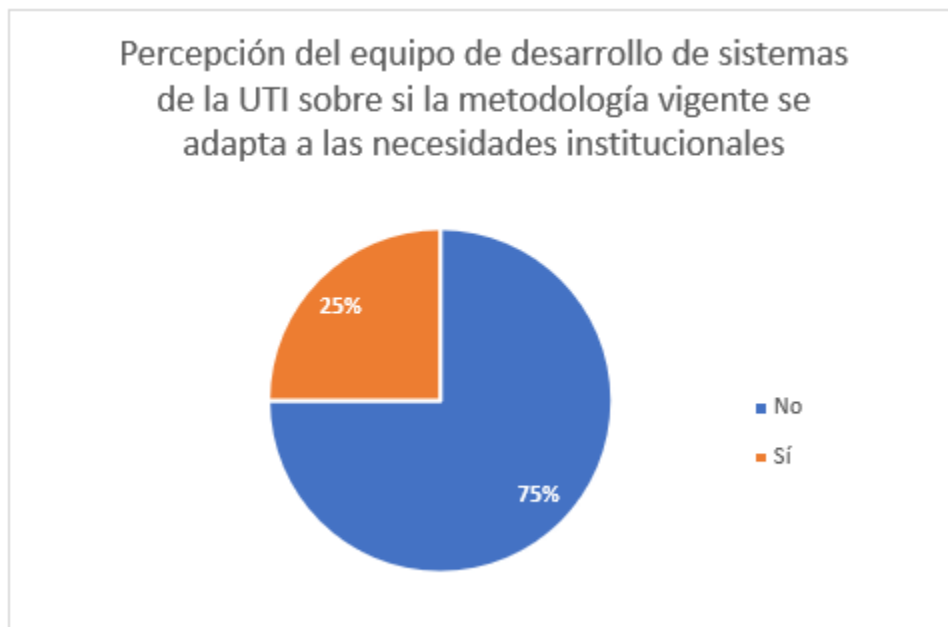


Figura 29: Percepción del equipo de desarrollo de sistemas de la UTI sobre si la metodología vigente se adapta a las necesidades institucionales

Fuente: Elaboración propia

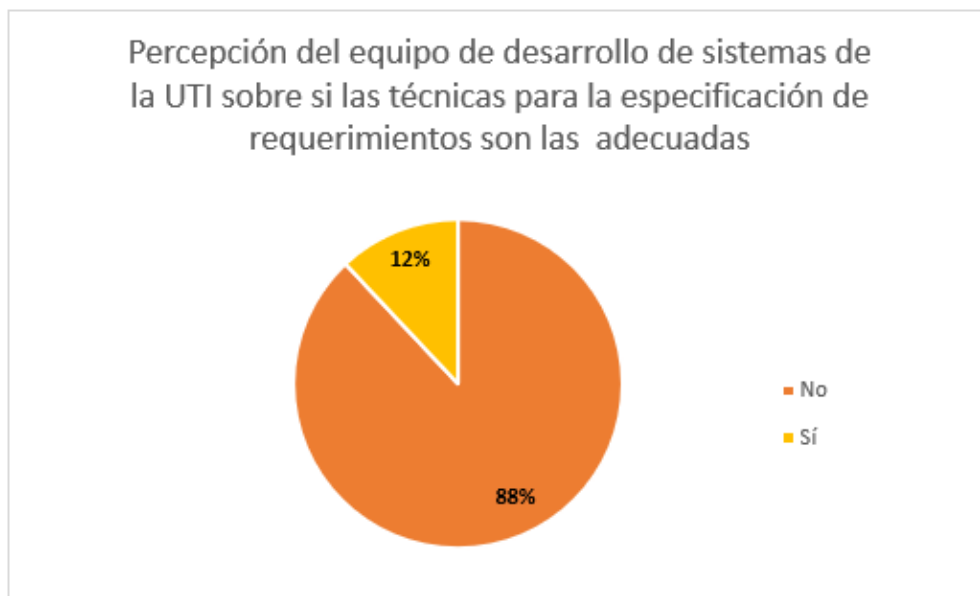


Figura 30: Percepción del equipo de desarrollo de sistemas de la UTI sobre si las técnicas para la especificación de requerimientos son las adecuadas

Fuente: Elaboración propia

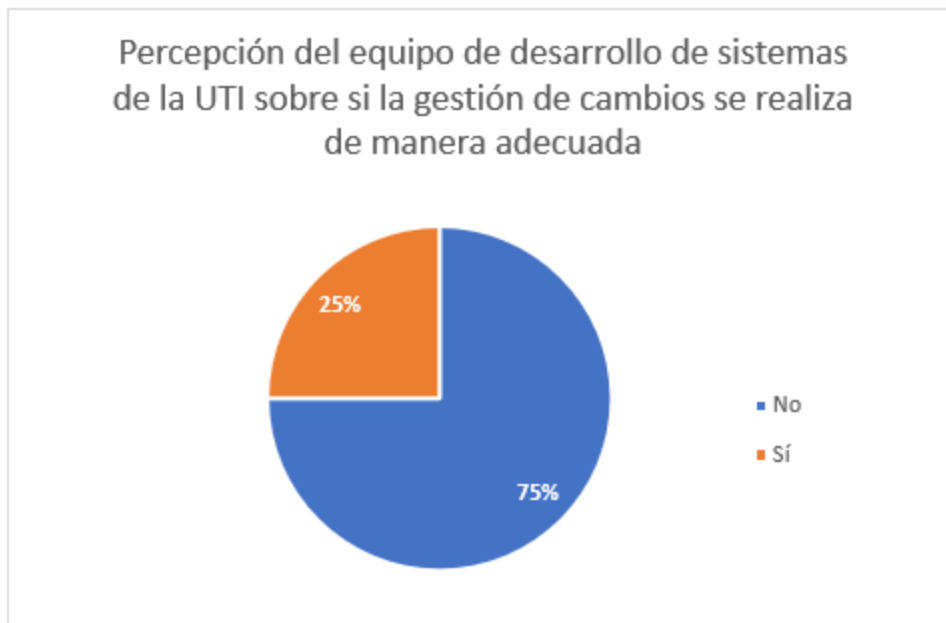


Figura 31: Percepción del equipo de desarrollo de sistemas de la UTI sobre si la gestión de cambios se realiza de manera adecuada
Fuente: Elaboración propia

Al consultarse en una pregunta abierta sobre las principales deficiencias que posee la metodología vigente, los desarrolladores destacaron la obsolescencia de la misma, indicaron que se desarrolló hace mucho tiempo, no se le han aplicado actualizaciones y no se ajusta a las tendencias actuales.

Por otra parte, en los aspectos con resultado positivo, destacan los siguientes:

- Se señaló que se definen adecuadamente los roles del patrocinador del proyecto, líder del proyecto y líder tecnológico con porcentajes iguales o superiores al 88%.
- Se calificó de forma positiva la disponibilidad de los usuarios en caso de que sea necesario aclarar algún requerimiento, un 88% indicó que son atendidos de forma oportuna en caso de dudas.
- Un 63% indica que se cuenta con las herramientas necesarias para el desarrollo de aplicaciones, tales como infraestructura, aplicaciones y capacitación.
- Un 100% señala que el diseño conceptual definido en la metodología es de utilidad para el desarrollo de los sistemas. Situación similar se presenta con respecto al diseño detallado, donde un 83% lo considera relevante para el proceso de desarrollo.

- Para el 75% existe suficiente flexibilidad en el proceso para gestionar cambios en procura de los objetivos.
- Con respecto a la implementación de nuevas aplicaciones, el 100% consideró que se coordina adecuadamente con el líder del proyecto.

Para complementar lo anterior, se muestran los siguientes gráficos con los principales resultados:

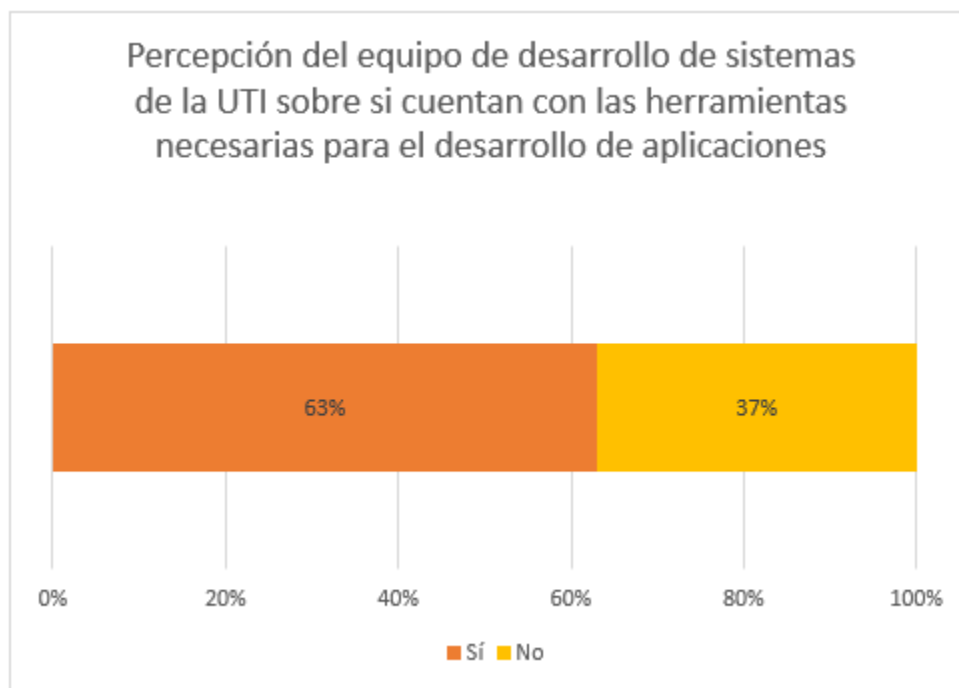


Figura 32: Percepción del equipo de desarrollo de sistemas de la UTI sobre si cuentan con las herramientas necesarias para el desarrollo de aplicaciones

Fuente: Elaboración propia

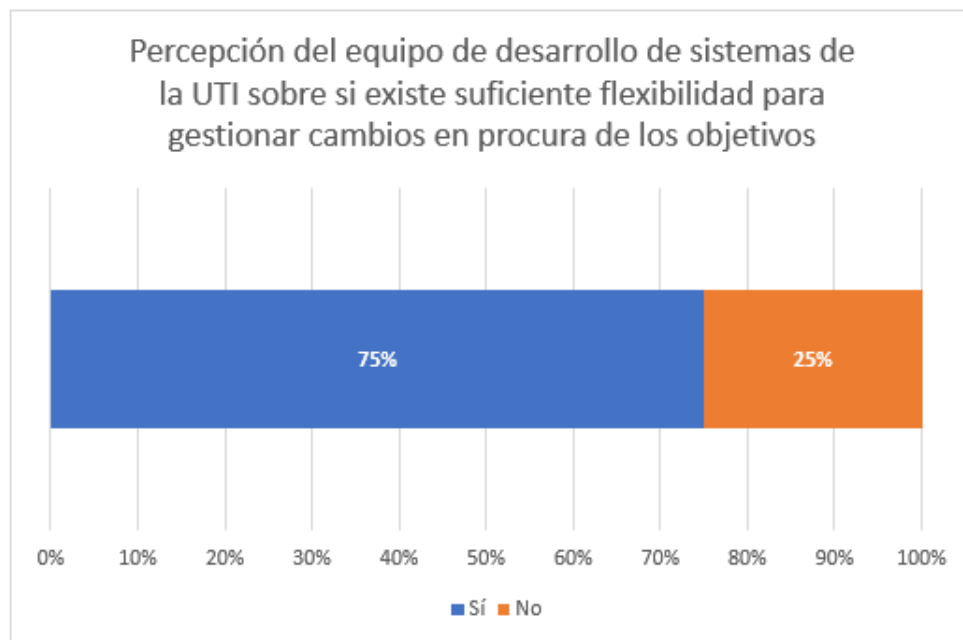


Figura 33: Percepción del equipo de desarrollo de sistemas de la UTI sobre si existe suficiente flexibilidad para gestionar cambios en procura de los objetivos

Fuente: Elaboración propia

Tal y como se aprecia en los resultados, la metodología vigente presenta ciertas deficiencias y carencias que provocan inconformidad en el equipo de desarrollo y que podría estar incidiendo en los resultados. Pese a lo anterior, es necesario destacar que hay algunos documentos de la metodología que son considerados de utilidad para las funciones de desarrollo de sistemas, por lo que no todo es negativo.

Estos resultados son insumos importantes para la propuesta que se desarrolló en el proyecto, con los cuales se busca solventar muchas de las deficiencias señaladas, así como potenciar los puntos positivos que se encontraron.

Aparte de las preguntas de la metodología, se hicieron dos preguntas con miras a lo que se propone en el proyecto. En primera instancia se les consultó a los desarrolladores si tenían conocimiento sobre las principales metodologías ágiles que se utilizan actualmente en la industria del software, un 88% indicaron que las desconocían. Pese a lo anterior el 100% coincidió en que su aplicación podría favorecer el desarrollo de aplicaciones en la institución.

4.3 Brechas o conclusiones del diagnóstico

Una vez hecho un análisis del proceso de desarrollo de sistemas y recopilada la percepción de los usuarios de aplicaciones y del equipo de desarrollo, en esta sección se procede a determinar las conclusiones con base en el diagnóstico realizado.

Tomando en consideración tanto lo analizado en el diagnóstico administrativo como la percepción del equipo de desarrollo de sistemas, resulta evidente que la metodología vigente no ha sido actualizada a las mejores prácticas de desarrollo de sistemas aplicadas en la industria, ya que muestran obsolescencia en muchas de las tareas contempladas en el proceso. Por lo tanto, es necesario actualizar dicha metodología con base en las mejores prácticas vigentes, adaptándola a las necesidades de la institución.

Adicional a lo anterior, la documentación que se solicita en algunas de las fases del proceso es muy exhaustiva y muestra algunas duplicidades con lo que se solicita en otras etapas. Por consiguiente, es necesario realizar una revisión de qué se debe documentar, tomando en consideración lo indicado por el equipo de desarrollo, quienes afirman que tanto el documento de diseño conceptual y de diseño detallado son de utilidad para el desarrollo del sistema.

La actualización que se proponga a la metodología, debe considerar además una mejora en la valoración que se realice antes de iniciar un proyecto, tomando en consideración elementos básicos como la gestión adecuada del tiempo de los colaboradores, infraestructura, costos, tiempo estimado, herramientas necesarias. Asimismo, debe mantener la valoración del alineamiento del proyecto propuesto con el plan estratégico de la institución.

Aparte de lo descrito anteriormente, otro de los factores críticos que destaca en el diagnóstico, es el proceso de definición de requerimientos. La percepción del equipo de desarrollo sobre el proceso y las técnicas utilizadas, así como el descontento de los usuarios con los productos que reciben, hacen ver que existe una deficiencia en dicha fase. Por lo tanto, en la propuesta por desarrollar, este debe ser uno de los puntos que más deben fortalecerse, procurando un pleno entendimiento de las necesidades del usuario y el producto que pueda desarrollarse, con empleo de técnicas de vanguardia utilizadas en la industria.

Una situación similar a la descrita en el punto anterior es la referente a la ejecución de pruebas, para la que se denota un problema crítico, ya que no se especifican los tipos de prueba

por realizar, los requerimientos que deben contemplarse en el ambiente de “testing”, ni los casos que deben considerar el usuario a la hora de llevar a cabo el plan de pruebas. Esto, sin duda, repercute en la calidad del producto que se ofrece, debido a que aumenta el riesgo de errores en la operación del sistema, así como el descontento de los usuarios, por lo que la propuesta de metodología debe buscar una mejor coordinación entre el equipo técnico y los usuarios, con el fin de solventar esta situación, así se lograría un producto mucho más depurado, que satisfaga de mejor manera las necesidades identificadas.

En relación con la organización del tiempo del proyecto, la propuesta debe considerar también una mejora importante, debido a que es usual que se presenten retrasos con los plazos establecidos en el cronograma, según lo indicado en el diagnóstico de percepción.

Otro de los aspectos que debe considerarse, es la gestión de cambios que se realizan en el alcance de los proyectos, ya que, de acuerdo con la percepción, no se realiza de forma adecuada. Para esta situación, deben contrastarse las metodologías ágiles, que ofrecen mayor flexibilidad, con las necesidades de la institución a la hora de definir un proyecto con un objetivo definido desde el inicio.

La propuesta de metodología debe contener un estándar con la nomenclatura utilizada tanto en los objetos de la base de datos como en el código de los programas. De acuerdo con los resultados del diagnóstico de percepción, la mayoría del equipo de desarrollo desconoce y omite el uso del estándar definido en la metodología, lo que podría dificultar el entendimiento del código para los otros programadores del equipo, lo que provocaría atrasos en la atención de incidentes y dificultades en cambios que deban realizarse posteriormente.

Pese a que la metodología vigente contempla una fase para evaluar los resultados al finalizar los proyectos y discutir las lecciones aprendidas, no se detalla algún instrumento o método para aplicar en dicha etapa del proyecto. Asimismo, no es una práctica constante, ya que la mitad del equipo de desarrollo reconoció que no la ejecuta. Dado lo anterior, en la propuesta debe contemplarse una mejora en este proceso, dada la importancia que tiene para la calidad del producto y para la mejora continua en la institución.

Por último, y según se denota en lo manifestado por el equipo de desarrollo, hay un desconocimiento en las metodologías ágiles de desarrollo de sistemas que se utilizan actualmente

en la industria, sin embargo, la totalidad de los encuestados coinciden en que la aplicación de este tipo de metodologías sería favorable en el proceso. Por lo tanto, para la propuesta por realizar, debe valorarse el uso de estas metodologías, tomando en consideración como podrían aportar a cerrar las brechas definidas anteriormente en procura de una mejora considerable en el proceso de desarrollo de sistemas. Asimismo, debe capacitarse al equipo de desarrollo en las últimas tendencias, con el fin de que puedan tener nuevas herramientas y mejores formas de trabajo para incrementar la calidad y oportunidad de los productos que ofrecen.

CAPÍTULO V: PROPUESTA DE PROYECTO

En el presente capítulo se realiza una propuesta de mejora a la metodología de desarrollo de sistemas de la Contraloría General de la República, buscando que responda a las necesidades actuales de la institución. Con este fin, se hace un análisis de la metodología vigente, tomando como base los resultados del diagnóstico realizado en el capítulo anterior.

Asimismo, para fundamentar la propuesta, se ejecuta un análisis comparativo entre las principales metodologías vigentes en la industria de desarrollo de software, buscando que el producto creado tome en cuenta las mejores prácticas utilizadas actualmente.

Una vez realizado lo anterior, se define la propuesta de metodología, buscando solventar las necesidades actuales de la institución, específicamente de los actores involucrados en el proceso y de los principales clientes.

5.1 Análisis de la metodología de desarrollo de software

La metodología vigente para el desarrollo de sistemas de información entró en vigencia en julio del año 2009, luego sufrió una ligera actualización en el año 2016.

Como se mencionó en el capítulo anterior, dicha metodología se elaboró en atención a lo dispuesto en las Normas Técnicas para la Gestión y el Control de las Tecnologías de Información (N-2-2007-CO-DFOE) en las que se solicita a las instituciones públicas que establezcan un Marco de Referencia para los procesos relativos a la gestión de TI.

Con base en los puntos recabados en el diagnóstico, es importante enfocar el presente análisis en una serie de factores relevantes, los cuales deben ser solventados en la propuesta que se realizará más adelante.

La metodología vigente tiene una década desde su entrada en vigencia, y dadas las prácticas y modelo de desarrollo contemplados (cascada), ya se encuentra en obsolescencia. Tal y como fue señalado en el diagnóstico, esta metodología no se adapta a las necesidades actuales de la institución ni está acorde con los recursos disponibles.

Uno de los principales objetivos de la institución, definidos en su Plan Estratégico 2013-2020, hace referencia a la transformación del proceso de fiscalización integral, razón de ser de la

CGR, con el fin de que sea sustentado en las tecnologías de información con el fin de incrementar su confiabilidad y oportunidad. Debido a lo anterior, la metodología debe responder de forma eficiente y ágil a los principales desafíos que enfrenta la institución en su papel de fiscalizador de la Hacienda Pública, puesto que la gestión de TI y el uso de sistemas de información es cada vez más relevante para potenciar las tareas que se realizan actualmente.

Pese a lo anterior, por los controles que actualmente imperan en el aparato estatal en el que se encuentra inmersa la CGR, así como la necesidad de formalización de varios aspectos, no es posible dejar de lado la documentación de algunos procesos, por lo que la implementación de una metodología ágil en su totalidad se torna difícil. No obstante, si es posible incluir los principios más relevantes, los cuales harían más eficiente el proceso y, sobre todo, se buscaría una mayor satisfacción del cliente por medio de la entrega pronta y continua de los sistemas de información, que satisfagan las necesidades que dieron origen al proyecto, el cual es un factor que no se está cumpliendo como debería, según lo obtenido en el análisis de la percepción del capítulo anterior.

Otro punto crítico que debe considerarse es el relacionado con el estudio de factibilidad que se realiza, para tomar la decisión de asumir un proyecto de desarrollo de un sistema de información. Actualmente, se hace la valoración inicial en el CGTIC tomando en consideración solamente el resultado esperado, efecto, objetivos, productos, enfoque, alcance, recurso humano de la unidad patrocinadora y prioridad. Posteriormente, en la etapa de iniciación, se solicita un informe de diagnóstico que contenga la situación actual, alcances del proyecto, objetivos, valoración de opciones de solución y recomendación sobre la opción óptima y un análisis de riesgos del proyecto.

Con respecto al estudio de factibilidad, para que el CGTIC pueda tomar una mejor decisión de sí el proyecto debe incluirse en los respectivos planes de trabajo, debería contar con un análisis de factibilidad del proyecto como insumo principal. Por lo tanto, es una de las mejoras que se harán en la propuesta.

Debe hacerse también una valoración del proceso de recolección de requerimientos, dado que, cómo se determinó en el diagnóstico, un alto porcentaje considera que no se definen adecuadamente ni se utilizan las mejores técnicas para ello. Asimismo, en la percepción de los usuarios se evidencia un descontento con algunos productos entregados.

Para dicho problema, la metodología propone una descripción narrativa del requerimiento, un prototipo de interfaz de usuario y casos de uso. De acuerdo con la experiencia descrita por los ejecutores del proceso, la más usada es la descripción narrativa, sin embargo, en algunas ocasiones, no son definidas en conjunto por la unidad patrocinadora y el analista de la UTI, por lo que no se da un entendimiento pleno entre actores. Para solventar dicha situación, se aplicarán principios de metodologías ágiles con las que se da un trabajo en conjunto entre el analista y el usuario para una definición más certera de las necesidades que el sistema de información debe solventar.

Por otra parte, con respecto a la programación, se abarcarán dos aspectos importantes. El primero relacionado a la actualización del estándar de programación, dado que, como se reflejó en el diagnóstico, un porcentaje alto de desarrolladores no utilizan el propuesto por la metodología, el cual no abarca la totalidad de aspectos donde sería necesario aplicarlo, lo que podría estar generando que se ignore. Sin embargo, el cambio más importante va a ser el abandono del modelo de cascada, por un proceso iterativo con base en metodologías ágiles, lo que impactará también la definición de tareas y el cronograma.

En relación con el plan de pruebas, debe corregirse la omisión e incumplimiento de plazos que se presentan con la aplicación de la metodología actual. De igual manera, este aspecto va a sufrir variaciones, debido a que las pruebas se ejecutarán con cada iteración, según los principios de metodologías ágiles que se propongan. Si es importante tener en cuenta los requerimientos que se deben probar, así como habilitar el ambiente de pruebas y los usuarios que se utilizarán. En este aspecto, y con la aplicación de los principios de metodologías ágiles, va a ser imprescindible una colaboración directa del usuario con el programador, así como el cumplimiento de los plazos con el fin de no atrasar las siguientes iteraciones.

Pese a que va a ser necesario que se defina un alcance al iniciar el proyecto, la gestión de cambios va a ser mucho más flexible al aplicar principios de metodologías ágiles durante la etapa de desarrollo, dado que, al darse iteraciones, se reduce el riesgo de cambios complejos y reprogramaciones en relación con un sistema desarrollado en cascada. Además, se hará un cambio en la documentación que hay que preparar con el fin de reducir la dificultad del trámite de cambios que se da actualmente en la metodología, en la que debe pasar por la aprobación de la Unidad Ejecutora.

Además, en relación con las sesiones de evaluación, la metodología solo establece que debe realizarse una sesión con el equipo del proyecto para discutir lecciones aprendidas. Sin embargo, según lo identificado en el diagnóstico, en la mitad de los casos estas sesiones no se dan. Por lo tanto, en la propuesta se hará una mejora, con el fin de implementar un instrumento de evaluación estándar para todos los proyectos.

5.2 Análisis comparativo entre metodologías utilizadas actualmente en desarrollo de software.

Con el fin de que la propuesta que se desarrolla en este capítulo esté actualizada con las metodologías más utilizadas en la industria del software, se hace un análisis comparativo que permita definir mejores prácticas que puedan ayudar con el objetivo de desarrollar una propuesta que responda a las necesidades actuales de la organización.

Para ello, se presenta el siguiente cuadro comparativo en el que se visualizan características propias de cada metodología, con sus respectivas ventajas.

Tabla 11: Comparación entre metodologías Cascada, Programación Extrema (XP) y Scrum

Variable		Cascada	XP	Scrum
Tipo de proceso		Lineal con etapas secuenciales	Ciclo Iterativo	Ciclo Iterativo
Duración promedio de iteraciones		No aplica	Una o dos semanas	Tres semanas
Jerarquía en Equipo		Estructurada	Equipo autogestionado	Equipo autogestionado
Trabajo en equipo		No es continuo	Continuo	Continuo

Participación del cliente	Baja	Alta	Alta
Planificación	Al inicio del proyecto	Se planifica durante todo el proceso	Se planifica durante todo el proceso
Flexibilidad para cambios	Baja	Alta	Alta
Entregables	Producto final	Cada una o dos semanas	Cada tres o cuatro semanas
Enfoque	Se planifica y documenta en todas las áreas	Se tienen un enfoque general sobre el que se trabaja y se va planificando	En su programación se incluyen aspectos de ingeniería y se define cómo el equipo debe definir el producto.
Ejecución de tareas	Según se defina por el equipo	Orden estricto priorizado por el cliente, sin embargo, en el proceso se permiten cambios dentro de una iteración.	Orden acordado por el equipo, pero no se permiten cambios hasta finalizar la iteración
Dificultad para medir el progreso	Baja	Media	Media
Enfoque en calidad	Alta	Media	Media
Gestión de riesgo	Alta	Baja	Baja

Fuente: Elaboración propia

Tomando en consideración la teoría de las necesidades de la institución para el desarrollo de *software*, la teoría de las distintas metodologías descrita en el marco teórico, las oportunidades de mejora identificadas del proceso identificadas en el diagnóstico, el análisis de la metodología y el cuadro comparativo anterior, se puede concluir que tanto las metodologías ágiles como la metodología cascada tienen características particulares que podrían aportar insumos importantes para la propuesta de metodología.

Aparte de lo anterior, las características de la institución, al ser parte del sector público costarricense, podrían dificultar la implementación de una metodología ágil en su totalidad, dado que deben rendirse cuentas por los recursos públicos que se utilizan, tener una adecuada gestión de riesgos, y establecer planes anuales con sus respectivos proyectos.

No obstante lo indicado, las metodologías ágiles poseen una serie de características que podrían implementarse en la propuesta, inclusive las dos metodologías estudiadas en este proyecto, Scrum y Programación extrema, podrían complementarse entre sí y aportar insumos importantes para la actualización de la metodología. Scrum incluyendo los aportes de la filosofía ágil, y la programación extrema, aparte de los principios ágiles, podría aportar una serie de aspectos técnicos para el desarrollo de aplicaciones que Scrum no contempla.

Dado lo anterior, la nueva propuesta que se expone a continuación será una metodología híbrida, en la que se incluyen aspectos de metodologías ágiles con el fin de brindar celeridad, mayor satisfacción del cliente y flexibilidad a la metodología vigente, que está estructurada en un modelo de cascada. Asimismo, contendrá aspectos de metodologías tradicionales en la gestión de otros aspectos del proyecto que no están directamente relacionados con la etapa de programación.

Realizando un análisis comparativo en la metodología de desarrollo en el sector público de diversos países, se encontró que se han empezado a utilizar metodologías híbridas con el fin de ir incluyendo gradualmente principios ágiles en el aparato estatal.

Un ejemplo de lo anterior fue un artículo publicado en el sitio DigitalGov, el cual es una página oficial del gobierno de los Estados Unidos, donde el autor cita lo siguiente:

“La lección que aprendimos con el tiempo fue que forzar un marco ágil en un paradigma de cascada existente para "mejorar nuestra organización" no hará mejoras inmediatas a la organización. En su lugar, el marco ágil debe estar sucediendo en todas las facetas de

nuestra organización y las prácticas de las metodologías cascada y scrum pueden combinarse para que la organización, en general, se vuelva gradualmente más ágil. A través de ajustes iterativos a la planificación, podemos asegurar que se cumplan las expectativas y que todas las prioridades estén documentadas de manera efectiva; a través de actividades de desarrollo iterativo, podemos garantizar que los elementos de mayor prioridad se trabajen y se entreguen como se espera; a través de entregas iterativas, podemos garantizar que los proyectos se entreguen con los objetivos previstos y examinados exhaustivamente por los clientes”. (Stauffer, A., 2018)

Por otra parte, en la Administración Pública de España, las metodologías híbridas, en las que se recogen las bondades y beneficios de las metodologías tradicionales y ágiles, también vienen cobrando relevancia, según se puede apreciar a continuación:

“Tradicionalmente, en el sector público español, la mayoría de los proyectos de desarrollo de SW se han llevado a cabo bajo metodología tradicional, y aunque la adopción de metodologías ágiles, tipo Scrum, se está incrementando de manera notable, su implantación ha sido más lenta, en comparación con el sector privado.

La naturaleza y cultura propia de las grandes organizaciones, junto con la complejidad organizativa y procedimental de las Administraciones Públicas ha generado que la adopción técnica y organizativa de este tipo de metodologías no sea un proceso precisamente ágil, y que debemos plantearnos escenarios a medio plazo, comenzando por pilotos acotados, cuyo éxito permitirá extender la misma filosofía al resto de la organización.

Aun siguiendo esta estrategia de implantación paulatina, nos encontramos con dificultades que una implantación “pura” de metodología ágil no es capaz de sortear en ciertas organizaciones, o al menos en un tiempo razonable (posibilidad de dedicación del personal del organismo, procesos del ciclo de vida del SW que no confiesan con la filosofía ágil, etc.). Por ello la implantación de metodologías híbridas, como solución temporal o definitiva, se está erigiendo en una alternativa muy adecuada para ciertas organizaciones.” (Cruz, N. & González, A., 2018)

Tal y como se aprecia en los dos casos anteriores, dado que en el sector público ha sido difícil que se adopten metodologías ágiles, las metodologías híbridas están surgiendo como una buena alternativa para ir dando ese salto paulatino hacia las metodologías de desarrollo ágiles.

5.3 Propuesta de mejora a la metodología de desarrollo de sistemas de la información de la Contraloría General de la República

Con base en el análisis de la metodología vigente y en la investigación de las mejores prácticas y metodologías usadas en la industria del desarrollo de software, en esta sección se propone una actualización a la metodología vigente, incorporando elementos de metodologías ágiles en algunas etapas.

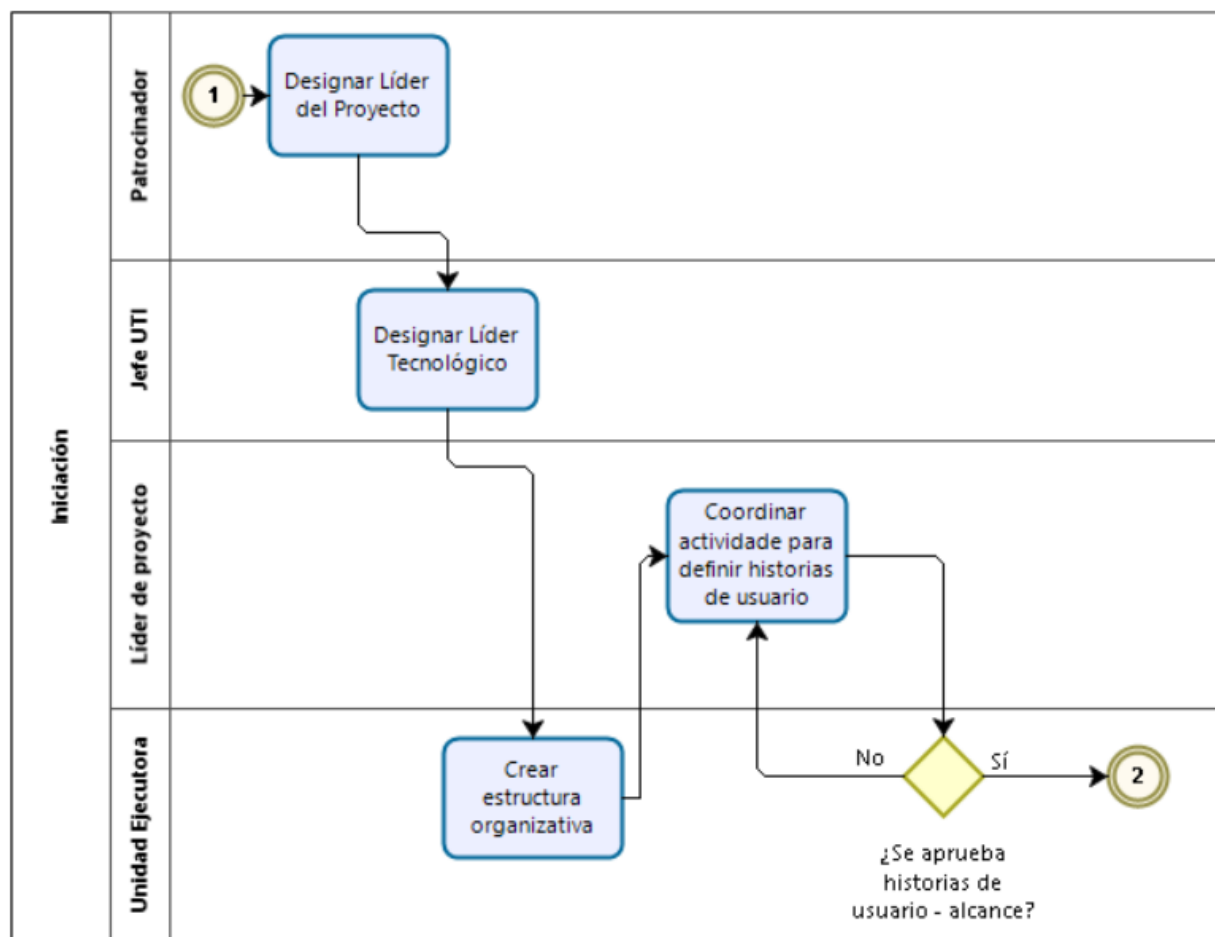
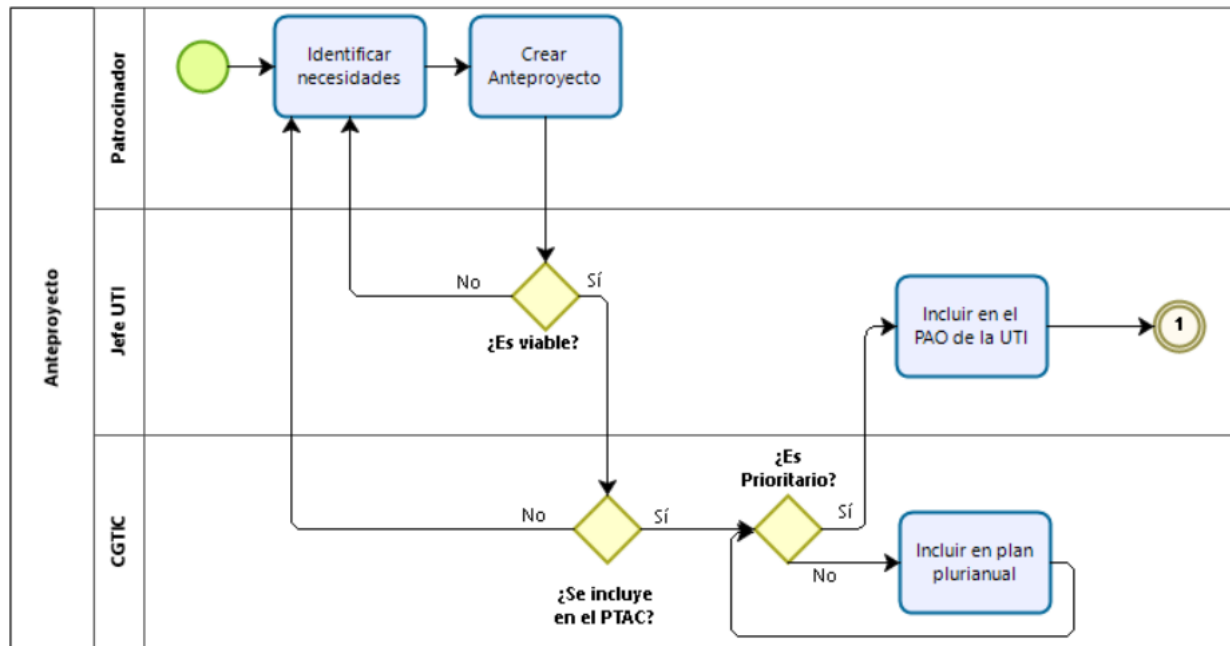
Al desarrollar esta propuesta, se busca dotar a la UTI de procesos más eficientes y ágiles, para solventar, de una mejor forma, las necesidades de sus clientes con respecto a sistemas de información.

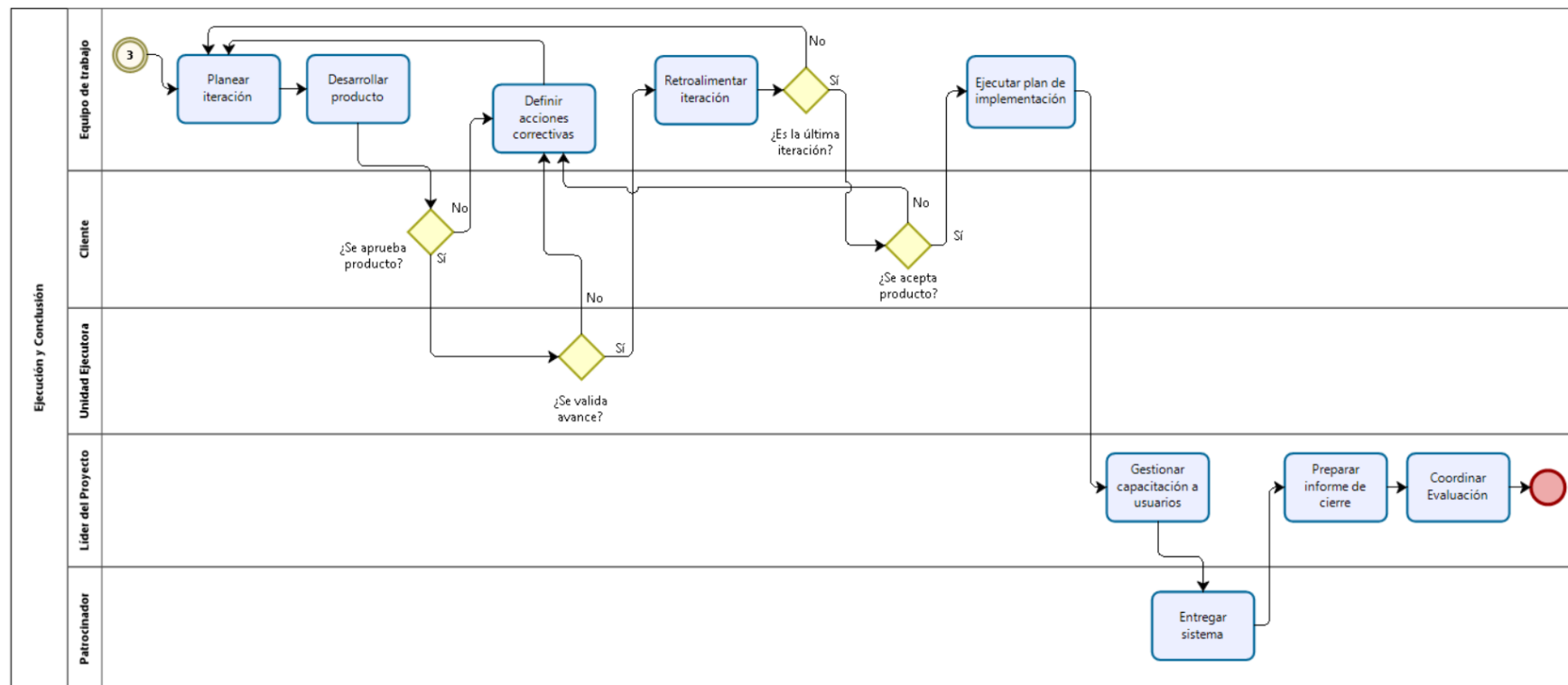
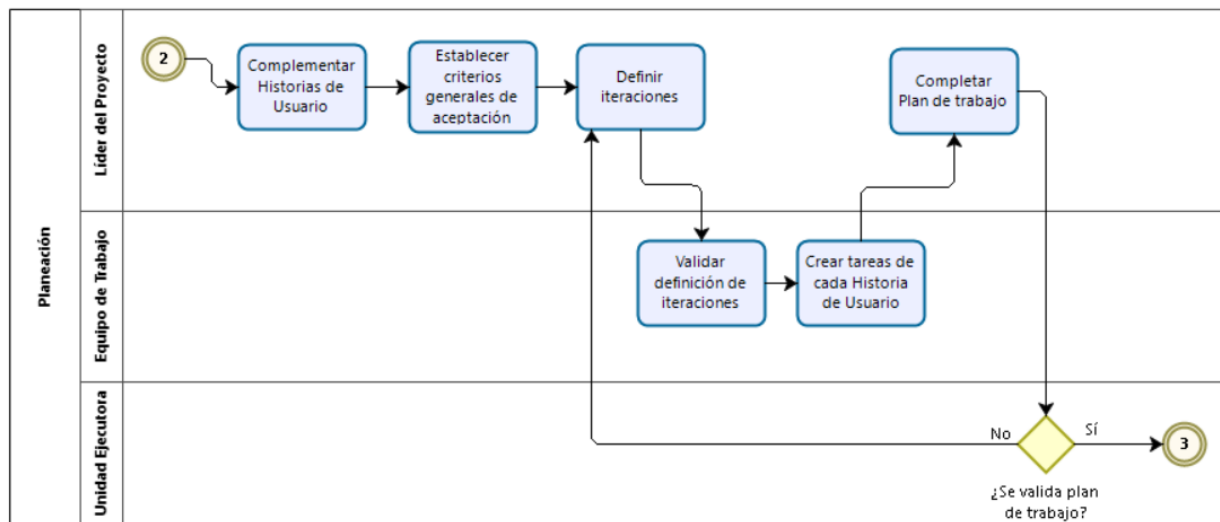
Adicional a lo anterior, se busca potenciar las oportunidades de mejora identificadas en los diagnósticos realizados, en los que destacan; la insatisfacción de los clientes con algunos productos, la deficiencia en la definición de requerimientos, carencias en la elaboración y ejecución del plan de pruebas, deficiencia en gestión de cambios, ampliación del estándar de nomenclatura, evaluación del resultado en la implementación de proyectos, entre otras.

A continuación, se presenta la actualización del diagrama del proceso de desarrollo así como la propuesta de actualización de la metodología de desarrollo.

5.3.1 Diagramas del proceso de desarrollo de software propuesto

En los siguientes diagramas, se ilustra el proceso de desarrollo propuesto. En esta metodología se desarrolla cada una de las tareas ahí contempladas.





5.3.2 Metodología propuesta

Propuesta de Metodología para el desarrollo de sistemas de información en la Contraloría General de la República

I. ASPECTOS GENERALES

A. Introducción

Con el fin de establecer un lenguaje común a nivel institucional; así como un estándar en la ejecución y documentación de proyectos de desarrollo de sistemas, se define en este documento la metodología para el desarrollo de sistemas de información para la Contraloría General de la República.

La presente guía metodológica, se elabora no sólo para efectos de estandarización, sino también en cumplimiento con lo establecido en el Manual de Normas Técnicas para la Gestión y el Control de las Tecnologías de Información, emitidos por la CGR.

En esta metodología se define una serie de pasos comunes por seguir, con el fin de mejorar las probabilidades de éxito de los proyectos de desarrollo de sistemas, teniendo siempre presente que también son aspectos importantes el nivel de motivación y mística de que estén impregnados en los integrantes del equipo de trabajo, de la disponibilidad de recursos y del nivel de apoyo que brinde oportunamente la alta Gerencia.

Con los sistemas que se desarrollen se introducirán en la organización, elementos tecnológicos que soporten y hagan más eficiente la ejecución o el desarrollo de un proceso. Todo sistema que se desarrolle siempre deberá estar orientado al logro de los objetivos institucionales y aportar un valor agregado a la organización, respondiendo a sus necesidades de manejo de la información y del conocimiento.

Es importante destacar que, pese a que en la presente metodología se incluyen prácticas de vanguardia utilizadas en la industria de desarrollo de sistemas, es importante que se mantenga en constante actualización dado el ritmo acelerado con el que cambia el entorno tecnológico.

B. Objetivo General:

- Definir la metodología para el desarrollo de sistemas de información, con el fin de satisfacer de la mejor forma las necesidades de la institución.

C. Objetivos Específicos:

- Definir y uniformar etapas, técnicas y métodos de trabajo en el proceso de desarrollo de sistemas de información.
- Establecer las etapas del proceso de desarrollo de sistemas de información con sus respectivas actividades y entregables.
- Señalar la organización, las funciones y las responsabilidades de los involucrados en el proceso de desarrollo de sistemas de información.

D. Etapas de un proyecto de desarrollo de sistemas de información

A continuación, se presentan las etapas de un proyecto de desarrollo de sistemas, con un breve detalle de los objetivos de cada una de ellas:

Etapas 0: Anteproyecto

- En la organización se identifica una serie de necesidades que pueden ser atendidas mediante el desarrollo de un sistema.
- Se determinan las expectativas generales de los interesados, así como el efecto y resultados esperados.
- Se identifican los actores involucrados en el proyecto por desarrollar.
- Se confecciona la ficha de anteproyecto.
- Se somete el anteproyecto a la evaluación del Comité Gerencial de Tecnologías de Información y Comunicación (CGTIC), el cual valorará su viabilidad y prioridad dentro de la organización.

Etapa 1: Iniciación

- Corroborar las expectativas generales de los usuarios, gerentes y de cualquier otro interesado, para establecer los resultados esperados y el enfoque del proyecto.
- Definir la organización del proyecto y seleccionar el equipo de trabajo.
- Describir el producto que se espera obtener con el proyecto, indicando sus principales funcionalidades definidas en un alto nivel.

Etapa 2: Planeación

- Revisar los objetivos y alcances del proyecto en función de un adecuado balance entre resultado, tiempo y recursos.
- Con base en las funcionalidades del producto definidas en la etapa anterior, crear las historias de usuario en las que estén representados los requisitos de los clientes.
- Listar las tareas y actividades que se deben ejecutar para cumplir con las historias de usuario definidas.
- Estimar el esfuerzo necesario para realizar cada tarea necesaria para cumplir las historias de usuario.
- Secuenciar u ordenar las tareas y actividades en función de las dependencias técnicas entre ellas y de los recursos disponibles. Para las actividades específicas del desarrollo del sistema, debe considerarse, en la priorización, el valor que otorgan a la institución y el riesgo que tienen asociado, asimismo, deben establecerse los criterios de calidad para la aceptación de funcionalidades.
- Elaborar el calendario de requerimientos de recursos en el tiempo, para lograr los alcances deseados.
- Obtener la aprobación para el plan de trabajo.
- Mantener los planes de trabajo balanceados durante todo el desarrollo del proyecto, en función de las variaciones que se produzcan en los alcances, tiempos y recursos.

Etapa 3: Ejecución

- Asignar, controlar, supervisar y liderar el desarrollo de las actividades planeadas.
- Efectuar reuniones de trabajo entre los integrantes del equipo y el líder del proyecto.
- Mantener una comunicación constante entre los diferentes participantes en el proyecto.

- Gestionar la solución de los problemas que puedan surgir durante la ejecución y asegurar la consecución de recursos (dinero, gente, infraestructura tecnológica), para llevar a cabo el proyecto.
- Definir las necesidades de los clientes con sus respectivos criterios de aceptación.

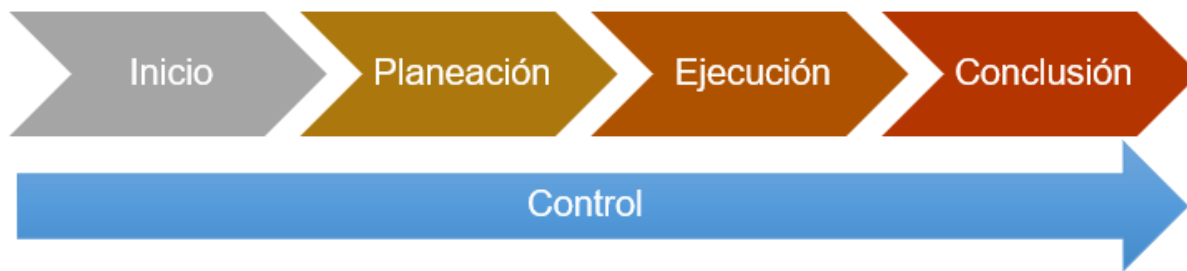
Etapa 4: Control

- Monitorear las desviaciones en lo planificado y determinar sus posibles causas.
- Efectuar las acciones correctivas para lograr la ejecución de lo planeado.
- Evaluar los requerimientos de cambios solicitados por los patrocinadores y los miembros del grupo; determinando el impacto en los alcances, y en los recursos.
- Detectar variaciones en los alcances, en la asignación de recursos o en el tiempo en que se deseen lograr los resultados.
- Asegurar el cumplimiento de las especificaciones o requerimientos definidos para la solución tecnológica, según se establece en el marco de la Gestión de la Calidad.
- Establecer los seguimientos y revisiones de los avances del proyecto.

Etapa 5: Conclusión

- Aplicar mecanismos para la verificación de la satisfacción de las expectativas de los clientes.
- Documentar las lecciones aprendidas durante su ejecución.
- Informar sobre la terminación y los alcances logrados.
- Consolidar toda la documentación generada.
- Elaborar el informe final de proyecto.
- Liberar los recursos asignados.
- Entregar el informe final al Patrocinador.

A continuación, se presenta un diagrama con las etapas de los proyectos de desarrollo de sistemas:



Etapas de un proyecto de desarrollo de sistemas

II. DETALLE DE ETAPAS

0. Anteproyecto

Todo sistema de información que se decida desarrollar, debe estar contemplado en el Plan Táctico de TIC (PTAC), el cual responde a las orientaciones que plantea el Plan Estratégico de Tecnologías de la Información y Comunicaciones (PETIC) de la Contraloría General de la República.

El proceso de actualización del PTAC, que se realiza periódicamente; establece según las prioridades de la organización, cuáles son los proyectos que se deben desarrollar con el fin de aportar los recursos que la organización necesita para mantenerse actualizada y apoyada para el cumplimiento de sus metas y objetivos.

Cuando se realice una propuesta de proyecto, debe elaborarse una ficha de anteproyecto (formato detallado en el anexo 1) que deberá ser presentada ante la jefatura de la UTI, unidad que evaluará la viabilidad del proyecto para someterlo a consideración del Comité Gerencial de Tecnologías de Información y Comunicaciones (CGTIC), el que se encargará de evaluarla y priorizarla. De los anteproyectos planteados, el CGTIC recomienda aquellos que deberían ser tomados en cuenta en la cartera de proyectos del PTAC.

Los anteproyectos que no sean incluidos en el PTAC deberán ser reevaluados en sus alcances y objetivos, cada vez con un mayor nivel de detalle, hasta que llegue el momento de incluirlo en una reformulación del PTAC. Dados los procesos de planeación estratégica, se le dará continuidad al anteproyecto, mientras el mismo responda a los objetivos estratégicos institucionales.

La UTI incorporará a su plan anual operativo (PAO), los proyectos establecidos en el PTAC, conforme con la prioridad definida y los recursos con que se disponga.

Cuando un anteproyecto requiera ajustes significativos en cuanto a los resultados esperados, alcance planeado, efecto, objetivos o productos, deberá reformularse la ficha de anteproyecto y volver a someterlo ante la jefatura de la UTI, quien le realizará un análisis y lo someterá a consideración del CGTIC.

Productos de la etapa:

- Ficha del anteproyecto

Puntos de control:

- Análisis de viabilidad del anteproyecto (UTI y Patrocinador).
- Evaluación del anteproyecto por parte del CGTIC.

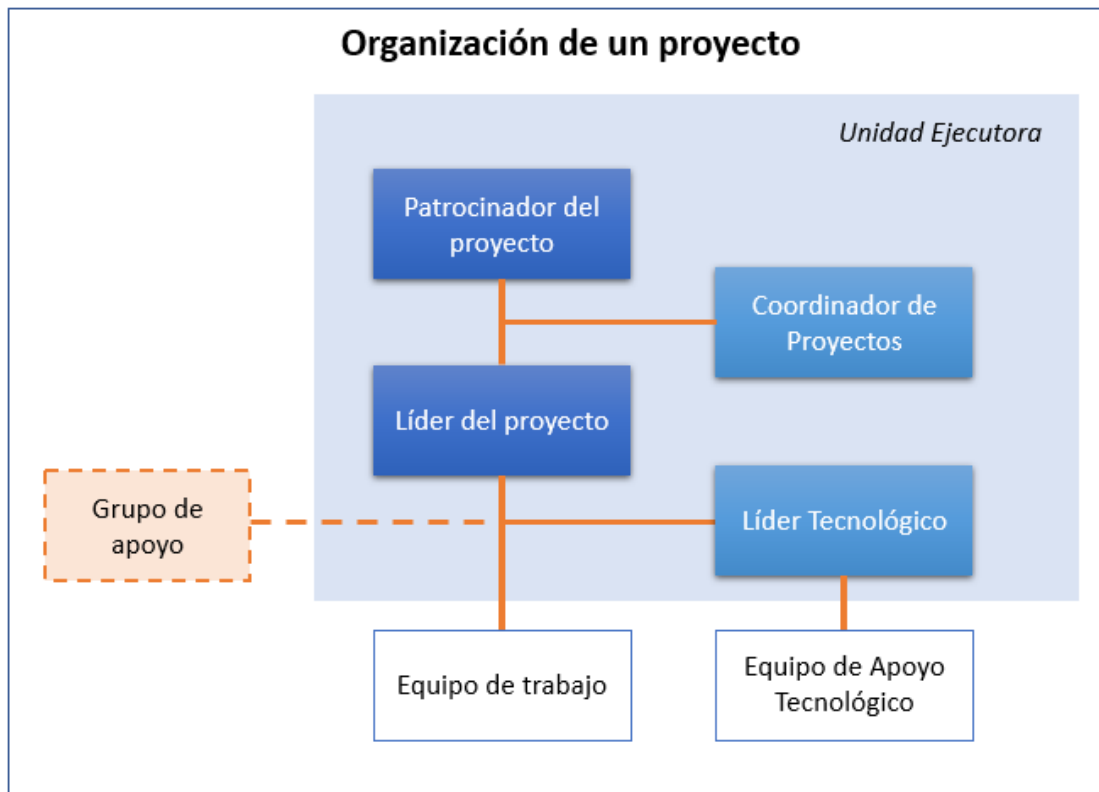
1. Iniciación

En esta etapa se corroboran los alcances globales del proyecto y las expectativas generales de los diferentes interesados. Además, se define la organización del proyecto estableciendo las funciones y responsabilidades de cada uno de los involucrados, así como el perfil deseable de los integrantes del equipo de trabajo.

Una vez establecido formalmente el equipo, deben realizarse las acciones necesarias para definir las principales funcionalidades que se requieren para el proyecto.

1.1 Organización para el proyecto

Todo proyecto debe ser formalizado por medio de una estructura organizativa que responda a la magnitud y complejidad del mismo. La organización del proyecto requiere la interacción de personas de las diferentes áreas, según la siguiente definición de estructuras:



A continuación, se presenta una explicación de cada uno de estos ejecutores del proyecto:

1.1.1 Unidad Ejecutora

La Unidad Ejecutora es el ente coordinador de los aspectos relacionados con el desarrollo del proyecto de TIC.

Sus responsabilidades principales son:

- Aprobar la organización, los recursos, y el cronograma del proyecto.
- Velar por la calidad de los productos de cada una de las etapas y hacer las recomendaciones necesarias.
- Resolver las situaciones que puedan afectar el buen funcionamiento del proyecto.
- Aprobar los productos generados y ejercer los puntos de control establecidos en cada etapa.

Está constituida por:

- Patrocinador(es) del Proyecto
- Coordinador de Proyectos (jefe de la UTI o a quien éste designe)

- Líder del Proyecto.
- Líder Tecnológico.

1.1.2 Patrocinador del proyecto

Es el jefe de la unidad organizacional para la cual se va a desarrollar un proyecto de TIC. Según sea el proyecto pueden verse involucrados más de un patrocinador.

1.1.3 Grupo de Apoyo

Lo conforman funcionarios cuya experiencia y conocimientos son de gran ayuda a nivel de asesoría para el equipo de proyecto. También lo pueden integrar grupos de usuarios con interés en los resultados del proyecto.

1.1.4 Coordinador de proyectos

Esta función la lleva a cabo el jefe de la UTI y puede ser ejecutada por un funcionario de la UTI que éste designe.

1.1.5 Líder del proyecto

El Líder del Proyecto es un funcionario designado por el Patrocinador para administrar y dirigir el proyecto.

1.1.6 Líder Tecnológico del proyecto

Este es un funcionario al cual, por su formación en tecnologías de información y comunicación, experiencia y capacidad, se le ha conferido la responsabilidad de administrar los aspectos tecnológicos del proyecto. Es el responsable directo del desarrollo del proyecto en lo que corresponde a la parte técnica, para ello aplica las políticas, normas y procedimientos de trabajo aprobados.

1.1.7 Equipo de apoyo tecnológico

Son los funcionarios especialistas en el área de tecnologías de información que apoyan la labor del líder tecnológico, cuando así se requiera, en las materias de su especialidad.

1.1.8 Equipo de trabajo

Está conformado por el personal asignado para el desarrollo del proyecto, el cual trabajará en conjunto durante todas las etapas del mismo.

Para el éxito del proyecto, se sugiere que el equipo contemple analistas y programadores, así como personal encargado de operaciones de TI para las tareas correspondientes a bases de datos, ambientes, infraestructura. Asimismo, debe contemplar usuarios expertos. La cantidad de estos participantes varía según la dimensión de cada proyecto.

La información relacionada con la organización del proyecto debe quedar debidamente formalizada y documentada, de acuerdo con el formulario establecido en el anexo 2.

1.2 Lista de funcionalidades del proyecto

Uno de los aspectos más críticos de todo proyecto, es detallar y comunicar lo que cada uno de los interesados (muchos de ellos patrocinadores) esperan. Por ello resulta imprescindible que para cada interesado se establezca lo que espera obtener cuando éste concluya.

Todas las expectativas planteadas deberán ser conocidas por todos los interesados en el proyecto y deberán quedar documentadas en este documento.

Debe plasmarse un listado con todas las funcionalidades de alto nivel o descripciones de producto, las cuales, en la etapa de planificación, serán refinadas por el líder del proyecto.

Las funcionalidades deben plasmarse como historias de usuario, las cuales permiten de manera simple documentar los requerimientos y funcionalidades deseadas por los usuarios.

Las historias de usuario deben responder a tres preguntas:

- ¿Quién lo necesita?
- ¿Qué se necesita?
- ¿Por qué lo necesita?

El formato que debería utilizarse es el siguiente:

Como (rol / persona), quiero (qué / evento / funcionalidad), para (objetivo).

Por ejemplo:

Como declarante, quiero que al incluir la matrícula del bien mueble en el sistema este se conecte con la base de datos del registro nacional para que automáticamente complete los datos del bien inmueble solicitados (marca, modelo, descripción, valor fiscal).

1.2.1 Herramientas para definir la lista de funcionalidades

Para recabar las funcionalidades del proyecto pueden utilizarse las siguientes herramientas:

1.2.1.1 Reuniones con grupos de usuarios

Con el fin de que el equipo de trabajo obtenga información de primera mano acerca de las expectativas del sistema, pueden hacer una reunión con los futuros usuarios o clientes del producto que se va a desarrollar. En un trabajo conjunto en esa reunión, pueden definirse las principales funcionalidades junto con sus criterios de aceptación, promoviendo un entendimiento común entre el equipo y los usuarios interesados.

1.2.1.2 Talleres de historias de usuario

El líder del proyecto facilita una serie de talleres en donde intervenga el equipo de trabajo junto con usuarios interesados. Por medio de trabajos individuales y discusiones en pequeños grupos, se busca definir las historias de usuario junto con los criterios de aceptación de estas. Asimismo, facilita dar prioridad a los requisitos y se valida que sean fáciles de entender y de estimar su esfuerzo.

Estos talleres son un buen ejercicio para discutir y aclarar todos los elementos de un producto y si es necesario, se puede profundizar en los detalles más pequeños para garantizar la claridad de lo que se requiere.

Para la ejecución del taller existe una serie de dinámicas que podrían utilizarse, por ejemplo, definir un desafío al inicio de la sesión que sean trabajados en grupos aleatorios en un periodo de tiempo. Una vez finalizado el tiempo se exponen las soluciones del grupo y se hace una discusión con el fin de obtener distintos puntos de vista. De las ideas y discusiones realizadas, se pueden obtener valiosos insumos para las historias de usuario.

1.2.1.3 Grupos Focales

En los grupos focales se reúne a una serie de usuarios para que, con base en una sesión guiada, proporcionen sus opiniones, percepciones o valoraciones sobre un producto, servicio o resultado deseado. Los miembros del grupo focal, tienen la libertad de hacerse preguntas el uno al otro y de obtener aclaraciones sobre temas o conceptos específicos. Por medio de cuestionamiento, la crítica constructiva y la retroalimentación, los grupos focales conducen a un producto de mejor calidad y con ello contribuyen a la satisfacción de las expectativas de los usuarios. En estas reuniones, los miembros del grupo de enfoque a veces llegan a un consenso en ciertas áreas, mientras que en otras áreas las opiniones pueden ser diferentes. Cuando los miembros del grupo tienen diferentes opiniones o puntos de vista, no se escatiman esfuerzos para resolver las diferencias con el fin de llegar a un consenso.

Las sesiones de grupos focales pueden ayudar a los equipos a crear ideas innovadoras, resolver problemas, y dar sugerencias para mejorar. Estas reuniones generan ideas y retroalimentación de los usuarios potenciales y desarrolladores de productos.

1.2.1.4 Entrevistas con usuarios

Por medio de entrevistas puede darse una inclusión del patrocinador y de los futuros usuarios dentro del proceso de definición de funcionalidades permitiendo un mejor contexto y visión a la hora de definir la lista. Con base en las entrevistas, se identifican y entienden las necesidades y expectativas del patrocinador y los usuarios, además de permitir que se identifiquen opiniones y hechos que facilitan al equipo entender lo que debe desarrollarse.

1.2.1.5 Cuestionarios

Los cuestionarios son una forma económica de obtener una perspectiva estadística cuantitativa y cualitativa de un gran número de futuros usuarios o interesados sobre las necesidades.

Sería de utilidad para aquellos sistemas que van a impactar a un número de usuarios considerables, de los que se requiera recopilar información de utilidad para el desarrollo del sistema como las necesidades existentes o preferencias específicas.

1.3 Insumos de la etapa

- Ficha del anteproyecto

1.4 Productos de la etapa

- Descriptivo de la organización del proyecto
- Historias de usuario en alto nivel

1.5 Puntos de control

- El patrocinador (o patrocinadores) del proyecto y la UTI, oficializa(n) los nombramientos del líder de proyecto, del líder tecnológico y la conformación del equipo de trabajo.
- La unidad ejecutora del proyecto evalúa las historias de usuario con las principales funcionalidades, las cuales conformarán el alcance que tendrá el proyecto.

2. Planeación

En esta etapa se debe especificar con detalle, las diferentes actividades por realizar, las cuales se verán reflejadas en un plan de trabajo. Para ello el equipo de trabajo deberá empezar por analizar los objetivos y alcances del proyecto, identificar los recursos requeridos y establecer un cronograma del mismo. Para apoyo en la administración de los proyectos, puede utilizarse algún software previamente autorizado por la UTI. Con respecto a las funcionalidades recabadas en la etapa anterior, según las historias de usuario, se hará una lista que contendrá la prioridad, riesgos, estimación de esfuerzo, responsable, dependencias y criterios de evaluación.

Como requisito para iniciar con esta etapa, el equipo de trabajo debe estar formalmente constituido y se deben haber recopilado las principales historias de usuario, las cuales determinan el curso de acción para las tareas que deberá realizar el equipo de proyecto, con miras a lograr el objetivo propuesto.

Debido a que la labor de planeación es cíclica, la misma se debe realizar tanto para el inicio, como durante la realización del proyecto. Si el proyecto mostrase atrasos considerables en alguna de sus

etapas, o bien cambios que se presenten durante su ejecución, se actualizará el plan del proyecto para el resto de las actividades por cumplir.

Elaborar el plan es una labor muy delicada y se debe realizar con la participación de todos los miembros del equipo. Como en toda actividad grupal, surgirá una serie de opciones sobre cómo realizar las cosas, por lo que será habilidad del líder de proyecto conciliar criterios, de modo tal que se llegue a un consenso sobre cuáles son las actividades que se deben desarrollar, para llegar a cumplir las finalidades del proyecto.

2.1 Elaboración del Plan de Trabajo

Para la elaboración del plan de trabajo, se recomienda cumplir con los siguientes pasos:

1. Se debe elaborar una lista o enumeración de los entregables del proyecto, para luego determinar todas las actividades que se deben realizar para cumplir cada meta. En esta etapa deben estar definidas las historias de usuario que se desarrollarán en la fase de ejecución, según se especifica en el punto 2.1.1.
2. Por cada actividad se debe tener claro cuál es su propósito y el producto que se quiere lograr a su conclusión. Según sea el caso, deberá contemplar también aquellas actividades necesarias para administrar los riesgos identificados en el proyecto. Para las actividades de la fase de programación, debe contemplarse también los criterios de aceptación tal y como se explica en el punto 2.1.1.2.
3. El producto que se espera lograr por cada actividad debe ser claro y conciso, si no es así se debe desglosar o partir la actividad, en tantas como sea necesario para que cada actividad tenga un producto concreto definido.
4. Cuando se tengan las actividades debidamente enmarcadas en cuanto al producto por lograr, debe indicarse si se requerirán particularidades técnicas o de recurso humano especializado. Estas particularidades deben documentarse en el formulario del anexo 3, con la finalidad de realizar las gestiones administrativas que sean pertinentes para contar con lo requerido en el momento indicado en la planificación de la actividad. De ser necesario algún cambio en la ejecución del proyecto, se realizará una actualización utilizando el mismo formulario.
5. Debe estimarse la duración de cada actividad.

6. Debe priorizarse el orden en que se desarrollarán las actividades por etapa. En el caso de las actividades de la fase de programación, se hará conforme el proceso establecido en el punto 3.1 por medio de iteraciones.
7. Una vez agrupadas las actividades en etapas, se debe proceder a determinar cuál es el tipo de relación que existe entre cada actividad, pudiendo ésta ser, predecesora inmediata o sucesora inmediata o sin relación. Predecesora inmediata será aquella actividad que debe haber sido concluida para que otra sucesora inmediata inicie.
8. Para aquellas actividades ajenas a la programación, deben identificarse las que se pueden ejecutar en paralelo. Entre una actividad predecesora inmediata y otra sucesora inmediata, puede haber un lapso que se denomina holgura, la cual puede ser negativa, cero o positiva. Será cero cuando una actividad de acuerdo con el plan, debe iniciar de inmediato, tan pronto concluye su predecesora. Será positiva cuando entre el tiempo de conclusión de la predecesora y el inicio de la sucesora, pueden transcurrir “n” cantidad de días. Será negativa cuando la actividad sucesora, puede iniciar según el plan, “n” cantidad de días antes de que concluya la actividad antecesora.
9. Con toda la información anterior, se debe proceder a balancear las actividades de manera tal que el entregable de la etapa se logre en el menor tiempo posible. Se entiende como balanceo el agrupar las actividades, de manera tal que muchas de ellas se puedan ejecutar en paralelo, cuando los recursos y las dependencias entre actividades lo permitan.
10. Todas aquellas actividades, exceptuando la programación, que al final del proceso de planeación presenten una holgura de cero, formarán lo que se conoce como la ruta crítica. Estas son las actividades que no se deben retrasar, ya que en caso de hacerlo, implica un atraso de la etapa o sub proyecto. Se debe tener presente que aquellas actividades que presenten holguras positivas, una vez consumidas éstas entran a la ruta crítica y podría también retrasar el proyecto.
11. Debe considerarse que, durante la fase de programación, podrían presentarse variaciones que impliquen actualizar la planificación definida al inicio del proyecto.

2.1.1 Creación de la lista de historias de usuario con sus respectivas actividades o tareas

Para la creación de la lista de historias de usuario que se desarrollarán en el sistema, así como sus respectivas actividades o tareas, debe establecerse una serie de elementos los cuales se definen a continuación:

2.1.1.1 Complementar historias de usuario

El líder del proyecto desarrollará una lista de historia de usuario con una serie de detalles con base en la lista definida en la etapa anterior. De ser necesario, para esta tarea puede solicitar colaboración de usuarios expertos y si es el caso, puede recurrir a las técnicas descritas en el punto 1.2.1.

Las historias de usuario deben ser debidamente refinadas con el fin de que generen un producto que pueda ser calculado y aprobado, y al que el equipo de trabajo se pueda comprometer a realizar.

En estas historias, se definirán tres aspectos claves: la prioridad, el nivel de riesgo y las dependencias.

La prioridad la definirá en una escala de alta, media o baja, o bien, si se desea puede usarse una escala de uno a cinco. Se considera cinco el valor con mayor prioridad.

Para el nivel de riesgo pueden utilizarse una serie de técnicas, sin embargo, se recomienda que se defina un valor con base en la probabilidad de ocurrencia y del impacto que tendría si se materializa por medio de una matriz de probabilidad e impacto.

En las dependencias se establece qué historia de usuario depende de otras para poder realizarse. Incluso la dependencia podría ser de algún recurso especializado.

2.1.1.2 Establecer criterios generales de aceptación

Debe establecerse un conjunto de reglas que apliquen para aceptar todas las historias de usuario definidas. Esta tarea es fundamental para evitar ambigüedades y ayuda a que el equipo se adhiera a las normas establecidas en el Marco de Gestión de Calidad.

Una historia de usuario se considera hecha, cuando al evaluarse cumple con los criterios generales de aceptación, así como los criterios específicos de cada una de ellas.

Estos criterios podrían establecerse en una lista de chequeo, en la que se podrían incluir puntos como, si se documentaron adecuadamente, si se hicieron validaciones de seguridad, si se aprobaron adecuadamente, entre otras.

2.1.1.2 Definir iteraciones con su tiempo estimado

Las historias de usuario van a ser divididas en distintas iteraciones, según el líder del proyecto, y pondrá un tiempo (esfuerzo) estimado para cada una de ellas. Estas estimaciones serán validadas con el equipo de trabajo en la siguiente fase.

2.1.1.3 Creación de tareas

El líder del proyecto se reúne con el equipo de trabajo con el fin de evaluar la propuesta de historias de usuario por cada iteración, así como el esfuerzo estimado de cada una de ellas.

Una vez que haya consenso, se procederá a dividir las historias de usuario en las tareas o actividades necesarias para cumplirlas con sus respectivos responsables y dependencias. Asimismo, se estimará el tiempo por cada tarea. Este proceso puede documentarse como se muestra en el anexo 4.

2.1.1.4 Definición de tareas para la primera iteración

El líder del proyecto discutirá con su equipo de trabajo las tareas de la primera iteración, en una reunión de aproximadamente siete horas. En ella se discutirán las historias de usuario que se incluyen, los criterios de aceptación, los riesgos asociados y las dependencias. Posteriormente se organizarán en cómo llevarán a cabo las tareas con el fin de cumplir la meta establecida para esta iteración.

Si resultará necesario añadir nuevas tareas, el líder del proyecto debe incluirlas en la lista general de tareas y asignarlas a otra iteración, ya que, en la medida de lo posible, se debe evitar modificar las metas establecidas para la iteración, una vez discutidas y aprobadas por el equipo.

El punto 2.1.1.4 se repetirá por cada una de las iteraciones planeadas.

2.2 Insumos de la etapa

- Descriptivo de la Organización del Proyecto
- Historias de usuario de alto nivel

2.3 Productos de la etapa

- Plan de trabajo (incluyendo lista de historias de usuario con sus respectivas actividades o tareas)
- Formulario de planeación de recursos

2.4 Puntos de control

- La Unidad Ejecutora del Proyecto revisa y da su visto bueno al plan de trabajo del proyecto y a la planeación de recursos, como requisito para poder continuar con la etapa de ejecución. Si el plan no satisface las expectativas de los integrantes de la Unidad Ejecutora, ésta puede solicitar al equipo de trabajo una reformulación.

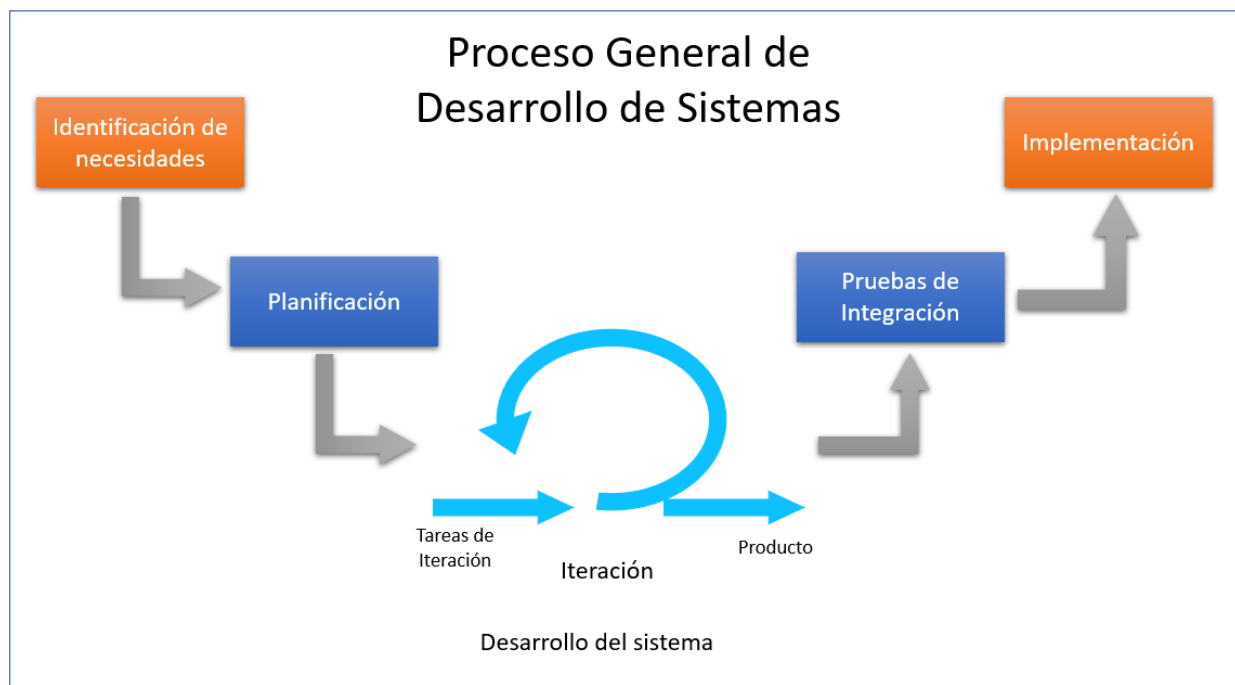
3. Ejecución

El objetivo de esta etapa es desarrollar y cumplir el plan elaborado en la etapa anterior.

Durante esta etapa el Líder de Proyecto deberá mantener informada a la Unidad Ejecutora respecto del avance en el desarrollo de las actividades, situaciones no planeadas que se presenten, actividades no programadas y resolución de problemas.

Dependiendo del mismo plan de trabajo y del tipo de proyecto, durante la etapa de ejecución se crearán productos intermedios que deberán ser revisados y aprobados de acuerdo con la matriz de responsabilidades establecida.

A continuación, se presenta un diagrama general en el que se aprecia el proceso:



En lo referente al desarrollo del sistema, se considera lo siguiente:

3.1 Entregables por iteración

Al finalizar cada iteración, se completa un producto o entregable, según lo planeado en la lista de historia de usuario y lo acordado por el equipo en la reunión de definición de tareas para cada iteración. Este producto debe incluir todas las características y funcionalidades definidas en las historias de usuario y haber cumplido con los criterios de aceptación en las pruebas realizadas. Asimismo, se entregará un informe con información clave del proyecto.

En cada iteración se van realizando tareas como implementación del modelo físico de datos, creación de roles y usuarios, ejecución de pruebas por parte del programador y usuario, entre otras necesarias para cumplir los objetivos definidos.

Para llevar el control por cada iteración, se sugiere poner en un lugar visible, un cuadro como el de la figura que se muestra a continuación, donde en la primera columna se visualizan las historias de usuario definidas para cada iteración, y en las columnas restantes se van moviendo las tareas de cada historia según la etapa en que se encuentren. Al finalizar la iteración se programa el cuadro para la siguiente, incluyendo las nuevas historias de usuario con sus respectivas tareas.

Cuadro de progreso de Iteración				
Historia	Por hacer	En proceso	En pruebas	Hecho
1				
2				
3				
4				

Cuadro de progreso de iteración

En el transcurso de cada iteración se deben identificar y conceptualizar los módulos que va requiriendo el sistema con su respectiva interacción y los roles de usuario necesarios para la operación del sistema. Estos se deben ir documentando en el informe de la iteración.

Asimismo, de ser necesario, el informe puede contener los siguientes aspectos:

- Descripción de procesos: Para los procesos identificados en cada historia de usuario se definen las entradas, procesamiento y salidas.
- Diagrama lógico del modelo de datos: Especificación del modelo entidad – relación y su normalización.
- Diseño físico de la base de datos: Se especifican aspectos como
 - Nombre de tabla y su descripción
 - Restricciones y referencias
 - Campos con su descripción, tipo de dato, valor que asume por defecto, rango o posibles valores permisibles, y mensajes informativos sobre los elementos.

- Estimación del volumen de datos: Estimación de la cantidad de registros que ingresarán para cada tabla, estableciendo cantidad inicial, crecimiento esperado y comportamientos particulares que pudieran presentarse.
- Controles y seguridad que se van a utilizar: Definir elementos que fortalezcan la seguridad y confidencialidad de la información, mediante roles en la base de datos, programas y control de acceso a las funcionalidades. Asimismo, se identifican aquellos eventos que deben dejar registros en bitácoras.
- Organización para la operación del sistema: Identificar y definir los requerimientos operativos y organizacionales en el ámbito de ambiente administrativos en el que se implantará el nuevo sistema.

Como se aprecia, en cada iteración se contemplan aspectos operativos, de ahí la importancia que dentro del equipo haya un miembro con experiencia en esta materia, con el fin de no retrasar las metas propuestas.

Por otra parte, en el transcurso de la iteración, pueden ir surgiendo requerimientos no funcionales que no se contemplaron en las historias de usuarios. Estos deben ser informados al líder del proyecto para que los incluya en la lista de historias de usuarios en las que se detallan todas las tareas, para que sean abarcados en futuras iteraciones.

Al finalizar cada iteración, deben quedar registrados obstáculos que pudieron afectar la productividad del equipo y ser discutidos en la reunión de cierre de la iteración, así como en las reuniones semanales que se describen en el punto 3.2.

3.2 Reuniones semanales

Los miembros del equipo se reúnen con el líder el primer día de la semana para informar el progreso en la iteración, y las actividades que se realizarán en la semana. Estas reuniones deben ser de corta duración, no más de 30 minutos. De ser necesario, se pueden programar otras sesiones similares durante la semana, procurando una duración más corta.

En la reunión deben abarcarse puntualmente las siguientes preguntas:

- ¿Qué terminé en la semana?

- ¿Qué terminaré esta semana?
- ¿Qué impedimentos u obstáculos (si los hay) estoy enfrenando en la actualidad?

Es recomendable que las dos primeras preguntas sean respondidas por los miembros del equipo de manera cuantificable, sin dar respuestas largas y cualitativas.

En caso de ser necesario, los miembros del equipo pueden organizar reuniones adicionales para discutir aspectos que requieran un análisis adicional.

Se recomienda que el equipo de trabajo esté ubicado en un espacio físico cercano, con el fin de facilitar la comunicación entre los mismos.

Las reuniones semanales se documentarán en una minuta según lo establecido en el anexo 5.

3.3 Reuniones de revisión de producto

El equipo de Scrum y el líder del proyecto se reúnen con usuarios o clientes previamente identificados para que se expongan las nuevas funcionalidades o productos creados, y estos sean aceptados de acuerdo con los criterios de cada historia de usuario. De ser posible, debe contarse con la participación del patrocinador y de la unidad ejecutora en estas reuniones.

Estas reuniones se convocan al finalizar cada iteración y permiten que los usuarios identifiquen el avance del proyecto hasta el momento, y puedan identificar o determinar si es necesario plantear algún cambio para futuras iteraciones.

Se espera que los entregables de cada iteración sean aceptados al cumplir con los criterios establecidos, sin embargo, podrían ser rechazados, por lo que tendrían que rectificarse en iteraciones posteriores.

En caso de que el producto de la iteración otorgue valor a la organización y se decida lanzar a producción, deberían considerarse los aspectos que se señalan en la etapa de implementación, para un pase a producción parcial. En estos casos, es obligatorio que el patrocinador participe en la reunión, con el fin de que apruebe el producto que será lanzado.

Para estas reuniones se debe definir una duración de cuatro horas.

3.4 Reuniones de Retrospectiva

Las reuniones de retrospectiva se dan entre el líder del proyecto y el equipo de trabajo. El objetivo de esta es identificar los elementos que se detallan a continuación en referencia a la última iteración:

1. Aspectos que el equipo tiene que seguir haciendo: mejoras prácticas
2. Aspectos que el equipo necesita empezar a hacer: mejoras que deben implementarse
3. Aspectos que el equipo debe dejar de hacer: problemas de proceso y embotellamiento

Dentro de las técnicas que podrían utilizarse está la del barco, consiste en que hipotéticamente, los miembros del equipo son los tripulantes y por medio de notas adhesivas, se ubican en la vela aquellos aspectos que ayudan a que el barco avance, mientras que en las anclas se anotan los aspectos que están frenando al barco. Los aspectos de las velas se potencian mientras que los de las anclas se mitigan.

En la minuta de esta reunión deben quedar consignados los acuerdos con respecto a las mejoras que se implementarán y los problemas que se mitigarán, así como las lecciones aprendidas en la iteración.

Para estas reuniones se define una duración de cuatro horas.

3.5 Ejecución de pruebas de integración y otras pruebas

Durante la etapa de ejecución del proyecto, al contar con el producto final, deben diseñarse y realizarse pruebas de integración; en ellas se realizan pruebas totales del sistema y de su integración con otros sistemas.

De ser necesario, pueden realizarse también pruebas de esfuerzo, pruebas de tiempo de respuesta y pruebas de tráfico en la infraestructura de comunicaciones.

3.6 Capacitación

El Patrocinador coordinará con el líder del proyecto todas las gestiones necesarias para impartir la capacitación a los usuarios del nuevo sistema. De ser necesario puede coordinar con el Centro de Capacitación o la Unidad de Potencial Humano.

Asimismo, el líder del proyecto debe coordinar con el equipo de trabajo la creación de casos de capacitación, con la preparación de usuarios y datos necesarios, considerando también la cantidad de veces que sea necesario replicarla.

3.7 Implementación

Se debe definir el método de implementación más adecuado para el proyecto, se considerará si se realiza un paralelo o no, método de carga de datos por utilizar y cualquier otra decisión estratégica para la finalización exitosa del proyecto. Esta estrategia debe ser acordada con el patrocinador del proyecto, el líder del proyecto, el líder tecnológico y la jefatura de la UTI.

El líder de proyecto, siguiendo criterios de impacto, materialidad, riesgo asociado, sensibilidad y criticidad de la información involucrada, deberá considerar la forma en que se realizará la implementación del sistema, para lograr un adecuado balance costo/beneficio. Entre otras podrá considerar la operación en paralelo, cuyo objetivo será verificar que el sistema nuevo genera resultados similares al sistema que estuviera en ese momento en funcionamiento – manual o automatizado-, pruebas de esfuerzo, cuyo objetivo es poner a prueba el sistema y la plataforma frente a una fuerte demanda de los servicios. Estas pruebas pueden ser reales o simuladas, dependiendo de la disponibilidad de recursos humanos y tecnológicos.

3.7.1 Plan de implementación

Es un breve documento en el que se describen las actividades por realizar para la implementación del sistema, desde el punto de vista técnico y organizacional.

De acuerdo con la complejidad del sistema se debe generar un plan de implementación que incluya la calendarización de actividades, por ejemplo: el traslado al ambiente de producción, la ejecución de pruebas en paralelo y las pruebas de esfuerzo.

3.7.1.1 Instalación del sistema

Las actividades por realizar son las siguientes:

a) Preparar los aspectos relacionados con el ambiente físico y computacional para dar inicio con la operación del sistema; de acuerdo con las características del sistema se recomienda lo siguiente:

- Solicitar el pase a producción de los diferentes componentes del sistema. Esta solicitud la debe realizar el patrocinador del sistema, utilizando el formulario indicado en el anexo 8.
- Activar los componentes de seguridad de acceso al sistema: identificación de usuarios, palabras de paso y atributos de usuario (roles), con la finalidad de garantizar que el ingreso al sistema en operación se realizará de forma segura.
- Verificar que los requerimientos de “hardware”, de “software” y de comunicaciones se encuentran disponibles.
- Verificar que se tenga el mobiliario, la instalación eléctrica (toma corriente, conexión a tierra, protectores de picos) y el espacio físico necesario y acondicionado para su operación.
- Verificar que los materiales que use el sistema se encuentren disponibles.

b) Realizar la instalación del sistema en el ambiente de producción en coordinación con el DBA.

3.7.1.2 Conversión y carga inicial de datos

Las actividades que se realizarán. podrían ser las siguientes:

a) Confirmar que el ambiente computacional ("software", "hardware") y personal, requerido para la carga o digitación de los datos al sistema, se encuentre disponible, según lo indicado en el plan. De lo contrario, realizar los ajustes necesarios para asegurar que la carga de los datos iniciales se haga en una forma satisfactoria.

b) Ejecutar las aplicaciones desarrolladas para la conversión y carga inicial de los datos al sistema.

c) Es recomendable realizar una depuración de los datos por convertir, para asegurar que no se incluyan datos erróneos al sistema.

- d) Verificar que los datos introducidos se encuentren correctos y completos, conforme con las cifras de control definidas.
- e) Comprobación por parte del líder del proyecto y del analista responsable, que la conversión y carga de datos se realizó en una forma satisfactoria.

3.7.1.3 Ejecución del paralelo

En caso de que se defina conveniente la ejecución de un paralelo, las actividades por realizar serían las siguientes:

- a) Determinar la duración y forma en que se realizará el paralelo, de acuerdo con la naturaleza y complejidad del sistema.
- b) Determinar las funciones y responsabilidades del personal técnico y del usuario que participa en el paralelo.
- c) Establecer los criterios de aceptación esperados para evaluar los resultados que se tengan al final de la ejecución del paralelo.
- d) Iniciar el paralelo brindando asistencia al usuario en forma continua, para asegurar que los procedimientos se realizan en la forma correcta. Asimismo, identificar y corregir los problemas que se presenten.
- e) Documentar los resultados de la ejecución del paralelo, considerando los criterios de evaluación establecidos. Indicar el criterio de aceptación considerado y el nombre de la persona que aprueba.

3.8 Insumos de la etapa

- Plan de trabajo para el proyecto
- Formulario de planeación de recursos

3.9 Productos de la etapa

- Entregables por iteración

- Informes de avance
- Minutas de reuniones
- Producto final
- Plan de implementación

3.10 Puntos de Control

- La Unidad Ejecutora del Proyecto revisa los informes de avance que emite el líder del proyecto, así como los ajustes y actualizaciones realizadas al cronograma de trabajo.
- La Unidad Ejecutora del Proyecto revisa los productos intermedios propios del tipo de proyecto de TIC que se desarrolla. En algunos casos el equipo de trabajo requerirá que se apruebe determinado producto intermedio, antes de continuar con el desarrollo de tareas subsiguientes.

4. Control

El control es una acción permanente durante el desarrollo del proyecto. Su fin primordial es detectar desviaciones del plan de ejecución en forma oportuna, de manera que permita tomar acciones correctivas y preventivas. De ser necesario, se deberá retomar el proceso de planeación, para ejecutar las acciones necesarias.

La meta del control es lograr que los objetivos definidos en el plan de trabajo se cumplan, a partir del seguimiento, ajuste y realimentación de las acciones planeadas y ejecutadas.

El proceso de control del proyecto valora como insumo los cambios en el entorno, en los recursos, en las necesidades por solventar, las acciones realizadas y el plan de trabajo; para emitir acciones correctivas y acciones preventivas.

4.1 Actualización del cronograma de trabajo

El cronograma debe actualizarse periódicamente con el detalle de las actividades que se han completado, el porcentaje de avance de las actividades que están en proceso y los ajustes propios de un proyecto en ejecución donde se presentan nuevas tareas no planeadas o la necesidad de ajustar fechas, producto de desfases que deben quedar debidamente documentados.

Se debe hacer todo el esfuerzo para que el plan se cumpla, sin embargo, todo plan no es más que una aproximación del futuro y por más cuidado y experiencia que se haya puesto en la elaboración del mismo, siempre existirá una serie de factores que se pueden haber escapado durante la elaboración del mismo. Otro aspecto muy importante es que el plan está enmarcado en una realidad de la organización, como esta es un ente vivo y cambiante, así lo será el plan.

4.2 Control de cambios

Durante la ejecución de cualquier proyecto se pueden presentar cambios que afecten directa o indirectamente el logro de los objetivos y de las metas. El proceso de control del proyecto deberá analizar; entre otros, los cambios en las condiciones del entorno, en los recursos, en las necesidades por solventar, en la tecnología, en los objetivos y cambios en la estrategia de solución.

Todo cambio que afecte el curso de acción, los alcances o la estrategia de solución, debe quedar documentado. Además, el líder del proyecto deberá hacer del conocimiento de la Unidad Ejecutora todo cambio importante, la cual tiene que aprobar si se atiende o no dicho cambio. Para llevar el control de cambios del proyecto se debe usar el formulario que se detalla en el anexo 6.

Para cambios que se originen en el proceso de programación del sistema, el líder del proyecto velará por reorganizar la lista de historia de usuarios y actividades, tratando de mantener los tiempos que se habían establecido en la planificación. En caso de que alguno de los cambios implique un cambio considerable en los plazos o en el alcance, deberá llenarse el formulario del anexo 6 y revisarse con la unidad ejecutora.

4.3 Insumos de la etapa

- Informes de avance
- Plan de trabajo
- Formulario de planeación de recursos
- Minutas de reuniones

4.4 Productos de la etapa

- Cronograma de proyecto actualizado
- Formularios de control de cambios
- Acciones correctivas

4.5 Puntos de control

- La Unidad Ejecutora del Proyecto revisa los informes que emite el equipo de proyecto y el control de avance (cronograma actualizado); cuando identifique tareas desfasadas aprobará acciones correctivas o preventivas tendientes a mitigar el desfase. La Unidad Ejecutora puede decidir retomar el proceso de planeación y solicitar una nueva versión del plan al líder de proyecto.
- La Unidad Ejecutora analiza los cambios propuestos, la justificación, el impacto y la recomendación, para definir si aprueba, rechaza o pospone la atención de dicho cambio. El patrocinador firma el formulario en el que se avala dicho resultado.

5. Conclusión

En esta etapa se revisa el cumplimiento de las metas iniciales y sus variaciones, para efectos de realizar el cierre del proyecto.

Toda etapa de conclusión de un proyecto debe cumplir con las siguientes actividades básicas:

- Enterar a los patrocinadores sobre los resultados del proyecto.
- Entrega de productos con su respectiva aprobación.
- Liberar los recursos que aún estén asignados al proyecto.
- Documentar los procesos finales.

5.1 Informe de conclusión del proyecto

Todo Líder de Proyecto, con el apoyo del resto de los integrantes del equipo, tiene que elaborar el informe final de cierre del proyecto, el cual debe cubrir los siguientes puntos:

- Resumen ejecutivo con los principales logros, comparados con las metas del proyecto.
- Puntos o tareas que quedaron pendientes, ya sea porque requieren de una mayor investigación o elaboración; o porque se deberán retomar para una segunda versión del proyecto.
- Recomendaciones para mejorar la administración y ejecución de proyectos futuros (lecciones aprendidas).

5.2 Aceptación de los productos finales del proyecto

La Unidad Ejecutora debe revisar y dar por aceptados a satisfacción los productos finales del proyecto. Si algún producto no se considera como terminado el proyecto no está en fase de conclusión.

El líder de proyecto deberá informar a la Unidad Ejecutora de los productos finales que entrega, mediante el formulario indicado en el anexo 7.

Luego de la respectiva revisión, el Patrocinador debe dejar constancia de la aceptación de los productos y solicitar su correspondiente paso a producción mediante el formulario indicado en el anexo 8 y se pone en marcha el Plan de Implementación

5.3 Entrega del sistema

Se comunica a los usuarios y a la administración en general (si fuese necesario), de la culminación del proyecto y la fecha de inicio de operación del sistema.

5.4 Expediente actualizado del proyecto

Toda la información relativa al proyecto que recién termina es de vital importancia para otros proyectos o trabajos futuros. Es por ello que el líder del proyecto deberá dejar debidamente completo y actualizado el expediente, el cual tiene que contener todos los entregables de cada fase del proyecto y debe mantenerse en formato digital, de acuerdo con el expediente electrónico definido en la CGR.

5.5 Insumos de la etapa

- Cronograma actualizado con ejecución de tareas.
- Documentos desarrollados durante el proyecto.

5.6 Productos de la etapa

- Informe de conclusión del proyecto
- Formulario de aceptación a satisfacción de los productos.
- Entrega del producto
- Expediente actualizado del proyecto

5.7 Puntos de control

- La Unidad Ejecutora del Proyecto revisa el informe final del proyecto y lo hace del conocimiento de todos los interesados.
- El Patrocinador da por aceptados los productos finales del proyecto.

6. Evaluación del producto entregado

Como una buena práctica se puede realizar una evaluación post implementación, la cual se puede aplicar a los usuarios del sistema, en un periodo comprendido entre dos semanas y tres meses según la complejidad y estacionalidad del sistema.

Para tales efectos se puede aplicar el cuestionario definido en el anexo 9 así como una sesión de evaluación con el patrocinador, líder de proyecto y equipo de trabajo.

7. Anexos de la metodología

7.1 Anexo 1: Ficha del Anteproyecto

Nombre del Proyecto:	Indique el nombre del proyecto
Fecha de Solicitud:	Indique la fecha en que se presenta la solicitud
Solicitado por:	Indique el nombre de la persona que solicita el proyecto

I- JUSTIFICACIÓN DEL PROYECTO

La justificación del proyecto debe abarcar los siguientes aspectos:

- Los antecedentes (breve y concisa explicación del origen del proyecto)
- Problemática o situación actual que motiva la propuesta del proyecto.
- Resultado esperado: Deben establecerse los resultados esperados de acuerdo con los objetivos estratégicos definidos por la Contraloría, a los cuales se considera que el proyecto contribuye. Debe anotarse la identificación y la descripción del objetivo tal como se establece en el documento de la Estrategia Institucional, para interrelacionar la planificación con la estrategia institucional. En el caso de existir contribución directa con más de un objetivo estratégico, éstos deben incluirse respetando un criterio de mayor a menor contribución.
- Efecto: Para que un proyecto sea viable dentro de la CGR, el mismo debe aportar un valor agregado a la misma, o sea que el mismo coadyuve a que la organización cumpla sus fines y objetivos. En estos términos se deben aportar los razonamientos que muestren como el proyecto ayudará a que la CGR sea más eficiente y productiva. Un proyecto que no se pueda justificar en los términos anteriores, simplemente no se debe realizar. En este apartado pueden señalarse los efectos negativos por la no realización del proyecto.

II- OBJETIVOS

Objetivo General:	<p>Un objetivo es aquello que se quiere conseguir, lo que se quiere hacer, una finalidad. Para su redacción se debe iniciar con un verbo en infinitivo que esté acorde con el grado de profundidad que se quiere alcanzar con el proyecto.</p> <p>Es necesario que el objetivo general cumpla con las siguientes características de calidad:</p> <ul style="list-style-type: none"> • Viable. • Lógico. • Acorde con la realidad. • Lenguaje claro y simple. • Concreto, redactado de una manera mensurable que permita verificar su cumplimiento en el tiempo. • Costo razonable.
Objetivos Específicos:	<p>Los objetivos específicos detallan o especifican al objetivo general. De ser necesario por la naturaleza del proyecto que se desarrollará se puede definir un objetivo general y varios objetivos específicos, que consideren las características antes enumeradas. Igualmente, los</p>

	objetivos específicos deben cumplir con las características de calidad descritas para los objetivos generales.
--	--

III- SITUACIÓN PROPUESTA	
Productos:	Indique el producto o productos que se esperan obtener con el proyecto para satisfacer los objetivos
Funcionalidades esperadas:	Indique de forma general las funcionalidades principales que se espera que contenga el sistema que se desarrollará
Alcance del proyecto:	Indique el ámbito de acción propio del proyecto
Principales usuarios:	Indique los futuros usuarios del proyecto
Beneficios esperados:	Indique que beneficios se espera obtener con la implementación del proyecto

IV. ESTUDIO DE MERCADO
Indique si en otras instituciones ha surgido la misma necesidad y cómo la han abarcado. Para elaborar este apartado es necesario que se investigue situaciones similares en otras instituciones que puedan ser un insumo para tomar la decisión de la viabilidad del proyecto.

V. RECURSOS	
Recurso Humano:	Indique las personas necesarias para conformar el equipo del proyecto con la cantidad de horas que se requerirán.
Infraestructura:	Indique si para el desarrollo del proyecto se requiere infraestructura tecnológica adicional a la que cuenta la institución.
Herramientas:	Indique si existe algún requerimiento especial de herramientas para desarrollar el proyecto.
Tiempo esperado:	Indique la duración estimada del proyecto.
Presupuesto para adquisiciones:	En caso de que se requiera realizar alguna adquisición ya sea de servicios, infraestructura o herramientas, indique el presupuesto estimado que se necesita.

VI. RIESGOS
Enumere los principales riesgos que podrían presentarse en el proyecto, junto con posibles medidas de mitigación.

V. ELABORACIÓN DEL ANTEPROYECTO			
Hecho por:	Indique el nombre del funcionario que está promoviendo en su etapa inicial el proyecto.	Fecha	
Revisado por:	Indique el nombre del funcionario que dio el visto bueno al anteproyecto para ser presentado ante el CGTIC	Fecha	

7.2 Anexo 2: Descriptivo de la organización del proyecto

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- CONFORMACIÓN DEL EQUIPO DE TRABAJO	
Patrocinador:	Indique el nombre del patrocinador o patrocinadores del proyecto, su puesto y el área/unidad/división a la que pertenece
Líder del proyecto:	Indique el nombre del líder del proyecto y el área/unidad/división a la que pertenece
Líder Tecnológico	Indique el nombre del líder tecnológico y el área/unidad/división a la que pertenece
Equipo de trabajo:	Indique los nombres de los integrantes del equipo de trabajo y el área/unidad/división a la que pertenece

II. APROBACIÓN DEL EQUIPO DE TRABAJO			
Hecho por	Indique el nombre del funcionario que realizó el documento	Firma y fecha:	
Aprobado por	Indique el nombre o los nombres de los funcionarios que aprueban el equipo de trabajo	Firma y fecha:	

7.3 Anexo 3: Planeación de Recursos Específicos

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- RECURSOS ESPECIFICOS REQUERIDOS	
Recurso Humano:	<p>En este campo se debe anotar las personas que no forman parte del equipo pero que son requeridas para el desarrollo de una tarea específica durante el ciclo de vida del proyecto, con el fin de que las mismas sean ubicadas a tiempo, según sea el caso. Además, deberá indicarse para cada persona el grado de dedicación al proyecto, dentro de un periodo debidamente especificado en esta sección.</p> <p>Según sea el caso se debe anotar el perfil requerido para el recurso; por ejemplo: se requiere un programador en JAVA con vasta experiencia, durante un lapso de cinco meses, para laborar a medio tiempo.</p>
Recursos Tecnológicos:	<p>En este campo se deben anotar aquellos recursos tecnológicos con que se debe contar en un momento determinado, para la buena marcha del proyecto. Es necesario indicar cantidades, tiempo y especificaciones técnicas.</p> <p>Por ejemplo:</p> <ul style="list-style-type: none"> -Se requiere contar con un servidor instalado en la red central de la CGR, a partir del mes de octubre, con las siguientes características técnicas: -Se debe contar, a partir de julio, con un Sistema Administrador de Bases de Datos, Oracle 11g, con 12 licencias de prueba, instalado en el servidor de pruebas de la CGR.
Materiales:	<p>En este campo se deben anotar aquellos recursos materiales, necesarios para la buena marcha del proyecto. Es necesario indicar cantidades, tiempo y características.</p> <p>Por ejemplo:</p> <ul style="list-style-type: none"> -Se requiere contar con una pizarra, post-it de cinco colores de dimensiones XX, y marcadores para pizarra para el día XX.
Recursos Financieros:	<p>En este campo se debe hacer referencia a un presupuesto de gastos que se tiene aprobado si es necesario realizar adquisiciones. Así como las posibles necesidades que deberían gestionarse y la fecha prevista en que se requieren.</p>

II. APROBACIÓN DE RECURSOS ESPECÍFICOS			
Hecho por	Indique el nombre del funcionario que realizó el documento	Firma y fecha:	
Aprobado por	Indique el nombre o los nombres de los funcionarios que aprueban el equipo de trabajo	Firma y fecha:	

7.4 Anexo 4: Formato de Historias de Usuario

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- HISTORIA DE USUARIO	
Número:	Indique el número de historia de usuario. Es consecutivo y único.
Nombre de historia:	Defina un nombre corto para la historia
Usuario:	Indique el tipo de usuario que propone la historia
Prioridad:	Indique la prioridad (alta, media baja)
Nivel de riesgo:	Indique el nivel de riesgo (bajo, medio, alto)
Iteración:	Indique la iteración en la que se desarrollara la historia de usuario
Descripción:	<p>Desarrolle la historia de usuario de acuerdo con el formato establecido.</p> <p>Las historias de usuario deben responder a tres preguntas:</p> <ul style="list-style-type: none"> • ¿Quién lo necesita? • ¿Qué se necesita? • ¿Por qué lo necesita?
Criterios de aceptación:	Indique los criterios que se probaran con el fin de determinar si el usuario acepta el producto desarrollado para satisfacer la historia de usuario.
Dependencias:	Indique las dependencias de la historia de usuario.

II. APROBACIÓN DE HISTORIA DE USUARIO

Hecho por	Indique el nombre del funcionario que realizó el documento	Firma y fecha:	
Aprobado por	Indique el nombre o los nombres de los funcionarios que aprueban el equipo de trabajo	Firma y fecha:	

Para las tareas asociadas a cada historia se llevará el siguiente formato:

Número de Historia	Enunciado	Actividad	Estado	Iteración	Prioridad	Esfuerzo	Observaciones

- En Actividad se definirá la tarea a realizar.
- El estado puede ser: por hacer, en proceso, en pruebas o hecho.
- Iteración se refiere al número de iteración para la cual está programada la tarea.
- La prioridad puede ser alta, media o baja.
- El esfuerzo corresponde a la métrica definida para controlar la iteración. Generalmente es un número que, a través de gráficos, permite medir el avance.
- En observaciones pueden señalarse comentarios adicionales, o bien las dependencias si es el caso.

7.5 Anexo 5: Formato para minutas de reunión

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- ASPECTOS GENERALES	
Fecha:	Indique la fecha de la reunión
Hora de inicio:	Indique la hora de inicio de la reunión
Hora de finalización:	Indique la hora en que finalizó la reunión
Participantes:	Enliste los participantes de la reunión

II-ASUNTOS TRATADOS	
Avances reportados:	Indique en resumen los avances reportados por los participantes y el seguimiento a acuerdos anteriores.
Actividades programadas:	Indique que actividades se programaron junto con su respectivo plazo
Acuerdos tomados:	Indique los acuerdos tomados, con el respectivo responsable y el plazo.

II. APROBACIÓN DE MINUTA			
Hecho por:	Indique el nombre del funcionario que realizó el documento	Firma y fecha:	
Aprobado por:	Indique el nombre o los nombres de los funcionarios que aprueban la minuta.	Firma y fecha:	

7.6 Anexo 6: Formato para control de cambios

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- CONTROL DE CAMBIOS EN EL PROYECTO	
Fecha:	Indique la fecha en que se solicita el cambio
Descripción del cambio:	Describa en qué consiste el cambio que se está solicitando
Justificación:	Indique el por qué se está solicitando el cambio
Impacto:	Debe valorar cual es el impacto para el proyecto, tomando en cuenta las tareas que hay que agregar a la lista y los recursos como tiempo, normativa y relevancia del cambio para el logro de los objetivos del patrocinador.
Resultado de la valoración:	Luego de valorar la solicitud, se resuelve si se aprueba, rechaza o posterga con la respectiva valoración.

II. APROBACIÓN DEL CAMBIO			
Solicitado por:	Indique el nombre de la persona que solicita el cambio.	Firma y fecha:	
Valorado por	Indique el nombre de la persona que valoro la solicitud.	Firma y fecha:	

7.7 Anexo 7: Entrega de productos finales

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- ENTREGA DE PRODUCTOS FINALES	
Fecha:	Indique la fecha en que se solicita el cambio
Productos entregados:	Describa los productos finales que se entregaron
Observaciones:	En caso de que exista alguna observación se indica en este apartado.

II. FIRMAS			
Hecho por:	Indique el nombre de la persona que realizó el documento.	Firma y fecha:	
Aprobado por:	Indique el nombre de la persona que aprobó la entrega.	Firma y fecha:	

7.8 Anexo 8: Aceptación del sistema y solicitud de pase a producción

Nombre del Proyecto:	Indique el nombre del proyecto
Código del Proyecto:	Indique el código asignado al proyecto en el Sistema de Gestión Documental Electrónica

I- ACEPTACIÓN DEL SISTEMA	
Aceptación:	Luego de realizar las pruebas correspondientes al sistema y verificar los criterios de aceptación definidos, lo doy por aceptado y solicito que sea trasladado al ambiente de producción.
Observaciones:	En caso de que exista alguna observación se indica en este apartado.

II. FIRMAS			
Patrocinador:	Indique el nombre del Patrocinador del proyecto	Firma y fecha:	

7.9 Anexo 9: Encuesta para medir la satisfacción de usuarios

I- ENCUESTA DE SATISFACCIÓN DE USUARIOS	
<i>Conteste las siguientes preguntas con la siguiente escala (nunca, pocas veces, muchas veces, siempre) a menos que en la pregunta se especifique otro tipo de respuesta</i>	
CALIDAD EN EL USO	
Eficacia:	¿El sistema le permite alcanzar los objetivos de las labores asignadas?
Satisfacción:	¿El desempeño del sistema es satisfactorio?
Utilidad:	¿El sistema le permite cumplir los tiempos requeridos en las labores asignadas?
Ausencia de riesgo	¿El sistema mitiga el riesgo de incurrir en errores u omisiones en la ejecución de las labores asignadas?
Comodidad	¿Está satisfecho con las herramientas tecnológicas ofrecidas para sus labores?
Capacitación	¿Recibió capacitación para el sistema? (Si / No)
CALIDAD EN EL PRODUCTO	
Totalidad de la cobertura funcional	¿El sistema contempla la totalidad de las tareas del proceso asignado?
Corrección funcional	¿El sistema proporciona resultados correctos y precisos?
Adecuación funcional	¿El sistema le facilita las labores asignadas?
Comportamiento relativo al tiempo	¿El tiempo de respuesta del procesamiento es adecuado?
Compatibilidad	¿El sistema tiene cualidades que facilitan la extracción o el intercambio de datos con otros sistemas? (Sí/No)
Facilidad de uso	¿El sistema es fácil de usar? (Sí / No)
Facilidad de aprendizaje	¿El sistema tiene tutoriales o manuales que le permiten una mejor comprensión de las funcionalidades? (Sí/No)

Estética de la interfaz del usuario	¿Los elementos de estética (colores, diseños) del sistema son satisfactorios? (Sí/No)
Disponibilidad	¿Considera que la disponibilidad del sistema es? (alta, baja, media)
Tolerancia a fallas	En caso de alguna eventualidad, ¿cómo es la restauración del sistema? (mala, buena, regular)
Seguridad	¿El sistema resguarda la confidencialidad, integridad y autenticidad de la información?
Capacidad de mantenimiento	¿El mantenimiento que se da al sistema es? (bueno, malo, regular)
Capacidad de ser analizado	¿Considera adecuada la manera en que debe reportar incidentes del sistema? (sí/no)
Seguimiento de la UTI	¿Considera que los errores se atienden de manera adecuada? (Sí/No)
CALIDAD EN LOS DATOS	
Exactitud:	¿Considera que los datos (campos, títulos, descripciones) en el sistema, son de calidad? (Sí/No)
Compleitud	¿Considera que la información que se maneja en el sistema es de calidad, tomando en cuenta que sea completa y precisa? (Sí/No)
Credibilidad	¿Cómo es su confianza hacia los datos generados por el sistema? (alta, media, baja)
Actualidad	¿La información del sistema está actualizada?
Conformidad	¿La información del sistema cuanta con formatos estandarizados para apoyar la calidad de los datos?
Eficiencia	¿La estandarización y la calidad que presentan los datos del sistema favorece el rendimiento?
Comprensibilidad	¿Los datos generados por el sistema son comprensibles? (Sí/No)

7.10 Anexo 10: Estándar de nomenclatura

I- SISTEMA Y OBJETOS DE LA BASE DE DATOS

Ítem	Declaración del nombre	Ejemplos
Sistema	El sistema tendrá un nombre descriptivo de su funcionalidad y además se le asignarán tres letras como siglas, que permitan estandarizar los nombres de roles y programas. Estas siglas deben tener de 3 a 4 letras.	SCP = Sistema Control de Procesos
Usuario dueño del sistema (máx. 8 caracteres)	Es el usuario dueño de esquema con que están creados todos los objetos de la base de datos (roles, tablas, vistas, etc.) El nombre se obtiene de la siguiente manera [siglas sistema]_own	SCP_OWN
Roles (máx. 30 caracteres)	SSS_ROL_Funcionalidad SSS: código del sistema, ROL: Constante Funcionalidad: Actividad que permite.	SCP_ROL_OPR
Tablas (máx. 30 caracteres)	SSS_TT_Nombre SSS: código del sistema, TT: Tipo de tabla CA Catálogo de datos TA Tabla de entidad TE Tabla temporal VM Vista materializada Nombre: Nombre en singular de la entidad.	SCP_TA_AUTORIZACION
Índices (máx. 30 caracteres)	SSS_IN_Entidad_ID SSS: código del sistema,	SCP_PK_AUTORIZACION SCP_FK_AUTORIZACION_INST

	<p>IN: Tipo de índice: PK Clave primaria FK Clave foránea IX Atributo en la tabla</p> <p>Tabla: Nombre de la entidad. ID: Entidad hacia la que se hace el FK o atributos en el caso de IX.</p>	SCP_IX_AUTORIZACION_NOM_A PE1
Vistas (máx. 30 caracteres)	<p>SSS_VI_Nombre</p> <p>SSS: código del sistema, VI: Constante</p> <p>Nombre: Nombre en singular de la vista.</p>	SCP_VI_AUTORIZACION
Secuencias (máx. 30 caracteres)	<p>SSS_SC_Tabla_ID</p> <p>SSS: código del sistema,</p> <p>SC: Constante</p> <p>Tabla: Nombre de la Entidad.</p> <p>ID: Identificador de la secuencia cuando existe más de una por entidad.</p>	SCP_SC_NUM_ASIENTO SCP_SC_NUM_ASIENTO_CLAV E1 SCP_SC_NUM_ASIENTO_CLAV E2
Constraints (máx. 30 caracteres)	<p>SSS_TT_Complemento</p> <p>SSS: código del sistema,</p> <p>TT: Tipo de constraints: Para las claves primarias se utiliza la constante "PK" seguido del nombre de la entidad SSS_PK_[entidad]</p> <p>Para las claves foráneas se utiliza la constante "FK" seguido del nombre de la entidad a la que pertenece el FK (ENTFK) y del nombre de la entidad referenciada (ENTREF) SSS_FK_ENTFK_ENTREF</p> <p>En los constraints de tipo check se utiliza la constante "CK", el nombre de la entidad a la que pertenece el CK (ENTCK) y una descripción del atributo a validar (ATRIBVAL)</p>	SCP_PK_AUTORIZACION SCP_CK_AUTORIZACION_NIVEL SCP_FK_AUTORIZACION_INST SCP_FK_EMPLEADO_DEPART

	SSS_CK_ENTCK_ATRIBVAL	
Tablespace	<p>El nombre de los tablespaces se asigna con las siglas del sistema, el indicador de tipo de tablespace y una secuencia de dos dígitos.</p> <p>Tipo: D=Datos I=Índices</p> <p>[siglas sistema][tipo][99]</p>	SCPD01
Enlace (DBLINK)	<p>Utilizar el literal "DBL_" seguido de la instancia que origina la consulta y la instancia de donde se obtendrán los datos de la siguiente manera:</p> <p>DBL_[BD Origen]_[BD Destino]</p>	DBL_CGR1_CGR3
Archivo de BD	<p>SSSTNN.dbf</p> <p>El nombre de los archivos se asigna con las siglas del sistema, indicador de tipo y una secuencia de dos dígitos con la extensión ".dbf"</p> <p>SSS: Código del sistema T: Tipo de archivo</p> <p>D=Datos / I=Índices / T=Temporal</p> <p>siglas sistema][tipo][99]</p>	SCPD01.dbf
Código almacenado en la BD	<p>SSS_TTT_Funcionalidad</p> <p>Se les debe asignar el nombre de la siguiente forma: [siglas de sistema][tipo][nombre]</p> <p>Donde los tipos son:</p> <p>PCK = Paquete de programas PRC = Procedimiento</p> <p>FNC = Función</p> <p>TRG = Trigger de BD</p>	<p>SCP_PCK_UTILITARIOS</p> <p>SCP_PRC_ASIGNA_VACAC</p> <p>SCP_FNC_CONVER_MONEDA</p> <p>SCP_TBR_EMPLEADOS</p> <p>SCP_TAR_EMPLEADOS</p> <p>SCP_TBR_EMPLEADOS_ACTPLA ZA</p>

	<p>TBR = Trigger de before row, maneja las validaciones o cambios en los valores de los campos, se divide según la acción (insert, update o delete)</p> <p>TAR = Trigger de after row, maneja las afectaciones a otras entidades, manejo de bitácoras o envío de correos.</p> <p>En caso de necesitar diferenciar entre varios debe indicarse la funcionalidad.</p>	
Servidor	<p>Sxyzwtmn En donde:</p> <p>X: (F)ísico / (V)irtual</p> <p>Y: (W)indows / (S)olaris /</p>	
	<p>(U)bunto / (L)inux</p> <p>Z: (A) 32 bits / (B) 64 bits W: (I)ntel / (R)isc</p> <p>T: (B)ase datos / (S)ervicio / (C)orreo / (A)plicación</p> <p>NN: Secuencia</p>	
Variables	<p>p_: Parámetros de funciones y procedimientos</p> <p>l_: Variables locales de funciones y procedimientos</p> <p>g: Variables globales de paquetes</p> <p>lc_: Constantes locales de funciones y procedimientos</p> <p>gc_ Constantes globales de paquetes</p>	

II-ABREVIATURAS PARA NOMBRAR CAMPOS EN LA BASE DE DATOS			
Abrev.	Significado	Abrev.	Significado
anio	año	ind	Indicador
apdo	apartado	mov	Movimiento
cant	cantidad	nom	Nombre
ced	cédula	num	Número
cod	código	obs	observaciones
cons	consecutivo	proc	procedimiento
desc	descripción	seq	secuencia (generada por bd)

dir	dirección	tel	Teléfono
email	correo electrónico	url	dirección web

III- PROGRAMAS

Los programas llevarán un nombre basado en las siglas del sistema, el tipo de programa y un mnemónico representativo a la funcionalidad del programa de la siguiente manera:

[siglas del sistema][tipo programa][mnemónico]

En el caso de sistemas que estén formados por sub sistemas o módulos es posible la utilización de un nivel de identificación del módulo:

[siglas del sistema][Módulo][tipo programa][mnemónico]

Tipos de programas:

F – Forma

R – Reporte

M – Menú

Si el nombre de la forma está formado por varias palabras o abreviaturas la primera letra de cada una de ellas debe estar en mayúscula

IV- DOCUMENTACIÓN EN EL CÓDIGO

Todos los programas deben poseer internamente una breve descripción de lo que realizan (comentarios al inicio del programa); además de información sobre la creación del programa y sus modificaciones. En los casos en que haya una función muy especial (que realice un cálculo especial) ésta debe ser detallada de manera que se conozca qué y cómo lo hace. Esto incluye a los programas tipo forma en la construcción de sus procedimientos y triggers, código almacenado en la base de datos, código java y cualquier otro programa desarrollado en la UTI.

Ejemplo de documentación de una función donde la sección de valores de salida aplica para este tipo de programas No para los procedimientos:

```

Create or Replace Function eje_f_aumento(p_base Number,
                                         p_porc Number) Return Number Is

  /* Calcula el nuevo salario base
  Parámetros:
    p_base = Es el salario base actual
    p_porc = Es el porcentaje de aumento a aplicar
  Valor de salida
    vNuevaBase = Nuevo salario base
  Formula =
    vNuevaBase = p_base + (p_base * p_porce / 100)
  Creado por:      Lic. Róger Araya F.      Fecha: 2-Abr-03
  Modificado por:                               Fecha:
  */

  -- Declaración de variables
  vNuevaBase Number(8,2);  -- Nuevo salario base

Begin
  -- Cálculo del nuevo salario base
  vNuevaBase := p_base + (p_base * p_porc / 100);

  -- ...
  -- Cualquier otra cosa que debe realizar el proceso
  -- ...

  -- Retornar el valor calculado
  Return vNuevaBase;

End;
/

```

Identación Recomendada:

```

procedure write_file
(
  p_dataset      in sys_refcursor
, p_separator    in varchar2 default g_dflt_separator
, p_label        in varchar2 default null
, p_heading      in varchar2 default 'N'
, p_rowcount     in varchar2 default 'N'
, p_directory    in all_directories.directory_name%type
, p_filename     in varchar2 )
is
  l_file utl_file.file_type;
  k_dir_path constant all_directories.directory_path%type := directory_path(p_directory);
begin
  if k_dir_path is null then
    raise_application_error(-20006, 'Directory object ''||p_directory||''' is not defined.');
```

```

  end if;

  begin
    l_file :=
      utl_file.fopen
      ( filename => p_filename
      , location => p_directory
      , open_mode => 'w'
      , max_linesize => 4000 );
    exception
      when utl_file.invalid_operation then
        raise_application_error(-20007, 'File ''||p_filename||''' could not be opened in directory ''||p_directory||''' ('||k_dir_path||'), TRUE);
      when utl_file.invalid_path then
        raise_application_error(-20008, 'File location ''||p_directory||''' ('||k_dir_path||') is invalid.', TRUE);
      when utl_file.invalid_filename then
        raise_application_error(-20009, 'Filename ''||p_filename||''' is invalid.', TRUE);
    end;

    for r in (
      select column_value
      from   table(csv.report(p_dataset, p_separator, p_label, p_heading, p_rowcount))
    )
    loop
      begin
        utl_file.put_line(l_file, r.column_value);
      exception
        when utl_file.write_error then
          raise_application_error(-20010, 'Operating system error occurred writing to file ''||p_filename||''');
      end;
    end loop;

    utl_file.fclose(l_file);
  end write_file;

```

V – MANEJO DE VERSIONES DE SOFTWARE

En la programación de formas se debe utilizar una variable para asignar el número de versión (esta variable ya existe en la forma base y toma su valor del parámetro param_version_forma). De esta manera basta con cambiar esta variable para que aparezca la versión correspondiente. Todo programa recién finalizado tendrá como número de versión el valor 1.0. Si después de la evaluación post implementación se efectúa algún ajuste, la versión pasa a ser 1.1.

Conforme sea solicitado el mantenimiento a un sistema se deberá evaluar el tipo de cambio asociado y el impacto en el control de versiones. Si se adiciona una nueva función al programa la versión aumenta al siguiente nivel (de 1.x a 2.0), si el cambio es pequeño, por ejemplo, adicionar un campo o modificar un aspecto en la pantalla la versión aumenta en 0.1 punto. Será conveniente mantener los programas fuentes de la versión actual y anterior.

CAPÍTULO VI: CONCLUSIONES Y RECOMENDACIONES

6.1 Conclusiones

Seguidamente se describen las principales conclusiones de conformidad con los resultados obtenidos con la ejecución de este proyecto

- La metodología para el desarrollo de sistemas de información vigente en la Contraloría General de la República no se adapta a las necesidades actuales de la institución, ni está acorde con los recursos disponibles. La misma no ha sufrido actualizaciones de peso desde el año 2009 lo que ha provocado que no contemple las mejores prácticas de desarrollo de sistemas vigentes en la industria, ni que se responda de manera ágil y eficiente a los desafíos de la institución en materia de sistemas de información.
- En el proceso de desarrollo de sistemas de información que se ejecuta en la CGR, se encontró una serie de oportunidades de mejora, dentro de la que se encuentra la documentación que se solicita en distintas actividades la cual es excesiva, duplicidad de tareas en diferentes fases, falta de claridad en procesos vitales tales como definición de requerimientos, gestión de cambios y evaluación de los productos.
- Los resultados del análisis de la metodología de desarrollo demuestran que las necesidades institucionales en tecnologías de información no se cubren satisfactoriamente, ya que se encontró un descontento de los usuarios con la calidad de los productos entregados, así como atrasos según los cronogramas establecidos.
- No se está dando un entendimiento entre los analistas, desarrolladores y usuarios del sistema, lo que provoca que no se satisfagan las expectativas de los usuarios del sistema y se generen re trabajos por cambios durante la ejecución del proyecto. Debido a lo anterior, surge la necesidad de una mayor integración en los equipos de trabajo, especialmente entre la parte técnica y los usuarios expertos. Asimismo, deben mejorarse las técnicas de recolección de requerimientos promoviendo mayor claridad para la parte técnica de las necesidades de los clientes y los criterios de aceptación de los mismos.
- No existe una métrica clara para medir la calidad de los productos otorgados. En algunos proyectos se omiten las sesiones de evaluación posteriores a la implementación y

también se reconoce que en algunos casos no se documentan las lecciones aprendidas durante el proyecto.

- Muchas de las mejores prácticas identificadas, podrían impactar positivamente el proceso de desarrollo de sistemas vigente. Dentro de esas prácticas destacan las de distintas metodologías ágiles, las cuales han cobrado relevancia en los últimos años en la industria del desarrollo de sistemas.
- No necesariamente adoptar una metodología ágil en su totalidad es beneficioso para la organización. Hay que considerar aspectos como el contexto, formalidades, documentación requerida y cultura organizacional. Por lo tanto, se concluye que debe realizarse una adaptación entre metodologías tradicionales y metodologías ágiles.
- Se desarrolló una propuesta de mejora a la metodología de desarrollo de sistemas de información vigente con el fin de que responda a las necesidades identificadas en el transcurso del desarrollo de este proyecto y que incorpore las mejores prácticas de la industria.
- El migrar de un modelo de desarrollo con la metodología “en cascada” a uno que contemple prácticas ágiles, especialmente iteraciones, donde el usuario tenga una alta participación, promueve un mayor control sobre la calidad del producto otorgado, así como una mejor satisfacción de las necesidades que originaron el proyecto, dado que los usuarios estarán validando durante el proceso los avances del sistema.

6.2 Recomendaciones

A continuación, se detallan las principales recomendaciones dirigidas a la Unidad de Tecnologías de Información:

- Implementar la propuesta de metodología de desarrollo de sistemas de información formulada en el presente proyecto. Con ello, responderá a las necesidades actuales de la institución y contemplará mejores prácticas de la industria.

- Capacitar a todos los actores involucrados en el proceso de desarrollo de sistemas de información con el fin de que conozcan y apliquen la metodología adecuadamente.
- Velar por que todas las etapas, actividades y estándares descritos en la metodología de desarrollo de sistemas, se cumplan en todos los proyectos.
- Implementar las técnicas y actividades recomendadas en la propuesta de metodología, para mejorar el trabajo conjunto y la comunicación que debe darse entre el personal técnico y los usuarios expertos.
- Analizar y evaluar periódicamente la metodología de desarrollo de sistemas, con el fin de que responda a las necesidades institucionales y esté adaptada a las mejores prácticas de la industria.
- Instaurar la calidad como uno de los pilares fundamentales de los proyectos, cumpliendo con las métricas recomendadas y documentando las lecciones aprendidas.
- Solicitar al departamento de publicaciones, estándares para el diseño gráfico de los sistemas de acuerdo con el libro marca institucional.

CAPÍTULO VII: APÉNDICES Y ANEXOS

7.1 Apéndices

Apéndice 1: Encuesta aplicada al equipo de desarrollo de software de la UTI



Josué Solano <jsolam@gmail.com>

Cuestionario Trabajo final de graduación

1 mensaje

Josué Solano <jsolam@gmail.com>
Para: javier.brenes@cgr.go.cr

17 de junio de 2019, 7:18

Buenos días Javier, espero que se encuentre bien.

Tal y como le había comentado, en mi trabajo final de graduación estoy planteando una propuesta de actualización de metodología para el desarrollo de sistemas de información en la CGR.

Como parte del diagnóstico, necesito conocer la percepción del equipo de desarrollo sobre algunos temas, por lo que le agradezco si pudiera enviarles el siguiente link para que me hicieran el grandísimo favor de completar un breve cuestionario.

<https://forms.gle/wAyWSNKuRyNJ7LXx8>

Muchas gracias

—
Josué Solano Morales.

Diagnóstico Metodología para el Desarrollo de Proyectos de TIC

Con el fin de realizar un diagnóstico de la Metodología vigente utilizada para el desarrollo de sistemas de información para efectos de mi trabajo final de graduación, le agradezco responder las siguientes preguntas.

Las preguntas son de respuesta Sí o No por lo que no le tomará más de 5 minutos completar el cuestionario.

Las respuestas son anónimas.

***Obligatorio**

1. 1. Sobre la metodología: *

Marca solo un óvalo por fila.

	Sí	No
¿Conoce a profundidad la metodología para el desarrollo de proyectos de tecnologías de información vigente?	<input type="radio"/>	<input type="radio"/>
¿Considera que la metodología vigente se adapta a las necesidades institucionales y está acorde con los recursos disponibles?	<input type="radio"/>	<input type="radio"/>

2. **¿Considera que antes de iniciar un proyecto se realiza adecuadamente un estudio de factibilidad donde se contemplen elementos básicos como recurso humano, infraestructura, costos, tiempo, herramientas? ***

Marca solo un óvalo.

- Sí
 No

3. **3. Con base en lo establecido en la Metodología vigente, ¿considera que para una adecuada ejecución de tareas en proyectos de desarrollo se definen adecuadamente los roles del... ***

Marca solo un óvalo por fila.

	Sí	No
... patrocinador del proyecto?	<input type="radio"/>	<input type="radio"/>
... líder del proyecto?	<input type="radio"/>	<input type="radio"/>
... líder tecnológico?	<input type="radio"/>	<input type="radio"/>
... coordinador de proyectos?	<input type="radio"/>	<input type="radio"/>

4. **4. Considera que ... ***

Marca solo un óvalo por fila.

	Sí	No
... el informe de diagnóstico que debe elaborarse en la etapa de iniciación del proyecto, es de su utilidad para el desarrollo de la aplicación?	<input type="radio"/>	<input type="radio"/>
... los requerimientos funcionales y no funcionales para el desarrollo de la aplicación se definen adecuadamente por parte de los futuros usuarios del sistema?	<input type="radio"/>	<input type="radio"/>
... las técnicas utilizadas para la especificación de requerimientos por parte de los usuarios son las más adecuadas?	<input type="radio"/>	<input type="radio"/>

5. **5. ¿En caso de requerir mayores detalles sobre algún requerimiento, las personas que los definieron se encuentran disponibles para atender oportunamente sus dudas? ***

Marca solo un óvalo.

- Sí
 No

6. 6. En el cronograma de tareas... *

Marca solo un óvalo por fila.

	Sí	No
... ¿se documentan todas las tareas a realizar?	<input type="radio"/>	<input type="radio"/>
... ¿se define adecuadamente la secuencia de dichas tareas para una adecuada gestión del tiempo?	<input type="radio"/>	<input type="radio"/>
... ¿se cumplen los plazos establecidos?	<input type="radio"/>	<input type="radio"/>

7. 7. ¿Considera que cuenta con las herramientas necesarias para el desarrollo de aplicaciones (Infraestructura, aplicaciones, capacitación)? *

Marca solo un óvalo.

- Sí
 No

8. 8. ¿Considera que la documentación del diseño conceptual (Identificación de módulos, diseño de interfaz de usuario, integración de sistemas e identificación de roles usuarios) es de su utilidad para el desarrollo de aplicaciones? *

Marca solo un óvalo.

- Sí
 No

9. 9. ¿Elabora un diseño detallado del sistema que comprende las siguientes etapas... *

Marca solo un óvalo por fila.

	Sí	No
... Descripción de procesos	<input type="radio"/>	<input type="radio"/>
... Diagrama lógico del modelo de datos	<input type="radio"/>	<input type="radio"/>
... Diseño físico de la base de datos	<input type="radio"/>	<input type="radio"/>
... Estimación del volumen de datos	<input type="radio"/>	<input type="radio"/>
... Definición de controles y seguridad a utilizar	<input type="radio"/>	<input type="radio"/>
... Organización para la operación del sistema	<input type="radio"/>	<input type="radio"/>

10. **10. ¿Considera que la documentación del diseño detallado, es útil para el desarrollo de la aplicación? ***

Marca solo un óvalo.

- Sí
 No

11. **11. Se define un diseño de pruebas que contemple lo siguiente:... ***

Marca solo un óvalo por fila.

	Sí	No
... Especificación de tipos de pruebas	<input type="radio"/>	<input type="radio"/>
... Requerimientos para las pruebas	<input type="radio"/>	<input type="radio"/>
... Casos y datos de prueba	<input type="radio"/>	<input type="radio"/>
... Usuarios para las pruebas	<input type="radio"/>	<input type="radio"/>

12. **12. ¿Los usuarios ejecutan las pruebas en el tiempo definido dentro del cronograma del proyecto? ***

Marca solo un óvalo.

- Sí
 No

13. **13. ¿Considera que se gestionan adecuadamente los cambios realizados al alcance del proyecto? ***

Marca solo un óvalo.

- Sí
 No

14. **14. ¿Considera que existe suficiente flexibilidad en la gestión de los cambios en procura de la consecución de los objetivos? ***

Marca solo un óvalo.

- Sí
 No

15. **15. ¿Conoce los estándares de nomenclatura definidos en la Metodología para el Desarrollo de Proyectos de TIC? ***

Marca solo un óvalo.

- Sí
 No

16. **16. ¿Aplica los estándares de nomenclatura definidos en la Metodología para el Desarrollo de Proyectos de TIC? ***

Marca solo un óvalo.

- Sí
 No

17. **17. ¿Se coordina adecuadamente con el líder del proyecto la implementación de las nuevas aplicaciones? ***

Marca solo un óvalo.

- Sí
 No

18. **18. ¿Se ejecuta alguna sesión del equipo de proyecto para evaluar la aplicación desarrollada y discutir lecciones aprendidas? ***

Marca solo un óvalo.

- Sí
 No

19. **19. ¿Tiene conocimiento sobre las principales metodologías ágiles utilizadas actualmente en la industria del desarrollo de software? ***

Marca solo un óvalo.


- Sí
 No

20. **20. ¿Considera que la aplicación de metodologías ágiles sería favorable en el desarrollo de aplicaciones en la institución? ***

Marca solo un óvalo.

- Sí
 No

21. 21. Indique las principales deficiencias que considera que posee la metodología de desarrollo vigente *

Con la tecnología de
 Google Forms

BIBLIOGRAFÍA

- 12 Principles Behind the Agile Manifesto. (2015, noviembre 4). Recuperado 19 de abril de 2019, de Agile Alliance website: <https://www.agilealliance.org/agile101/12-principles-behind-the-agile-manifesto/>
- Asamblea Legislativa de la República de Costa Rica. *Ley Orgánica de la Contraloría General de la República.* , Pub. L. No. 7428 (1994).
- Asamblea Nacional Constituyente. *Constitución Política de la República de Costa Rica.* , (1949).
- Auditoría Interna - CGR. (2018). *Estudio sobre los controles implementados por la Unidad de Tecnologías de Información en la creación y operación de las aplicaciones* (N.º I-AIG-09-2018). Contraloría General de la República.
- Calvo, E., & Ramírez, M. (2016). *Guía Metodológica para la Gestión de Proyectos en la Dirección General del Servicio Civil* (Proyecto de graduación para optar por el grado académico de Maestría en Gerencia de Proyectos.). Instituto Tecnológico de Costa Rica, San José.
- Canós, J, Letelier, P, & Penadés, M. (2003). *Métodologías Ágiles en el Desarrollo de Software. Metodologías Ágiles en el Desarrollo de Software.* Presentado en *Metodologías Ágiles en el Desarrollo de Software*, Alicante.
- Castro, C, G., A. (2013). *Metología de gestión de Proyectos TIC: Administración de los recursos de la información.* (Primera). Buenos Aires: Edulp.
- Chiquito, D. (2017). *Análisis e implementación de gestión de procesos de negocios (bpm) y metodologías ágiles para optimizar el desempeño funcional del proceso de desarrollo de*

- software, en una empresa privada, de soluciones tecnológicas.* Escuela Superior Politécnica del Litoral, Guayaquil, Ecuador.
- Contraloría General de la República. *Normas técnicas para la gestión y el control de las Tecnologías de Información.* , Pub. L. No. N-2-2007-CO-DFOE (2007).
- Contraloría General de la República. *Reglamento Orgánico de la Contraloría General de la República.* , Pub. L. No. R-1-2011-DC-GC (2011).
- Contraloría General de la República. (2013). *Plan Estratégico Institucional 2013-2020.* Publicaciones CGR.
- Cruz, N., & González, A. (2018, julio 13). ¿Tradicional o ágil? La metodología ágil como alternativa a la transformación. *Axpe Consulting.*
- Dominguez, L. (2012). *Análisis de sistemas de información* (Primera Edición). Tlalnepantla, México: Red Tercer Milenio.
- Estructura organizacional "CGR | Costa Rica. (s. f.). Recuperado 23 de febrero de 2019, de <https://www.cgr.go.cr/01-cgr-transp/organigrama/organigrama.html>
- Fernández, K., Garrido, A, Ramírez, Y., & Perdomo, I. (2015). *PMBOK y PRINCE2, Similitudes y Diferencias.* *Revista Científica Universidad Distrital Francisco Jose de Caldas*(23).
- Hernández, R., Fernández, C., & Baptista, M. (2014). *Metodología de la investigación* (Sexta). México D.F.: McGraw Hill Education.
- ISACA. (2012). *COBIT 5: Procesos Catalizadores.* Estados Unidos: ISACA.
- Laudon, K., & Laudon, J. (2016). *Sistemas de Información Gerencial* (Decimocuarta). México: Pearson Educación.
- Lozada, J. (2014). *Investigación Aplicada: Definición, Propiedad Intelectual e Industria.* *Cienciamérica, Universidad Tecnológica Indoamericana, 3.*

- Organización de Cooperación y Desarrollo Económico. (2006). *La modernización del Estado: El camino a seguir*. Instituto Nacional de Administración Pública.
- Organización Internacional de las Entidades Fiscalizadoras Superiores. (2013). *Declaraciones de Lima y México*. INTOSAI General Secretariat.
- Organización para la Cooperación y el Desarrollo Económicos. (2011). *Buenas Prácticas para el Apoyo a Entidades Fiscalizadoras Superiores (EFS)*. Recuperado de https://www.oecd.org/dac/effectiveness/Buenas_Pr%C3%A1cticas_para_el_Apoyo_a_Entidades_Fiscalizadoras_Superiores-%20for%20web.pdf
- pmoinformatica.com. (2017). La Guía del PMBOK 6ta edición y las metodologías ágiles. Recuperado 4 de mayo de 2019, de <http://www.pmoinformatica.com/2017/12/pmbok-6-metodologias-agiles.html>
- Pressman, R. (2010). *Ingeniería del Software. Un enfoque práctico* (Séptima). México.
- Project Management Institute. (2017). *Guía de los Fundamentos para la Dirección de Proyectos (Guía del PMBOK)* (Sexta).
- Real Academia Española. (2014). Diccionario de la lengua española. Recuperado 11 de abril de 2019, de Diccionario de la lengua española "Edición del Tricentenario website: <http://dle.rae.es/>
- Rivera, F. (2015). *Administración de proyectos. Guía para el aprendizaje* (Segunda). México: Pearson Educación.
- Schwaber, K, & Sutherland, J. (2017). *La Guía de Scrum*. Recuperado de <https://www.scrumguides.org/docs/scrumguide/v2017/2017-Scrum-Guide-Spanish-SouthAmerican.pdf>
- Sommerville, I. (2011). *Ingeniería de Software* (Novena). México.

- Sousa, K, & Oz, E. (2017). *Administración de los sistemas de información* (séptima). México: Cengage Learning.
- Stair, R., & Reynolds, G. (2017). *Principios de Sistemas de Información* (Décima). México: Cengage Learning.
- Stauffer, A. (2018, junio 26). How One Team is Making the Transition to Agile. *DigitalGov*. Recuperado de <https://digital.gov/2018/06/26/how-one-team-making-transition-agile/>
- Unidad de Sistemas y Tecnologías de la Información - CGR. *Metodología para el desarrollo de proyectos de tecnología de información y comunicaciones.* , (2016).
- Vélez, S., Zapata, A., & Henao, A. (2018). Gestión de Proyectos: Origen, instituciones, metodologías, estándares y certificaciones. *Entre Ciencia e Ingeniería*, 12(24).